The International Defense and Homeland Security Simulation Workshop

September 12-14 2011
Rome, Italy

EDITED BY
Agostino Bruzzone
Wayne Buck
John A. Sokolowski

Printed in Rende (CS), Italy, September 2011

ISBN 978-88-903724-3-8

THE INTERNATIONAL DEFENSE AND HOMELAND SECURITY SIMULATION WORKSHOP, DHSS 2011

September 12-14 2011

Rome, Italy

Organized by

DIPTEM – University of Genoa

Liophant Simulation

Simulation Team

IMCS – International Mediterranean & Latin American Council of Simulation

Mechanical Department, University of Calabria

MSC-LES, Modeling & Simulation Center, Laboratory of Enterprise Solutions

Modeling and Simulation Center of Excellence (MSCOE)

Miss Latvian Center – Riga Technical University

Logisim

LSIS – Laboratoire des Sciences de l'Information et des Systemes

MIMOS – Movimento Italiano Modellazione e Simulazione

MISS – University of Perugia

MISS – Brasilian Center, LAMCE-COPPE-UFRJ

MISS - McLeod Institute of Simulation Sciences

M&SNet - McLeod Modeling and Simulation Network

LATVIAN SIMULATION SOCIETY

ECOLE SUPERIEURE D'INGENIERIE EN SCIENCES APPLIQUEES

FACULTAD DE CIENCIAS EXACTAS. INGEGNERIA Y AGRIMENSURA

UNIVERSITY OF LA LAGUNA

CIFASIS: CONICET-UNR-UPCAM

INSTICC - INSTITUTE FOR SYSTEMS AND TECHNOLOGIES OF INFORMATION, CONTROL AND COMMUNICATION

I3M 2011 INDUSTRIAL SPONSORS

PRESAGIS

CAE

CAL-TEK

MAST

AEGIS TECHNOLOGIES

I3M 2011 MEDIA PARTNERS

MILITARY SIMULATION & TRAINING MAGAZINE

EURO MERCI

EDITORS

AGOSTINO BRUZZONE
MISS-DIPTEM, UNIVERSITY OF GENOA, ITALY
agostino@itim.unige.it

WAYNE BUCK
HQ SUPREME ALLIED COMMANDER TRANSFORMATION, USA
Wayne.Buck@act.nato.int

JOHN A. SOKOLOWSKI
VIRGINIA MODELING, ANALYSIS AND SIMULATION CENTER, USA
JSokolow@odu.edu

THE INTERNATIONAL MEDITERRANEAN AND LATIN AMERICAN MODELING MULTICONFERENCE, I3M 2011

GENERAL CO-CHAIRS

AGOSTINO BRUZZONE, *MISS DIPTEM, UNIVERSITY OF GENOA, ITALY*
MIQUEL ANGEL PIERA, *AUTONOMOUS UNIVERSITY OF BARCELONA, SPAIN*

PROGRAM CHAIR

FRANCESCO LONGO, *MSC-LES, UNIVERSITY OF CALABRIA, ITALY*

THE INTERNATIONAL DEFENSE AND HOMELAND SECURITY SIMULATION WORKSHOP, DHSS 2011

GENERAL CO-CHAIRS

AGOSTINO BRUZZONE, *MISS-DIPTEM, UNIVERSITY OF GENOA, ITALY*
JOHN SOKOLOWSKI, VMASC, *USA*

PROGRAM CHAIR

WAYNE BUCK, *HQ SUPREME ALLIED COMMANDER TRANSFORMATION, USA,*

DHSS 2011 INTERNATIONAL PROGRAM COMMITTEE

David Alberts, OSD, USA
Tayfur Altiok, Rutgers University, USA
Linda Brent, The Asta Group, USA
Agostino Bruzzone, University of Genoa, Italy
Wayne Buck, HQ Supreme Allied Commander Transformation, USA
Giovanni Cantice, Italian Army, Italy
Erdal Cayirci, University of Stavanger, Norway
Franco Cotugno, SME, IV Reparto Logistico, Advanced Technologies Office, Italy
Gianluca De Leo, ODU, USA
Jean-Yves Donnart, THALES Training and Simulation, France
Gene Fredriksen, Tyco International, USA
Claudia Frydman, LSIS, France
Peggy Graviz, The Aegis, USA
Drew Hamilton, Auburn University, USA
John Illgen, Northrop Grumman, USA
Peter Kincaid, IST UCF, USA
Sam Kukich, US Coast Guard, USA
Fred Lewis, NTSA, USA
Francesco Longo, University of Calabria, Italy
Robert Lutz, JHU/APL, USA
Francesco Mastrorosa, M&S COE, Italy
Agatino Mursia, Selex Comm, Italy
Jackson P. Nelson , NTSA, USA
Tuncer Ören, University of Ottawa, Canada
Paolo Proietti, Mimos, Italy
Stefano Saetta, University of Perugia, Italy
Ferdinando Scarazzini, CIMSO COI, Italy
Rick Severinghaus, The Aegies, USA
John Sokolowski, VMASC, USA
Robert Sottilare, Army Research Laboratory, USA
Alberto Tremori, Simulation Team, Italy
William Tucker, The Boeing Company, USA
Walter F. Ullrich, HallDale Media Group, Germany
Levent Yilmaz, Auburn University, USA
Simone Youngblood, John Hopkins University, USA
Bill Waite, The Aegis, USA
Richard Zaluski, GICSR, USA
Aldo Zini, Cetena, Italy

TRACKS AND WORKSHOP CHAIRS

ADAPTIVE AND PREDICTIVE COMPUTER-BASED TUTORING
CHAIR: ROBERT SOTTILARE, ARMY RESEARCH LABORATORY, USA

COLLECTIVE "C2" & CYBER SECURITY
CHAIR: DAVID ALBERTS, OFFICE OF THE ASSISTANT SECRETARY OF DEFENSE (OASD), USA

CYBERSECURITY
CHAIRS: RICHARD ZALUSKI, GICSR, USA; GENE FREDRIKSEN, TYCO INTERNATIONAL, USA

HEALTHCARE SIMULATION & HOMELAND SECURITY
CHAIRS: RICHARD BOYD, LOCKHEED MARTIN, USA; GIANLUCA DE LEO, OLD DOMINIUM UNIVERSITY, USA

HUMAN SOCIAL CULTURAL BEHAVIOUR (HSCB) MODELING
CHAIR: JOHN A. SOKOLOWSKI, VIRGINIA MODELING, ANALYSIS AND SIMULATION CENTER, VA, USA

INNOVATIVE OPERATIONS PLANNING DECISION SUPPORT: AUGMENTED REALITY SOLUTIONS
CHAIR: MICHELE TURI, CESIVA. ITALY

LVC SIMULATION ARCHITECTURES: STATE OF THE ART AND FUTURE DEVELOPMENTS
CHAIR: AGATINO MURSIA, SELEX-COMMS, ITALY

MULTILEVEL SIMULATION TO SUPPORT TRAINING AND OPERATIVE SIMULATION: TOOLS AND FEDERATIONS, CURRENT SITUATION AND WAY AHEAD
CHAIR: FERDINANDO SCARAZZINI, CIMSO-COI, ITALY

PORTS AND LITTORAL PROTECTION
CHAIRS: TAYFUR ALTIOK, RUTGERS UNIVERSITY, USA; FRANCESCO LONGO, UNIVERSITY OF CALABRIA, ITALY

SERIOUS GAMES: TOWARD AN EFFECTIVE TRAINING
CHAIR: PAOLO PROIETTI, MIMOS, ITALY

URBAN AREA SECURITY
CHAIRS: ALBERTO TREMORI, SIMULATION TEAM, ITALY; TAYFUR ALTIOK, RUTGERS UNIVERSITY, USA

General Co-Chairs' Message

Welcome To DHSS 2011!

The International Defense and Homeland Security Simulation Workshop, DHSS 2011, has been established for the first time this year in Rome. The main idea behind DHSS 2011 is to provide experts and scientists with a dedicated forum in which Modeling & Simulation innovations in terms of methodologies, technologies and architectures for Defense and Homeland Security applications where to discuss critical issues and future research and development activities.

In fact these thematic areas were already present in previous editions of I3M Multiconference, therefore this year, the Organizers decided to group them in a this new specific workshop DHSS2011; the decision allowed to create synergies with other major international initiatives in Homeland Security and Defense M&S communities that lead to the great success of I3M2011 in Rome. In this sense the DHSS success is resulting from the concrete efforts, carried out by the Organization Committee, to reinforce networking in this strategic R&D areas and to create synergies with the newly born Modeling & Simulation Center of Excellence (MS-COE) just established in Rome. To this end, it was also decided to co-locate DHSS 2011 in the same period and location of the NATO CAX Forum (organized by MS-COE) and to provide DHSS attendees with the opportunity to attend (free of charge) the all I3M Conferences and CAX Forum sessions.

Consequently, DHSSis a unique opportunity to meet researchers and practitioners, not only from the Academia, but also from Defense and Homeland Security sectors (i.e. Agencies, Institutions, Governmental Organizations and Services) as well as from Industry, especially considering even that this workshop is part of the large International Mediterranean and Latin American Modeling Multiconference, I3M (that includes also EMSS, MAS, IMAACA and HMS) that in the 2011 edition is involving several hundreds *simulationists* from 48 different countries.

In addition it is important to outline that DHSS provide an interesting framework for industrial defense and airspace business sector; traditionally the industry presence is assured by the I3M events (i.e. EMSS historically known as the Simulation in Industry Symposium), however this year several additional companies will be present with solutions centered on DHSS topics; this will guarantee to have an interesting exhibition area in which several companies will have the opportunity to present their cutting-edge M&S technologies.

Therefore DHSS 2011 is mainly a "scientific" workshop: among many initial submissions, the DHSS 2011 International Program Committee has selected a restricted number of papers based on technical, scientific and scholarly quality parameters; therefore a particular thanks goes to authors and reviewers: their invaluable work in terms of reciprocal feedback has surely contributed to increase the quality of DHSS 2011.

Least but not last, DHSS 2011 will be held in the wonderful framework of Rome, the eternal city! With roughly 3000 years of history and with the epithet of "Roma Caput Mundi", Rome unique beauties and attractions (the Vatican City, the Colosseum, the Spanish Square, the Roman Forum, etc.) offer to its visitors a travel "around the World & along the Years".

We wish you a fruitful conference and a pleasant stay in Rome.

Agostino G. Bruzzone
Simulation Team MISS DIPTEM
University of Genoa, Italy

John Sokolowski
VMASC
Old Dominion University, USA

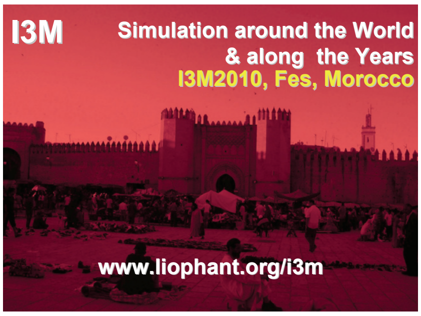
ACKNOWLEDGEMENTS

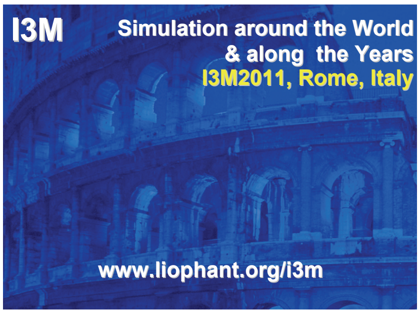
The DHSS 2011 International Program Committee (IPC) has selected the papers for the Conference among many submissions; therefore, based on this effort, a very successful event is expected. The DHSS 2011 IPC would like to thank all the authors as well as the reviewers for their invaluable work.
A special thank goes to all the organizations, institutions and societies that have supported and technically sponsored the event.

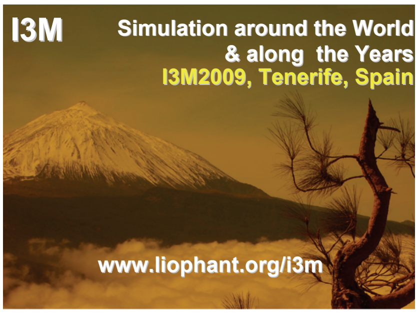
LOCAL ORGANIZATION COMMITTEE

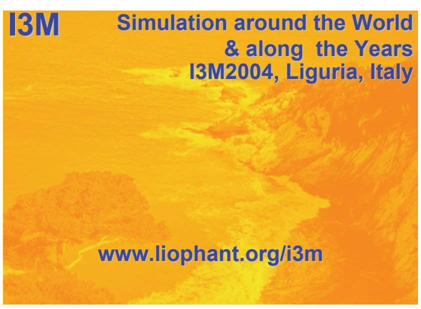
AGOSTINO G. BRUZZONE, *MISS-DIPTEM, UNIVERSITY OF GENOA, ITALY*
ENRICO BOCCA, *SIMULATION TEAM, ITALY*
FRANCESCO LONGO, *MSC-LES, UNIVERSITY OF CALABRIA, ITALY*
FRANCESCA MADEO, *UNIVERSITY OF GENOA, ITALY*
MARINA MASSEI, *LIOPHANT SIMULATION, ITALY*
LETIZIA NICOLETTI, *CAL-TEK SRL*
FEDERICO TARONE, *SIMULATION TEAM, ITALY*
ALBERTO TREMORI, *SIMULATION TEAM, ITALY*

This International Conference is part of the I3M Multiconference: the Congress leading *Simulation around the World and Along the Years*









Index

INTERFACE FORM
FOR AN UNDERWATER WARFARE SIMULATION ENVIRONMENT

Kyung-Min Seo[(a)], Changbeom Choi[(b)], Jung Hoon Kim[(c)] and Tag Gon Kim[(d)]

[(a)(b)(d)]Dept. of Electrical Engineering, Korea Advanced Institute of Science and Technology
[(c)]Naval Systems R&D Institute, Agency for Defense Development

[(a)]kmseo@smslab.kaist.ac.kr, [(b)]cbchoi@sim.kaist.ac.kr, [(c)]kimjh@add.re.kr, [(d)]tkim@ee.kaist.ac.kr

ABSTRACT
The effectiveness analysis is influenced by the Measures of Effectiveness (MoEs), Measures of Performance (MoPs), alternatives, threats, scenarios, operation concepts, etc. For effectiveness analysis, modeling and simulation (M&S) technology is an important method, which is used to evaluate numerous designs and operational concepts for a real-world system. This paper proposes implementation of Interface Forms (I/Fs), which operates somewhat like experimental frame. Proposed illustrates how to design an experimental frame for appropriate modeling objectives. The experimental result shows that we can test alternative tactics and the behavior analysis was successful.

Keywords: Experimental frame, DEVS formalism, underwater warfare model

1. INTRODUCTION

The effectiveness analysis is designed to compare the effectiveness of the alternatives based on military worth (Office of Aerospace Studies 2008). It is influenced by the Measures of Effectiveness (MoEs), Measures of Performance (MoPs), alternatives, threats, scenarios, operation concepts, etc. From the simulation point of view, MoPs are typically a quantitative measure of a system characteristic. For example, the speed of a missile hitting its target is a performance measure. Such measures enter as factors into outcome measures, often called MoEs, that measure how well the overall system goals are being achieved (e.g., how many battles are actually won by a particular combination of weapons, platforms, personnel).

For effectiveness analysis, modeling and simulation (M&S) technology is an important method, which is used to evaluate numerous designs and operational concepts for a real-world system. M&S technology facilitates decisions about future equipment procurements such as a mobile decoy or a torpedo. In addition, assessment of submarine tactical development during an engagement against a torpedo can be conducted using M&S techniques.

A framework for M&S is divided into a system and an experimental frame. The system refers to a set of interacting or interdependent components that we are interested in modeling. It may be a real or virtual environment. An experimental frame is a specification of the conditions under which the system is observed or experimented with. Once the models are built, their effectiveness has to be analyzed. Therefore, various experiments need to be generated to evaluate various effectiveness analyses. The experimental frame is capable of generating different experiments needed to evaluate the system effectiveness. In the experimental frame, various scenarios can be set up and the MoEs, which are collected, can also be specified. In this case, simulations of flexible combinations are possible, such as alternatives scenarios with an experimental frame.

The objective of this paper is the implementation of Interface Forms (I/Fs) for an underwater warfare simulator. I/Fs operate somewhat like experimental frames, as described earlier. Proposed I/Fs provide the developed simulator platform information and tactical information, and we observe the simulation result and analyze the result with the proposed I/F. This paper contributes to the defense M&S community in two ways:

- It illustrates how to design an experimental frame for appropriate modeling objectives
- It provides flexible experimental frames to provide insights about how various factors, such as tactics and the performance of underwater weapons, influence the MOEs of the system.

The structure of this paper is as follows. Section 2 presents a framework for M&S and the DEVS formalism. Section 3 explains Interface Forms (I/Fs) for the DEVS-based underwater simulator, and Section 4 illustrates some case studies and experimental results. Finally, Section 5 concludes this research and proposes future extension for a more complete solution.

2. RELATED WORK

We first introduce a framework for M&S. We also introduce the DEVS formalism that we apply for modeling the underwater warfare system in this paper.

2.1. Framework for modeling and simulation (M&S)

This subsection is devoted to establishing a framework for modeling and simulation (M&S). As can be seen in Figure 1, the basic entities of the framework are the source system, model, simulator, and experimental frame. The source system is the real or virtual environment that we are interested in modeling. It is viewed as a source of observable data. The data that has been gathered from observing or otherwise experimenting with a system is called the system behavior database. A model is a system specification that is a set of instructions, rules, and equations. In other words, we write a model with a state transition and output generation mechanisms to accept input trajectories and generate output trajectories, depending on its initial state setting. As a set of instructions, a model needs some agent capable of actually obeying the instructions and generating behavior. We call such an agent a simulator. Therefore, a simulator is any computation system capable of executing a model to generate its behavior.

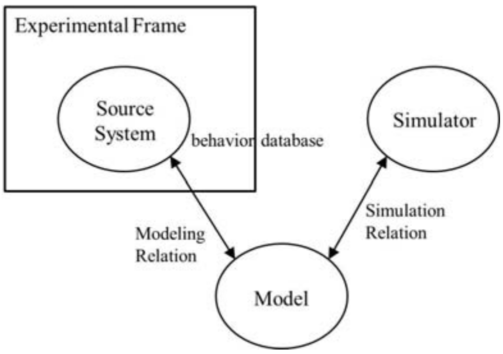


Figure 1: Basic entities in M&S and their relationships

Seo and Song(2011) proposed to design an underwater warfare modeling methodology using the DEVS formalism. For more efficient model development, they propose a generic three-part underwater platform model, which is flexible enough to be easily re-usable for developing different underwater platform models with different behaviors and structures. They developed a simulator using DEVSim++, which was developed by Park and Kim(1996) at KAIST in Korea. The developed simulation supports users in evaluating the effectiveness of underwater warfare systems through Monte Carlo simulation and assesses tactical development and anti-torpedo countermeasure effectiveness. In this paper, we use Seo and Song(2011)'s underwater warfare model and simulator and focus on how to develop I/Fs for an efficient experimental frame. In the subsection, we will describe an experimental frame in more detail.

2.1.1. Experimental Frame

An experimental frame is a specification of the conditions under which the system is observed or experimented with. As such, an experimental frame is the operational formulation of the objectives that motivate a modeling and simulation project.

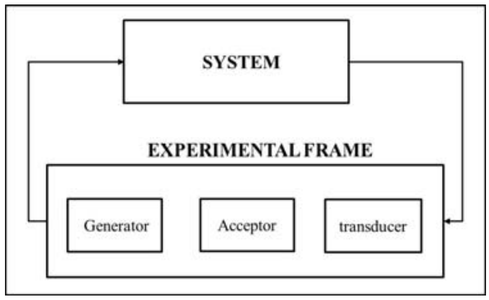


Figure 2: Experimental frame and its component

As described in Figure 2, an experimental frame typically has three types of components: a generator, which generates inputs to the system; an acceptor, which monitors an experiment to see that the desired experimental conditions are met; and a transducer, which observes and analyzes the system outputs. In practice, many experimental frames can be formulated for the same system. This means that we might have different objectives in modeling the same system. For example, in underwater warfare, we can evaluate the survival rate of our submarine according to various maneuver patterns for detour when opposing torpedoes are approaching. In this paper, we proposed two kinds of I/Fs for the experimental frame. The first I/F takes on the role of a generator and the second I/F performs the role of an acceptor and a transducer.

2.2. DEVS Formalism

The DEVS Formalism is general formalism for discrete event system modeling based on set theory, and it is one of the M&S theories which are applied in various military simulations (Zeigler, Praehofer, and Kim 2000). The DEVS Formalism supports to specify the discrete event models in hierarchical and modular manner. The DEVS Formalism exhibits the concepts of system theory and modeling, and with this formalism, the user can model the target system by decomposing large system into smaller components which coupling scheme among them. There are two kinds of models in the formalism: Atomic model and Coupled model.

The Atomic model is a specification of basic model behavior as timed state transition. Formally, an Atomic model can be defined by 7-tuples as follows:

$$M = < X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta >,$$

Where

X: a set of input;

Y: a set of output events;

S: a set of sequential states;

δ_{ext}: $Q \times X \rightarrow S$, an external transition function, where $Q = \{(s,e) | s \in S, 0 \leq e \leq ta(s)\}$ is the total state set of M;

δ_{int} : $S \rightarrow S$, an internal transition function;

λ : $S \rightarrow Y$, an output function;

ta : S → Real, time advance function.

Coupled model is a specification of hierarchical model structure. It provides the method of assembly of atomic and/or coupled models to build the hierarchy of complex system. Formally, a Coupled model is defined as follow;

$$DN = < X, Y, M, EIC, EOC, IC>,$$

Where

X: a set of input;
Y: a set of output events;
M: a set of all component models;
EIC $\subseteq$ DN.X × $\cup$M.X: external input coupling;
EOC $\subseteq$ $\cup$M.Y × DN.Y: external output coupling;
IC $\subseteq$ $\cup$M.Y × $\cup$M.X: internal coupling;
SELECT: $2^M - \emptyset \rightarrow M$: tie-breaking selector.

3. INTERFACE FORMS FOR DEVS BASED UNDERWATER WARFARE SIMULATOR

The underwater warfare model, which was developed by Seo and Song(2011), consists only of the core of the simulation software, so an experimental frame is needed to utilize the simulation model. I/Fs include an experimental frame and interface between the simulation software and an experimental frame. For example, a detailed human computer interaction interface will be needed in the use of simulation training, or a statistical result organizer will be needed to run a simulation experiment. These interfaces will provide the simulation results; we call this an experimental frame, as described earlier. In this section, we propose two kinds of I/Fs as an experimental frame for the DEVS-based underwater warfare simulator.

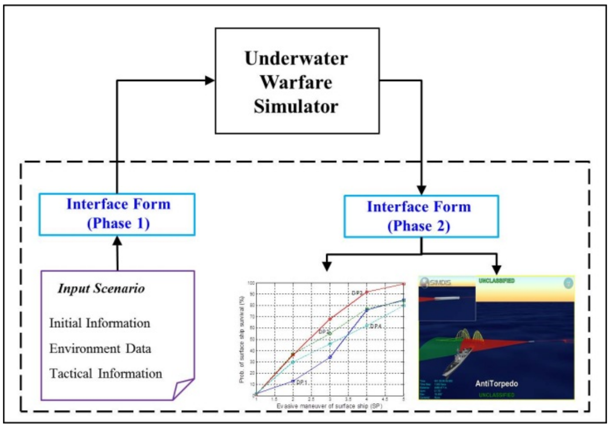


Figure 3: Interface Forms for Experimental Frame

Figure 3 illustrates the high-level view of the framework for underwater warfare M&S. We proposed two kinds of I/Fs: Phase I, which is the scenario identification I/F, and Phase II, which is for simulation analysis. The scenario identification I/F provides the underwater warfare simulator scenario information, such as the initial parameters for platforms and environmental and tactical information. Specifically, scenario identification I/F takes on the role of supporting the model to determine the manner of action dynamically, according to the predefined tactics in the I/F. This means that we can combine several tactical modules with this I/F to achieve the mission purpose when the simulator has these tactical modules and we know these modules. For example, suppose that the simulator provides several maneuver patterns, such as straight, snake, circular, or turn maneuvers. In this case, effectiveness, like the mission success rate, will vary depending on how well several maneuver patterns can be combined. Therefore, the scenario identification I/F enables users to assess alternative tactical deployments for maneuver patterns. In the case of the M&S framework without the proposed scenario identification I/F, there are problems, such as rewriting and modifying the model every time tactical information is changed, because tactics should be defined statically in the model. Scenario identification I/F does have a benefit, including the simulation of the various scenarios without modifying the model when tactics are changed; this is with the modifying scenario description I/F only.

The simulation analysis I/F takes the role of verifying the behavior analysis of simulation. This I/F provides the user graphical traces of the platform and Monte Carlo simulation. From a display perspective, simulation analysis I/F provides the common structure that can be shown in the simulation display tool. In this paper, we used SIMDIS for the display tool, which is a set of software tools that provide two- and three-dimensional interactive graphical and video displays of live and post-processing simulation, tests, and operational data (U.S. Naval Research Laboratory, 2006). We will describe the two I/Fs in more detail in the subsection.

3.1. Phase I : Scenario Identification

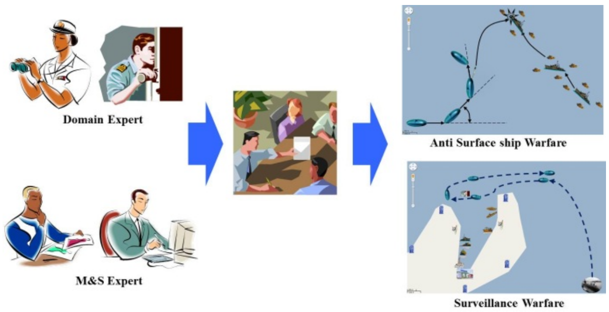


Figure 4: Collaborative Process between Domain and M&S Engineers

In order to identify scenario information in the domain specific system, it needs to cooperate with the domain and M&S engineers. In other words, a domain-specific model, such as a military model, is developed with the integration of domain knowledge and M&S methodology. A domain engineer is involved in performing the domain requirement analysis and design, and an M&S engineer is in charge of the overall process related to the M&S of discrete event systems satisfying the domain requirements. It would be difficult for the M&S engineer to identify scenario information to develop domain-specific models solely using his M&S

knowledge. We call this stage the requirement analysis (Sung, Moon, and Kim 2010). Figure 4 shows the collaborative process between the domain and M&S engineers for scenario identification. Requirement analysis will require the participation of the domain engineer and M&S engineer because the M&S engineer cannot develop detailed model design without the domain knowledge. Domain information is often gathered through questionnaires or direct interviews with domain engineers. Domain and M&S engineers define the M&S objectives and the overall functions of the simulation software by distilling the domain information. As a result of this stage, the domain engineers develop textual descriptions, called requirement specifications, of the software.

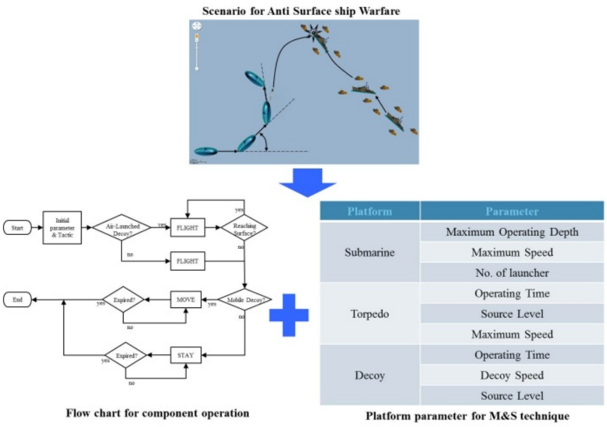


Figure 5: Scenario Identification

With these requirement specifications, the M&S engineer identifies platforms to be modeled, parameters to be used in each platform, and military tactical information. Figure 5 describes this process. This information is utilized for the input information of the scenario identification I/F.

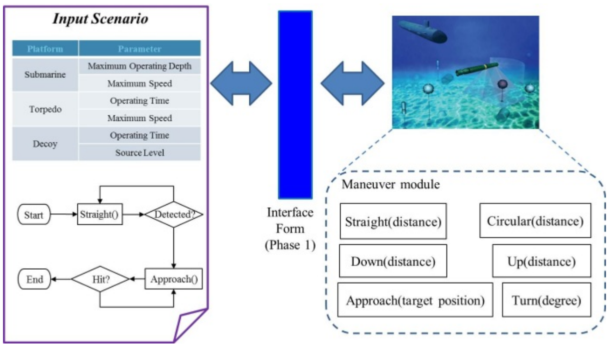


Figure 6: I/F for Scenario Identification

Next, we identified inputs for scenario identification I/F, which are platform parameters and tactical information. Figure 6 shows the relationship among input scenario, I/F, and simulator. The underwater vehicle, in Figure 6, has six maneuver modules. We can composite several maneuver modules for the maneuver tactic. As described earlier, effectiveness, like the mission success rate, will vary depending on how well several maneuver patterns are combined.

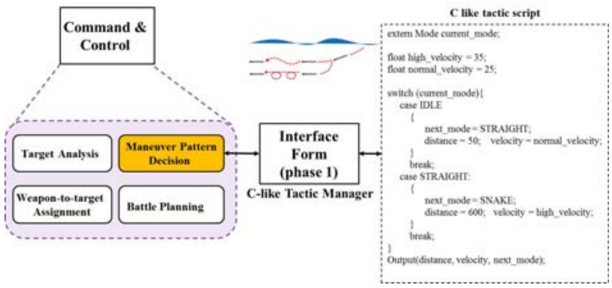


Figure 7: Scenario Identification I/F using C-like Tactic Manager

Errore. L'origine riferimento non è stata trovata. shows implementation of I/F using a C-like tactic manager. The C-like tactic script is influenced by the simulator and the simulator simulates according to the C-like tactic script. A user can modify the script during simulation and the modified script is reflected immediately. Therefore, the user can test and evaluate various tactics during simulation. In the case of the simulation model without this I/F, there are problems, such as rewriting and modifying the model every time tactics are changed, because tactics should be defined statistically in the model.

3.2. Phase II : Simulation Analysis

In this subsection, we will describe the second I/F for simulation analysis. This I/F takes the role of verifying the behavior analysis of simulation. The I/F provides the user graphical traces of the platform and Monte Carlo simulation. After the I/F is established, the simulation software will be verified and validated

. As described in Figure 8, M&S engineers test the simulator to check the accuracy of converting a model representation into simulation software. We call this the simulation verification. After verifying that the model is implemented as designed, the statistical analysis will follow to compare the simulated data to the real-world data; this procedure is the simulation validation.

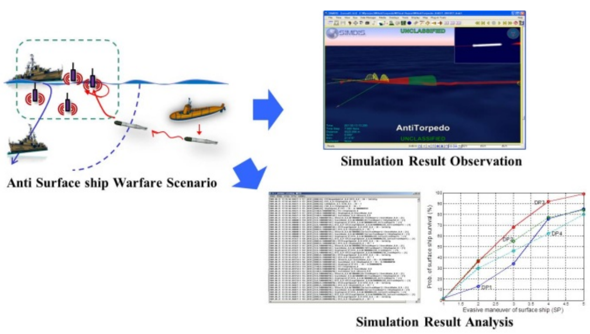


Figure 8: Simulation Result Analysis

In this I/F for simulation display, there are two steps. The first step is to register the platforms to be displayed, and the next step is to record simulation time and spatial information of the platform, such as position information, yaw, pitch and roll, and velocity. Therefore, the platform and environmental initialization for registration and platform input data are needed for simulation display.

Platform initialization consists of platform ID and platform name. For example, if we need one submarine, the object ID may be 1 and object name is "blue submarine." Environmental initialization consists of wind speed, sea flow, etc.; however, this information is optional. As illustrated in the platform, input data consists of object ID, time, position, orientation, and velocity, as illustrated in Figure 9.

Object Name	Object ID	Time	Spatial Information	Yaw	Pitch	Roll	Velocity Vector

Figure 9: Platform Input Data for Phase II I/F

In this paper, we use SIMDIS for simulation display. SIMDIS provides support for high-fidelity analysis and display of test and training mission data to a growing user base of nearly 8,000 users. This highly specialized visualization tool provides unique capability for two- and three-dimensional interactive data display and analysis. Figure 10 shows the I/F for simulation analysis using SIMDIS format. Platform initial information and input data are converted to a file format suitable for SIMDIS.

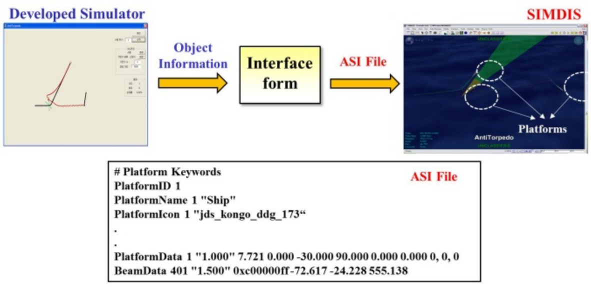


Figure 10: SIMDIS format for simulation analysis I/F

The I/F for simulation analysis also allows for a statistical evaluation of underwater warfare system effectiveness through Monte Carlo simulation. The feature of Monte Carlo simulation allows for random variations in certain platform parameters and simulated events to develop probabilistic assessments of system effectiveness. For example, the torpedo is launched randomly within the scenario guidelines and the reliability of the decoy is influenced by the normal random variable. These random variables are defined at the I/F for scenario identification.

4. CASE STUDY: COMPONENT OPERATION FOR SUBMARINE WARFARE

To demonstrate our contributions, this section illustrates two component operations for submarine warfare. We used the underwater warfare simulator developed by Seo and Song(2011), which is based on the DEVS formalism for underwater warfare.

4.1. Component operations for submarine warfare

A submarine performs various component operations such as anti-surface ship warfare (ASW), anti-submarine warfare (ASW), mine warfare (MW), surveillance warfare (SW), etc. In this paper, we consider two ASWs, and the brief scenario illustrated in Figure 11, as follows:

1. Enemies (submarine and surface ships) are approaching our submarine.
2. When the submarine detects the enemies during its barrier mission, it starts Target Motion Analysis (TMA) procedures to estimate the kinetic state, such as range, course, velocity, etc.
3. When enemies are located within attack range of the submarine, the submarine launches a torpedo toward the detected enemies.
4. After launching a torpedo, the submarine makes a detour for evasion.

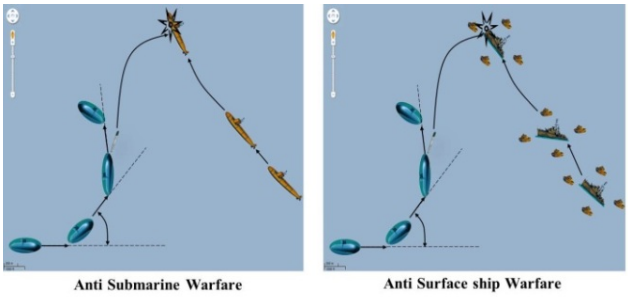


Figure 11: Scenarios for Two Component Operations

4.2. Experimental Results

Figure 12 through Figure 14 show experimental results applied to proposed I/Fs. Experiment 1 is for anti-submarine warfare; its objective is to use interface forms and check the results. Experiment 2 is for anti-surface ship warfare; and its objective is to evaluate various maneuver tactics.

Experiment 1 : Anti-submarine warfare

Figure 12 shows the I/F for scenario identification. The left side of Figure 12 shows the tactic script to vary the torpedo's maneuver pattern. The structure of the script is just like C-like code. When we decide the torpedo maneuver pattern from the tactic script, the underwater warfare simulator operates the maneuver pattern developed in the simulator. The right side of Figure 12 shows the I/F for scenario identification. The left side of Figure 12 shows platform parameters.

Figure 13 shows the simulation result of anti-submarine warfare. The I/F has a benefit, namely, the simulation of the various scenarios without modifying the model when tactics are changed, but with modifying scenario description I/F only. We can revise the tactic script at simulation run time.

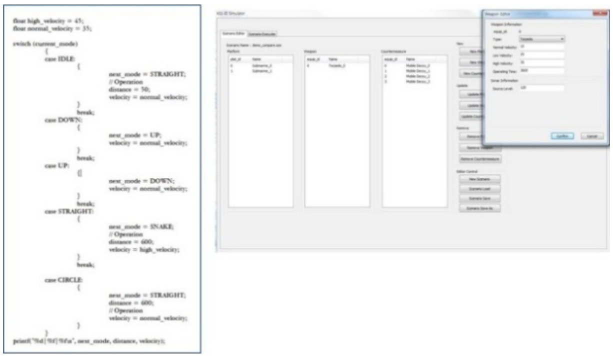


Figure 12: Scenario Identification I/F for Anti-submarine Warfare

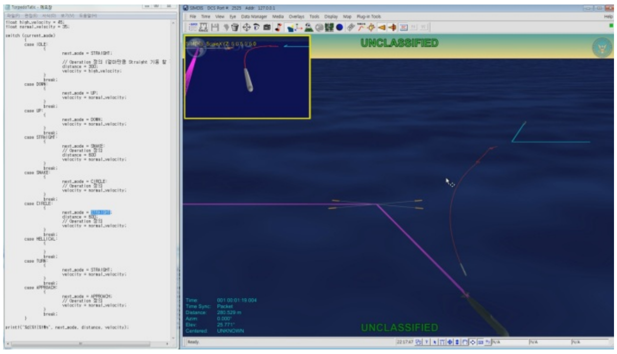


Figure 13: Simulation Result for Anti-submarine Warfare

Experiment 2 : Anti-surface ship warfare

The second experimental result depicted in Figure 14 shows the survival probability according to the search patterns of the torpedo. In this experiment, four different patterns, depicted in Table 1, are used. With the scenario description I/F, we can combine any maneuver patterns, which are designed in the simulator. The result shows that the combination of all three search patterns results in a higher probability of survival. The I/F enables users to assess alternative tactical deployments for torpedo maneuver patterns.

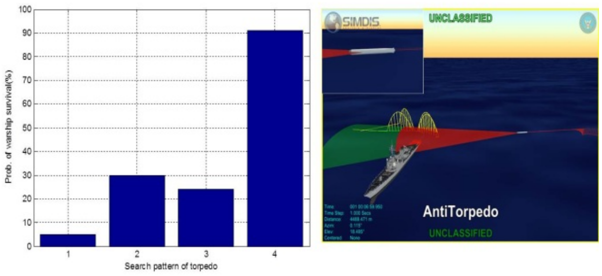


Figure 14: Simulation Result for Anti-Surface ship Warfare

Table 1: Maneuver Pattern Cases

Pattern	Description
1	Straight and snake maneuver patterns used
2	Straight and snake maneuver patterns used
3	Snake and circular maneuver patterns used
4	All of three patterns used

5. CONCLUSION

In this paper, we proposed implementation of Interface Forms (I/Fs) for underwater warfare simulator. I/Fs operate somewhat like experimental frames. Proposed I/Fs provide the developed simulator platform information and tactical information, and we observe the simulation result and analyze the result with the proposed I/F. Proposed I/F illustrates how to design an experimental frame for appropriate modeling objectives, and provides flexible experimental frames to provide insights about how various factors, such as tactics and the performance of underwater weapons, influence the MOEs of the system. The experimental result shows that we can test alternative tactics and that the behavior analysis was successful. Extension of the general concept should be considered in a future work.

ACKNOWLEDGEMENTS

This work was supported by Agency for Defense Development, Korea under the contract UD090024DD.

REFERENCES

Kim, T.G., Park, S.B., 1992. The DEVS formalism: Hierarchical modular systems specification in C++. *Proceedings of the European Simulation Multi-conference*, pp. 152-156, Jun., 1992, York, United Kingdom.

Office of Aerospace Studies, 2008. Analysis of Alternatives (AoA) Handbook. Available from: http://www.ndia.org

Seo, K.M., Song, H.S., Kwon, S.J., Kim, T.G., 2011. Measurement of Effectiveness for an Anti-Torpedo Combat System Using a Discrete Event System Specification-based Underwater Warfare Simulator. *The Journal of Defense Modeling and Simulation: Application, Methodology, Technology*. vol. 8, no. 3: pp. 157–171.

Sung, C.H., Moon, I.C., Kim, T.G., 2010. Collaborative Work in Domain-Specific Discrete Event Simulation Software Development: Fleet Anti-air Defense Simulation Software. *Proceedings of 2010 Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pp. 160-165, Larisa, Greece.

U.S. Naval Research Laboratory, 2006. *SIMDIS User's Manual*. Available from: https://simdis.nrl.navy.mil.

Zeigler, B.P., Praehofer, H., Kim, T.G., 2000, Theory of Modeling and Simulation. United State: Academic Press.

EXCEL-BASED ANALYSIS AND DYNAMISATION OF PROBABILITIES FOR LOGISTICS PLANNING

Markus KLUG[(a)], Andreas ALEXA[(b)]

[(a)] University of Applied Sciences Technikum Vienna; Department for Advanced Technologies
[(b)]Austrian National Defence Academy; Institute for Higher Military Command and Control

[(a)]markus.klug@technikum-wien.at, [(b)]andreas.alexa@bmlvs.gv.at

ABSTRACT
The existing parameters for the planning of logistics support as given in the guidelines for logistics planning at Brigade level and similar guidelines at Battalion level include a huge amount of (fixed) probabilities. These probabilities were derived from historical data. Because of this, these parameters are questioned during war-games, due to the fact that logistics officers' own experiences show different results in certain real-life cases. This paper deals with these parameters and how Excel-based spreadsheets are used to explain the different values, supporting them once the statistical laws of large numbers have been achieved. In addition, it shows how such spreadsheets can be used for situations where a small number (below the afore-mentioned laws) needs to be investigated for logistics purposes.

Keywords: Logistics Planning, Excel-based Probabilities, Law of Large Numbers, Spreadsheet-based Simulation

1. INTRODUCTION

Logistics planning in the Austrian and various other European armed forces, as well as in the US, are based on the Day of Supply (DOS) approach. Certain figures are predetermined for the appropriate planning of different units concerning daily demands with respect to water, food, ammunition, fuel and various other parameters. (Austrian Ministry of Defence and Sports, 2010)

These figures were derived from historical conflicts, using operations research approaches to shape future demands (Shrader 2006; Shrader 2008; Shrader 2009), and are used in the education and training of supply sergeants. The parameters are definite numbers, albeit changed as appropriate by multiplicative factors according to terrain, weather conditions and mission intensity.

For the US Navy, an accurate planning of logistics was of great importance even in the early 1970s, when logistical implications started to be implemented in computer-based war-games. (Perla, 1990)

However, this DOS approach is already being questioned in the US itself, where, in 2008, an article suggested changing to a demand-oriented "Sense and Respond Logistics" approach, reducing the stockpile from 60 days for the Operation Desert Storm in 1991 to 5- 7 days for the Operation Iraqi Freedom in 2003. (Hammond 2008). This approach may alter the existing view on logistics, as given, for example, in the NATO Logistics Handbook and subsequent related documents. (NATO 2007). Nevertheless, the DOS-based logistics planning approaches are still taught to Austrian as well as foreign officers participating in the appropriate courses.

On the other hand, the above-mentioned Austrian guidelines for logistics planning contain a warning notice "Just applying the guidelines without taking the current situation into account will lead to results that are not applicable to the supply officer". The question still remains as to how these responsible officers should become aware of inappropriate conditions, if these general parameters no longer apply.

Austria faces the challenging task of being the "Logistics Lead Nation" for the European Battle Group during the second half of 2012. For a common multinational planning and information interchange, the DOS-based approach is still the basis for all calculations.

With respect to the current different international missions of the Austrian Armed Forces, there are several challenges to proper supply planning still to be overcome, which are described in more detail in the next section.

2. CHALLENGES TO LOGISTICS PLANNING

Among the various DOS parameters which calculate the average demand for different goods several parameters exist for risks of failure or damage, including to vehicles, which will now be taken as examples. Two main aspects will be investigated in more detail, summed up in the well-known bathtub curve derived from reliability engineering. (Matyas 2010)

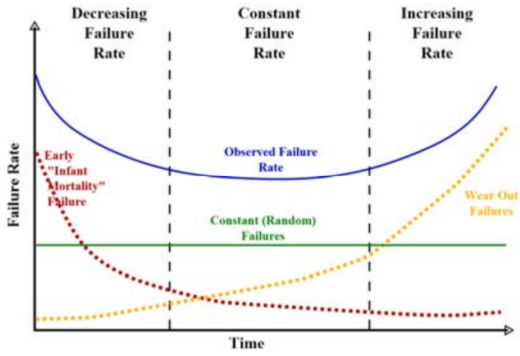


Figure 1: Bathtub curve (Wikipedia, 2009)

Especially the figures given for the failure of systems suggest that this rate remains constant during the whole lifetime of these systems, which does not hold true in different cases.

2.1. An example of the increased risks of early failure

In 2008, the Austrian Armed Forces created a "Special Operations Vehicle" for the EUFOR Chad Mission. This development was based on the existing cross-country vehicle platform Puch G 290 (the Austrian brand of the Mercedes G series) called "Sandviper" and was carried out by the Austrian Armed Forces internal agency for arms and defence technology.

Due to the fact that these significant changes were implemented purely for this mission, long-term experiences with the failure of operations could not be gained when this vehicle was put to daily use in Africa. Even taking into consideration the fact that the agency was reducing the technical parts to the absolute minimum, it is important to apply an increased early failure rate, also because no experiences with this vehicle have been made in a desert zone.

Figure 2: The Puch G 290/LP "Sandviper" (Austrian Armed Forces Photograph 2011)

2.2. An example of the increased risks of wear-out failure

On the other hand, Austria is still using the Aérospatiale Alouette III helicopter, which was first introduced in 1960 and has already been withdrawn in various other states (France 2004, Ireland 2007, Switzerland 2010, among others).

This helicopter, which has been in service in Austria since 1967, is also part of the Austrian missions to Bosnia, as well as sometimes to Kosovo, supporting the EUFOR and KFOR, respectively. Due to the age of the equipment, normal parameters for the failure of parts are not likely to apply any more.

Figure 3: The Aérospatiale Alouette III (Austrian Armed Forces Photograph 2011)

2.3. The problem of limited amounts not fulfilling the law of large numbers

As already mentioned, a couple of vehicles are used in daily operation outside Austria. This, as well as specific equipment available only in limited amounts (e.g. armoured recovery vehicles), alters the pure algorithmic approach for calculating the amount of spare parts and the average number of vehicles available, as well as for appropriate planning. This change is based on the statistical law of large numbers, which is not fulfilled in various cases, especially concerning the technical equipment in use.

3. THE MATHEMATICAL APPROACH TO CHALLENGES IN LOGISTICS PLANNING

Especially with respect to the given probabilities, an algorithmic calculation of the amount of failures and the damage to vehicles divided into two categories (normal and armoured vehicles) under certain operational conditions does not correspond to reality (computing not purely integer numbers) or to the failures observed.

3.1. The Bernoulli distribution and a first spreadsheet model

The probabilities of failure or damage during an operation are given as a percentage. For example, a 6% probability needs to be taken into account for armoured vehicles during a one-day march under normal conditions. Assuming an armoured infantry with 14 infantry fighting vehicles (IFV), the algebraic result of calculating the number of vehicles affected at the end of the day is 0.84.

If two vehicles fail during the day, the calculation method that has already been called into question and the underlying parameters not providing integer numbers can be completely rejected.

To illustrate what is meant by a mathematical Bernoulli distribution (i.e. throwing a coin, falling on one side with a given probability), a first basic Excel sheet can be created to start a so-called “spreadsheet-based simulation approach”, according to (Schriber 2009).

Table 1: Random determination of failures according to the given probability

IFV No.	Random Probability	in Operation / Failure
1	0.851195351	in Operation
2	0.556390769	in Operation
3	0.054002108	FAILURE
4	0.028278218	FAILURE
5	0.672174937	in Operation
6	0.363501829	in Operation
7	0.344775482	in Operation
8	0.234264572	in Operation
9	0.237118103	in Operation
10	0.029055798	FAILURE
11	0.512215943	in Operation
12	0.032268999	FAILURE
13	0.060050503	in Operation
14	0.737855443	in Operation
	Number of failures:	*4*

Table 1 shows the vehicle number in the left-hand column, a randomly generated number between 0 and 1 in the central column and the assessment of the parameter in the right-hand column. Every randomly generated number less than or equal to 0.06 is assumed to be a failure at the end of the day. The number of failures is calculated at the end of the table. A new scenario is generated by pressing the “F9” button on the keyboard.

In most cases, the number of failures is 1, sometimes decreasing to 0 and, more rarely, increasing to 2 or even 3 and rarely to 4. A certain sequence of 20 computed scenarios counting the number of failures may be the following:

2, 0, 1, 0, 2, 1, 0, 1, 1, 0, 0, 1, 1, 2, 1, 1, 3, 0, 2, 0

which means:

- 7 times no failure
- 8 times one failure
- 4 times two failures
- 1 time three failures

It then becomes clear that simply an algebraic calculation of given probabilities using the given small amount of vehicles taken into account in the case of small numbers is not sufficient and that a failure rate of up to three vehicles does not necessarily mean that the given parameter is wrong, even though the percentage of failures in this case rises to 21-43%.

3.2. The Poisson distribution and the confidence interval

Given this result, it becomes clear that uncertainty as regards planning first increases. The question “How many vehicles do I need to take into account then?” is soon raised.

This is how the Poisson distribution – and in this particular case subsequently also a higher time resolution – is introduced.

Given the parameter of 0.84 failures per day (viewed statistically), the Poisson distribution can be calculated, again using Excel:

Table 2: Calculation of (cumulative) probabilities for vehicle failures

Number of IFV vehicles	Probability, that this number of vehicles fails	Probability, that maximum this number of vehicles fails
0	43.17%	43.17%
1	36.26%	79.43%
2	15.23%	94.67%
3	4.26%	98.93%
4	0.90%	99.83%
5	0.15%	99.98%
6	0.02%	100.00%

Here, it can easily be seen that the results gained before have been verified. With a 95% or 97.5% confidence, up to three vehicles need to be repaired. On the other hand, in terms of fuel support, all the vehicles need to be taken into account.

This result becomes even more interesting when it is discretized on a two-hourly basis. With respect to the new time intervals the failure probability per two hours is 0.07 (=0.84/12). The corresponding table is as follows:

Table 3: Calculation of (cumulative) probabilities for vehicle failures on a 2h basis

Number of IFV vehicles	Probability, that this number of vehicles fails	Probability, that maximum this number of vehicles fails
0	93.24%	93.24%
1	6.53%	99.77%
2	0.23%	99.99%
3	0.01%	100.00%

Assuming a certain reliability (in this case, 95%, 97.5% or 99% leads to the same results), the loss of one vehicle needs to be taken into account.

The following graph shows the computed borderlines for the development of the number of vehicles:

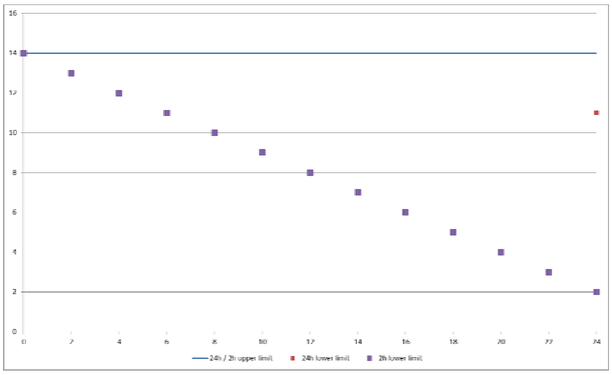


Figure 4: Illustration of the development of the number of vehicles over a 24h approach vs. a 2h approach

3.3. A spreadsheet-based simulation approach

Nevertheless, to come back to the Bernoulli distribution, a spreadsheet-based simulation approach can now be integrated.

A scenario is computed, where, on the given two-hourly basis, the number of vehicles during the day is calculated. Again, a row with random variates is taken and assessed against the probability:

Table 4: Preparation of the spreadsheet approach

Random Variate	Failure	Number of vehicles remaining
0.202956394	0	14
0.722896618	0	14
0.48978746	0	14
0.068235188	1	13
0.218636315	0	13
0.013813846	1	12
0.898089867	0	12
0.170398473	0	12
0.964601858	0	12
0.018793236	1	11
0.101832165	0	11
0.343930076	0	11
0.491970795	0	11

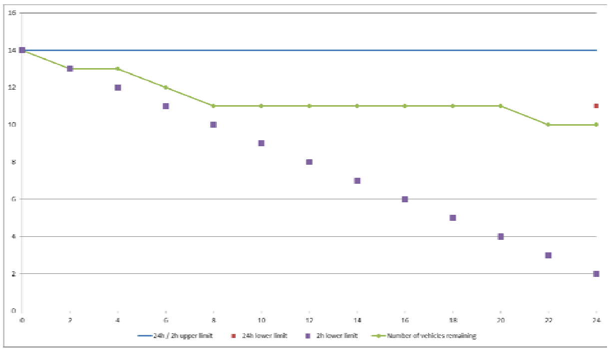


Figure 5: Development of the number of vehicles - graphical illustration

Again, by pressing F9, different scenarios are computed, which in general result in somewhere between 11 and 14 vehicles remaining.

3.4. Destruction / failure analysis of a large number of vehicles

With respect to the large number of vehicles used, as trained at Brigade level, this approach can be scaled up to a larger number of armoured vehicles.

A probability of 20% destruction was assumed for combat vehicles. Taking 394 combat vehicles into account, the application of the Poisson distribution gives the following picture:

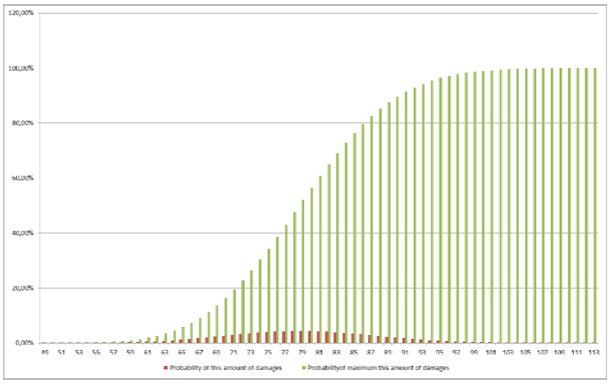


Figure 6: Increasing the number of vehicles, motivating the density and distribution function

This purely Microsoft Excel-based analysis implies that, in contrast to the given estimate of around 79 destroyed vehicles, the amount of remaining vehicles may be somewhere between 62 and 97, with a 95% probability for the confidence interval, and between 51 and 110 vehicles, assuming a 99% probability.

These new figures, which differ from the 79 destroyed combat vehicles (the probability of achieving this number is just 4.5%), are the basis for further risk analysis at different levels.

From the supply side, it should be assumed that only a small amount of vehicles has been lost and therefore personnel, material and fuel needs to be provided, ending up with just 14.5% in terms of losses.

On the other hand, further attack or defence capabilities need to assume that the maximum amount will not be available after one day of combat. In this case, one is faced with a 5.6% increase in effort for dressing stations, the search and rescue tasks of personnel and, finally, the salvaging and recovery of materials.

Due to the fact that only integer numbers can realistically be used, the same parameters are used for an hourly-based analysis of destruction or other failures (formerly this was done on a daily basis). For this, the given parameters are converted into average failures per hour and are applied in a similar manner to that already shown.

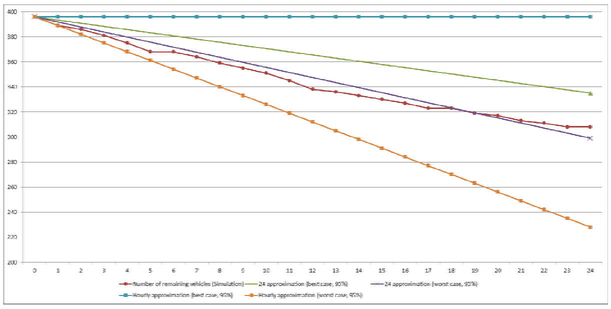


Figure 7: Hourly-based destruction /failure analysis, again including a simulated scenario

While the 24-hour approximation is still illustrated with the straight lines ending up with an hourly-based analysis, this shows a realistic development.

In this case, the red line with the dots illustrates – so to say – an Excel-based simulation of destruction, nearly always ending up within the limits of the daily estimates, but nevertheless often crossing the imaginary borders during these 24 hours.

This clearly illustrates how probabilities are balanced and try to converge to the algebraic mean, just as the strong law of large numbers is mathematically proven. In addition, it leads to a better understanding of how the Austrian war-game simulation environment “FüSim” internally may realise this kind of randomness, bringing staff officers closer to topics of random variates, from the simulation point of view.

4. PLANNING IMPACT

There are several impacts on planning which are briefly described below.

4.1. Saving and rescuing

During a mission, a certain amount of additional consequences need to be taken into account. One of these is to rescue a minimum of 50% of the heavily armoured vehicles for future repair and use. The hourly graph shows the impact of such additional requirements on the planning of the necessary equipment.

4.1. Calculation of the necessary supply and available forces

Based on the 24-hour expectations, the lower and upper boundaries now illustrate for how many forces the resupply of goods needs to be provided, assuming the best case scenario that a maximum amount has survived.

4.2. Overestimation due to uncertainty

Already in this case, an overestimation is likely to be reduced. Only this amount, which is necessary even in the worst case, will be provided for a mission so that further equipment and material exceeding the maximum necessary amount can be avoided.

The effect gained is to reduce the necessary amount to be transported, stored, observed and finally shipped back to the original destination.

4.3. Assessment of planning parameters

Finally, the graphical results can also be applied to assess the real figures, which are provided from time to time. Especially in the event that the lower (hourly) limits are overstepped, planning parameters need to be questioned immediately, because they do not hold true anymore.

5. SUMMARY AND OUTLOOK

The importance of this kind of analysis is also borne out by the new challenges of irregular warfare, where existing parameters from former wars between armed forces cannot be applied any more. It provides a first approach to risk analysis, when developments from the real world start to differ from planning calculations.

This kind of mathematical analysis is becoming increasingly important in the training of Austrian staff officers. Therefore, a dedicated Operations Research Course including this kind of analysis will be provided for the education of Austrian staff officers from autumn 2011 on.

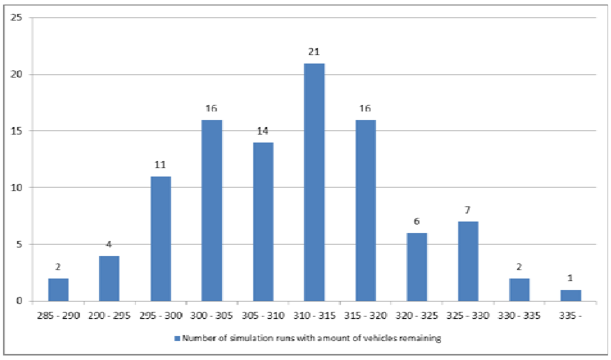


Figure 8: Excel-based histogram of 100 scenarios basing on the model of chapter 3.4

If one looks at this figure, it becomes clear that the average failures are the most likely, but it also gives a clear picture of how this may vary in such a case. Based on this figure, as well as on Figure 6, it becomes easier to introduce statistical distribution and density functions.

This approach is also the basis for using further software such as @RISK, which provides a certain amount of distribution functions that can already be approximated by a small Excel-based macro.

ACKNOWLEDGMENTS

Markus Klug wishes to thank the Austrian Armed Forces for their support of this paper.

REFERENCES

Austrian Ministry of Defence and Sports. 2010. *Regulation for internal services of the Austrian Armed Forces: "Material Maintenance"*. Austrian Ministry of Defence and Sports. Vienna.

Hammond, M. F. 2008. *Sense and Respond: Military Logistics in a Global Security Environment. Army Logistician*. Available from:

http://www.almc.army.mil/alog/issues/SepOct08/sense_response1.html. [Accessed 9 July 2011]

Matyas, K. 2010. *Instandhaltungslogistik*. Vienna. Carl Hanser Verlag.

NATO. 2007. *NATO Logistics Handbook*. NATO. Available from: http://www.nato.int/docu/logi-en/logistics_hndbk_2007-en.pdf. [Accessed 9 July 2011]

Perla, P. P. 1990. *The art of wargaming: a guide for professionals and hobbyists*. Annapolis. Naval Institute Press.

Schriber, T. J. 2009. Simulation for the Masses: Spreadsheet-based Monte Carlo Simulation. *Proceedings of the 2009 Winter Simulation Conference*. 1-11. 13.–16.12.2009 Austin, TX.

Shrader, Charles R. 2006. *History of Operations Research in the United States Army; Volume I: 1942-1962*. Washington: Govt Printing Office.

Shrader, Charles R. 2008. *History of Operations Research in the United States Army; Volume II: 1961-1973*. Washington: Govt Printing Office.

Shrader, Charles R. 2009. *History of Operations Research in the United States Army; Volume III: 1973-1995*. Washington: Govt Printing Office.

U.S. Army. 2009. *Bathtub curve*. Wikipedia. Available from: http://upload.wikimedia.org/wikipedia/commons/7/78/Bathtub_curve.svg. [Accessed 9 July 2011]

AUTHORS BIOGRAPHY

Markus Klug is a lecturer on logistics and simulation topics at the University of Applied Sciences Technikum Wien. In addition, he is a member of the Austrian Armed Forces Expert Staff belonging to the quartermaster department. He received his MSc equivalent degree in Technical Mathematics from the Technical University Vienna. Besides this, he served as a board member for the German-speaking simulation society, ASIM, from 2003 to 2007. His professional experience in modelling and simulation includes large-scale simulation projects on various topics including supply chain and simulation-based production optimization, as well as the simulation of military problems carried out at the Austrian Research Centers, where he has been head of the workgroup “Process Optimization” for several years. Currently, his research interests cover military operations research, GIS integration into simulation, simulation-based and simulation-focused e-learning, as well as cloud-computing approaches.

Andreas Alexa is a lecturer and researcher on national and multinational military logistics at the National Defence Academy of the Austrian Armed Forces in Vienna. During his military service in the Austrian Armed Forces he served on various logistics assignments. From 2007 to 2010, he attended the Joint Command and General Staff Course. He received his Master’s degree in Security and Defence Policy from the University of Vienna.

A SIMULATION STUDY OF C4I COMMUNICATIONS NETWORK UNDER CYBER ATTACKS

Agatino Mursia[(a)], Lucio Ganga[(a)],
Alessandro Leonardi[(b)], Giacomo Morabito[(b)], Corrado Rametta[(b)], Lloyd Wihl[(c)]

[(a)] Selex Elsag Spa, Italy
[(b)] Dipartimento di Ingegneria Elettrica, Elettronica ed Informatica, University of Catania, Italy
[(c)] Scalable Network Technologies, Inc. USA

[(a)] Name.Surname@selex-comms.com, [(b)]Name.Surname@dieei.unict.it, [(c)]lwihl@scalable-networks.com

ABSTRACT
The use of wireless communications in today's tactical networks is a necessity but at the same time increases the vulnerability to cyber attacks. It is essential to consider the effects of these attacks during network design and operation. Network simulation software, which employs models of real tactical radios, and hardware in the loop functionalities, can be very useful to evaluate the effects of such attacks on the system performance. In this paper, we present the results of our study aimed at investigating the behavior of wireless communication networks under some common cyber attacks and we also consider the effect of some countermeasures in order to minimize the effects caused by the attacks.

Keywords: simulations, EXata/Cyber, human in the loop, performance, cyber warfare, networks

1. INTRODUCTION

The use of wireless communications in tactical scenarios has gained a primary role in the modern approaches of patrol and fight (Maseng and Nissen 2008).
The units involved in tactical warfighting are more and more often equipped with electronic devices able to exchange, through a data flow (continuous or burst), sensitive contents with other units located in the same geographical area and/or with higher hierarchical levels. The adoption of these new technologies allows a strong interaction among the different components involved in the operation and improves the coordination and efficacy of the action in the area of interest.
However, the use of these technologies implies, necessarily, an increase of the risks coming from cyber attacks towards the network and the supported data flows.
Consequently, it becomes mandatory, during the design of network architectures and the selection of proper communication protocols, to take into account the above issues and dispose the required procedures to counteract them (Lipson 2002, Di Pietro 2002).
One of the most efficient ways to achieve the above objectives is the use of simulation software. This software can be employed both in the design phase of networks and equipment, and in the planning of operations. The flexibility, modularity and performance prediction make simulation software essential to validate the architectural choices related to the networks and equipment to be employed in the different scenarios.
The goal of this paper is to discuss the results of our study aimed at investigating, through different simulation scenarios, the behavior of wireless communication networks under some particular cyber attacks. The scenarios under investigation are characterized by teams and/or mobile units operating in a specified area and using communication facilities having Mobile Ad hoc NETworks (MANET) capabilities. The considered scenarios include models representing the operational functions of some communication radio devices working in extra-urban areas.
These models are derived from the real tactical radio devices and integrate software models that implement protocols related to levels 1, 2 and 3 of the ISO/OSI protocol stack, created "ad hoc" to counteract and/or minimize the damage caused by the cyber attacks.
Simulation environments are deployed by using specialized software simulators. In this paper we use EXata/Cyber®, (Scalable Network Technologies 2011), in order to simulate the communication network architecture, the radio devices employed, and the cyber attacks and defenses.
Moreover, EXata/Cyber® provides hardware-in-the-loop (HITL) functionalities which we exploit to connect systems running real applications in order to obtain a more realistic behavior of the traffic sources.
In particular, we focus on command, control, communications, computers, and (military) intelligence applications (C4I) which play a key role in tactical and

military missions. Data flows generated by hardware-in-the-loop devices go across the simulated network in the same way they cross the real one, allowing to estimate the effects of cyber attacks on the effectiveness and reliability of the data transfer among sources and destinations, (Shen 2009).
Moreover, we describe and provide the performance of some countermeasures (network architectures, network protocols, etc.) with the aim of reducing the effects of the cyber attacks taken into account in our study.
The rest of this paper is organized as follows. In Section 2 we will describe the system model. In Section 3 we will describe the elements which compose the system architecture. In Section 4 we will describe the considered scenarios and in Section 5 we will show the results obtained. Finally, in Section 6 we will draw our conclusions.

2. SYSTEM MODEL

We consider a MANET consisting of N mobile wireless nodes deployed in an extra-urban area. In order to protect communications, all nodes use the following security procedures:

- MAC layer protection: where access to the MAC layer is protected by specific mechanism (i.e., crypting or frequency hopping).
- Network/Application layer encryption: this ensures that adversary cannot read messages exchanged at the application layer.

Accordingly, it is difficult for the adversaries to attack the network, unless they have gained control over a certain number, say M (with $M>0$) nodes.
In fact, if this is the case, the adversaries can perform different types of attacks depending on the protocol layers that they are able to modify.
In particular,

- if they can introduce modifications at the physical/link layer, they can perform jamming attacks. In particular, a malicious node transmits continuously over a channel with sufficiently high power such that all other nodes in the vicinity are unable to send or receive any other signals. This creates a reduction in the network throughput and a decrease in the battery duration.
- If they can introduce modifications at the network layer, they can perform routing attacks. In this case the adversaries create inconsistencies in the routing tables or try to break end-to-end paths between the pairs source/destination.
- If they can introduce modifications at the application layer, they can perform Denial-of-Service (DoS). In this case the nodes controlled by the adversaries generate a large number of service requests for a certain number of nodes so as to cause interruption of the service they provide (note that in ad hoc networks a service provided by all nodes is relaying traffic).

In this paper we study the impact of such attacks on the network performance. More specifically we consider the impact of the knowledge of the adversaries about the technical solutions utilized by the network on the attack effectiveness. For example, if the adversaries know the routing protocol utilized by the network, then they can attack the node that is the most critical for the current topology. If the adversaries know the characteristics of the MAC protocol utilized, then they can perform a more effective jamming attack.

3. SYSTEM ARCHITECTURE

In order to study the impact of the previously described cyber attacks on the system performance, we have considered a testbed which uses both simulated and real devices.

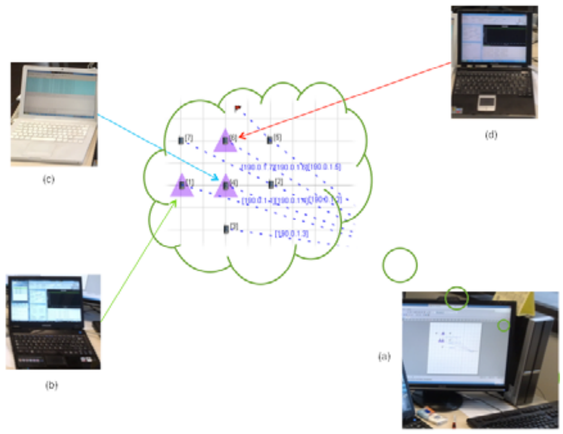


Figure 1 - System Architecture

The system architecture, shown in Figure 1, is composed of the following parts:

3.1. Emulation platform

The emulation platform is realized using a computer desktop running Windows OS and the EXata/Cyber® software suite (Figure 1.(a)). In the emulation platform we can choose the number of nodes, N, the physical parameters such as the radio type, frequency, antennas type, etc., the MAC layer, the routing protocol if needed, and the nodes' movement pattern.

3.2. Traffic source

The traffic source, which represents the C4I communication sender, has be modeled in two ways, according to the scenario of interest. The first is using an EXata/Cyber® virtual node, the second is using a notebook running Windows OS together with the *connection manager* tool, and the *iperf* tool running in client mode (Figure 1.(b)). The iperf tool emulates UDP or TCP traffic sent by the node. This real notebook is then mapped inside the EXata/Cyber® emulator as a mobile or fixed node.

3.3. Traffic destination

The traffic destination can be realized in two ways, as a synthetic or emulated node, similarly to the traffic source. Concerning the second way, the *iperf* tool is used in server mode (Figure 1.(c)) and this real notebook will then be mapped inside EXata/Cyber® to a mobile or fixed virtual node.

3.4. Malicious node

The malicious node can be realized either as a virtual node inside EXata/Cyber® or as a notebook running Windows OS with the EXata Connection Manager tool which will be mapped to a virtual node in EXata/Cyber® (Figure 1.(d)). The choice of configuration will be related to the scenario of interest.

4. SCENARIOS

We have investigated three different network scenarios. All three scenarios use radio models, with a channel capacity of 2 Mbit/s and a radio range of 250 m. In scenario 1, we have simulated a DoS attack and we have considered the effect of a jammer node against the node which performs DoS. In scenario 2, we have simulated the use of directional antennas in order to reduce the 'eavesdropping' effect. Finally, in scenario 3, we have compared the performance of some common wireless ad hoc routing protocols when there are some jammer nodes in the area.

4.1. Scenario 1 - DoS attack

This scenario is characterized by N = 7. The location of the nodes in the field of interest is shown in Figure 2. We consider a malicious node (M = 1), identified in Figure 2 as node 6, performing a DoS attack against node 4 which represents the traffic destination of a communication initiated by node 1. Both the data communication (node 1 - node 4) and the DoS attack are realized using the *iperf* tool, (Iperf, 2001), and are characterized by the following parameters: size of UDP packets equal to 512 bytes, bit rate equal to 2 Mbit/s, duration 300 s. The DoS attack lasts for 240 s starting from time t = 60 s. The considered countermeasure is realized by identifying the source of the DoS, i.e., node 6, and using a jamming attack against this node. The node which realizes this countermeasure is node 5 which is a virtual node simulated inside EXata/Cyber®. In particular, node 5 moves in the neighborhood of node 6 and once arrived inside its coverage area it starts the jamming attack. The jamming attack is implemented through the HITL command interface of Exata/Cyber®. The jamming attack consists in transmitting continuously over a channel with sufficiently high power, such that all other node in the vicinity of the jammer and communicating on the same channel will find the channel busy, and therefore, will not be able to transmit or receive any other communication.

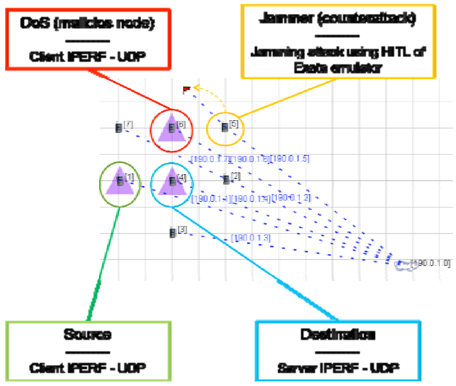


Figure 2 - Scenario 1

4.2. Scenario 2 - Eavesdropping

This scenario is characterized by N = 16. The location of the nodes in the field of interest is shown in Figure 3. We consider a malicious node (M = 1), identified in Figure 3 as node 16, which is able to *sniff* the data traffic flowing in the network. We consider data communication which is both encrypted and unencrypted. In the first case, the malicious node can only locate the nodes involved in the communication; in the second case, the malicious node can also decode the data packets. The *sniffing* procedure is performed using the "eavesdropping" capability implemented in EXata/Cyber® which is modeled as the node's MAC layer operating in promiscuous mode, and enabled to promiscuously listen to nearby wireless communications. Data communication is realized through four CBR data flows, and is characterized by the following parameters: size of UDP packets equal to 512 bytes, bit rate equal to 41 kbit/s, duration 60 s. The eavesdropping attack starts at the beginning of the simulation (time t = 0 s). The malicious node moves inside the simulation field along an established path, as shown in Figure 3. Because it is not possible to identify the eavesdropper node (which is implemented as a passive node) and consequently reducing its functionality (for example using jamming), the considered countermeasure is to use directional antennas both in the source/destination nodes and relayers involved in the communication. The directional antennas are modeled in EXata/Cyber® using the *switched-beam antenna* which is a special type of patterned antenna having different gains in different directions. The switched-beam antenna utilizes multiple antenna patterns and switches the pattern according to the direction of arrival or transmission.

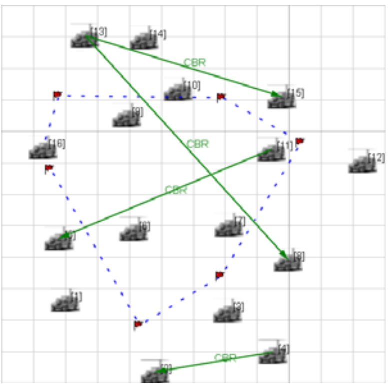


Figure 3 - Scenario 2

4.3. Scenario 3

This scenario is characterized by N = 50 nodes. The location of the nodes in the field of interest is shown in Figure 4. In particular, they are located in a square grid pattern (distance between nodes equal to 200 m) in order to have a better knowledge of the routing protocol's capability to change the path when the attack starts. We consider two malicious nodes (M= 2) moving inside the area using a random waypoint mobility model with the following parameters: range of velocity: 0 - 108 km/h and pause time equal to 20 s. The malicious nodes perform a jamming attack against all the relayer nodes in their neighborhood. The coverage range of the jammer is equal to 125 m. The jamming attack is modeled using the EXata/Cyber® capabilities as in scenario 1. Data communication is realized through five CBR data flows, and is characterized by the following parameters: size of UDP packets equal to 512 bytes, bit rate equal to 102 kbit/s, duration 300 s. The jamming attack starts at the beginning of the simulation (time t = 0 s). The routing protocols considered are: AODV (Perkins 1999, Johnson 1994) and ANODR (Kong 2003). In Section 5.3, we will compare the performance of the routing protocols in terms of robustness to find an alternative path.

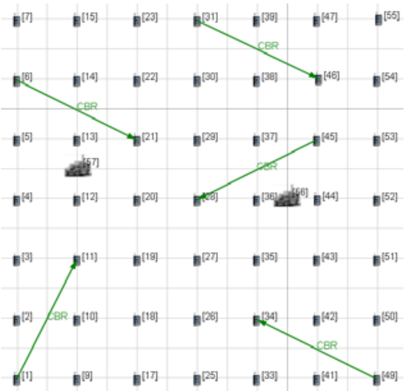


Figure 4 - Scenario 3

5. RESULTS

In this section we describe the impact of the considered attacks and the benefits of the chosen countermeasures on the system performance.

5.1. Results of Scenario 1

In scenario 1 we evaluated both the throughput and the jitter values related to node 4, because these metrics show both the impact of the attack and the efficacy of the countermeasure. Looking at Figure 5 it is possible to note that from t = 0 s to t = 60 s, the number of packets received from node 1 and the related jitter are almost constant. In particular, the value of the received bits corresponds to the maximum value achievable in the communication. Once the DoS attack starts, at time t = 60 s, we observe that the value of throughput becomes unstable, fluctuating in a wide range and sometime reaching the zero value. The value of the jitter increases accordingly. This trend lasts until t = 140 s, which is the time needed for the node 5 to identify the malicious node 6 and starting a jamming attack against it. As a result, node 6 is inhibited to transmit data packets to node 4, and the throughput value again reaches the maximum value such as at the beginning of the test.

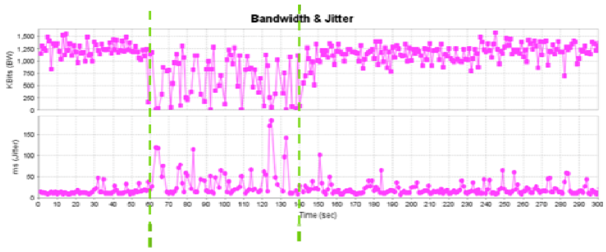


Figure 5 - Results of Scenario 1

5.2. Results of Scenario 2

In scenario 2 we evaluated the differences in the number of packets *sniffed* by node 16 (i.e., packets flowing in the network and received by node 16), when using omnidirectional antennas, and when using directional ones. From Table 1, we note that, as expected, when using the directional antennas, the number of data packets sniffed by the malicious node is reduced when compared to the case when the nodes use omnidirectional ones. This is because, with directional antennas, it is more difficult for the malicious node to be in the radio coverage of the nodes involved in the communication.

	With omnidirectional antennas	With directional antennas
Number of sniffed packets	1476	902
Percentage of sniffed packets over the total number of packets flowing in the network	30 %	18 %

Table 1 - Results of Scenario 2

5.3. Results of Scenario 3

In this scenario, we evaluated the performance in terms of throughput (percentage of received packets), delay and jitter, of the following routing protocols: AODV and ANODR. During the simulation runs we observed that the AODV protocol is the most reactive one. In fact, when the jamming attack is active, the AODV protocol succeeds almost always to find an alternative path towards the destination. This is the reason for the higher percentage of received packets when compared with ANODR. However, if we look at the values of delay and jitter, we note that ANODR shows better performance. This is due to the fact that the values of delay and jitter for the ANODR are calculated considering only the packets which reach the destination when the jammer is inactive, and consequently these packets show less delay because they follow the route with the minimum number of hops.

	AODV	ANODR
Percentage of received packets	88.53 %	67 %
End-to-end delay [s]	0.504	0.107
Jitter [s]	0.012	0.003

Table 2 - Results of Scenario 3

6. CONCLUSIONS

In this paper we used a high fidelity software emulation of networks that includes cyber warfare models, EXata/Cyber, to evaluate the effects of some common cyber attacks on the effectiveness and reliability of the data transfer between sources and destinations in a wireless MANET. We also investigated the effect of some countermeasures on the system performance. Results from the software emulation were in accordance with expectations.

ACKNOWLEDGMENTS

We would like to thank Riccardo Rovelli and Amanda Zanotti of Antycip Simulation for providing the EXata/Cyber® emulation software.

REFERENCES

Di Pietro, R., Me, G., 2002. Military secure communications over public cellular network infrastructure. *Proceedings of MILCOM 2002*.

Iperf, 2001. Available from: http://sourceforge.net/projects/iperf/

Johnson D. B., 1994. Routing in Ad Hoc Networks of Mobile Hosts. *Proceedings of the Workshop on Mobile Computing Systems and Applications*.

Lipson, H. F., 2002. Tracking and Tracing Cyber-Attacks: Technical Challenges and Global Policy Issues. Techical Report. Available from: http://www.cert.org/archive/pdf/02sr009.pdf

MAK, 2011. Available from: http://www.mak.com

Maseng T., Nissen, C.A., 2008. Network centric military communications. *IEEE Communications Magazine*, 46(11), 77-78.

Perkins C. E., Royer E. M., 1999, Ad-hoc on demand distance vector routing. *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99).*

Scalable Network Technologies, 2011. Available from: http://www.scalable-networks. com

Shen, J., Gan, Z., Gang, C., 2009. The Study of Military Simulative Training Based on "Network in the Loop". *Proceedings of Computer Network and Multimedia Technology.*

Kong J., Hong X., 2003. ANODR: anonymous on demand routing with untraceable routes for mobile ad-hoc networks. *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing (MobiHoc '03).*

Wihl L., Varshney M., Kong J., 2010. Introducing a Cyber Warfare Communications Effects Model to Synthetic Environments. *Proceedings of the Inter-service / Industry Training, Simulation and Education (I/ITSEC) Conference 2010.*

Bruzzone A.G., Cantice G., Morabito G., Mursia A., Sebastiani M., Tremori A. (2009) "CGF for NATO NEC C2 Maturity Model (N2C2M2) Evaluation", *Proceedings of I/ITSEC2009, Orlando, November 30-December 4*

Tremori A. (2010) "Research activities about usage of Intelligent Agents and Computer Generated Forces (IA-CGF) for Asymmetric and Non Conventional Operations", *Invited Speech, NATO CAX Forum2010, Munich Germany, August 30/September 2*

Bruzzone A.G. (2010) "CGF & Data Fusion for Simulating Harbor Protection & Asymmetric Marine Scenarios", *Proceedings of SIM&SEA2010, La Spezia, June 8*

Alessandro Leonardi received the Laurea degree in Electrical Engineering and the Ph.D. degree in Computer and Telecommunications Engineering from the University of Catania, Catania (Italy), in 2002 and 2007, respectively.
From September 2005 to February 2006, he was a visiting student with the RNG, Rice University, Houston, TX, under the guidance of Prof. Edward Knightly. Currently, he is a Post-Doctoral Researcher with the DIEEI, University of Catania. His research interests include design and performance evaluation of telecommunication protocols and algorithms, in particular in the area of wireless communications and networking.

Giacomo Morabito was born in Messina, Sicily (Italy) on March 16, 1972. He received the Laurea degree in Electrical Engineering and the PhD in Electrical, Computer and Telecommunications Engineering from the Istituto di Informatica e Telecomunicazioni, University of Catania , Catania (Italy), in 1996 and 2000, respectively. From November 1999 to April 2001, he was with the Broadband and Wireless Networking Laboratory of the Georgia Institute of Technology as a Research Engineer. Since April 2001 he is with the Dipartimento di Ingegneria Informatica e delle Telecomunicazioni of the University of Catania where he is currently Associate Professor. His research interests focus on analysis and solutions for wireless networks.

Corrado Rametta received the Laurea degree in Electronics Engineering from the University of Catania, Italy, in 2008. His final Thesis was on "Study and Evaluation of an Adaptive Technique for Collisions Control in IEEE 802.11 Networks".
He is currently a Ph.D. student with the DIEEI of the University of Catania under the guidance of Prof. Sergio Palazzo. His research interests include: wireless mesh, ad hoc and sensor networks, performance analysis of communication networks.

Lucio Ganga received the Laurea degree in Telecommunications Engineering from the University of Catania, Italy, in 2008. From September 2008 to April 2009 he has done a period of stage at Selex Communications (a Finmeccanica Company). Currently he is a System Engineer at Selex Elsag, and his skill are Network Simulations and interoperability between different Simulation tools through HLA Protocol.

Lloyd Wihl joined Scalable Network Technologies (SNT) in January 2006, where he is Senior Application Engineer, providing worldwide pre-sales support for potential clients, developing system prototypes and technology interfaces, guiding future product development, providing guidance for customer model development, training customers, and managing customer engineering service contracts.
He has over 29 years of experience in the Modeling, Simulation and Training industry. His experience prior to SNT includes 24 years at CAE, where he developed system architectures for military simulation and training, and led multi-million dollar projects in the areas of synthetic military environments, network-centric systems, distributed mission training, air traffic management, space systems, visual systems, and flight simulation.

Agatino Mursia, currently with SELEX Elsag Spa as Head of the Modeling and Simulation (M&S) team of the BU Coomunication for Defence Systems, graduated in Electronics Engineering at the University of Padua in 1989. Since then he has been working in the industry, first with Selenia Spazio Spa, Marconi Italia and then Marconi Communications USA as responsible for the development (Hw and Sw) of communication and networking equipments.
Back to Italy, in 2003 he was appointed as responsible of M&S activities with the role of exploring and promoting the use of M&S as a competitive advantage of products/solutions. He has promoted the use of COTS tools and international simulation standards for the Modeling and Simulation of communications and networking effects into the definition and validation of complex net-centric architectures.
He is also the chairman of the Finmeccanica MindSh@re community SET2 dedicated to simulation that includes most of the Group companies.

MODELING AND SIMULATING DYNAMIC HEALTHCARE PRACTICES

Eugene Santos Jr.[(a)], Keum Joo Kim[(a)], Fei Yu[(a)], Deqing Li[(a)], Phoebe Arbogast[(b)], Elizabeth A. Jacob[(a)], Joseph Rosen[(b)]

[(a)]Dartmouth College, Thayer School of Engineering,
8000Cummings Hall, Hanover, NH 03755-8000, USA
[(b)]Dartmouth-Hitchcock Medical Center
One Medical Center Drive, Lebanon, NH 03756, USA
[(a)]esj@dartmouth.edu, [(b)]Joseph.Rosen@dartmouth.edu

ABSTRACT
Recent research has been undertaken to reduce medical errors and to prevent adverse events that may result from unsafe and insecure situations in complex healthcare practices. Integration of individuals into teams is one of the most challenging but promising issues in the research. Modeling and simulating the complex, dynamic healthcare practices are useful to train individual team members, and subsequently enhance individual and team competencies to boost team performance. In this paper, we propose a methodology to model and simulate dynamic medical situations in healthcare practices by integrating gap analysis with intent inferencing. In intent inferencing, individuals' goals are deduced from their perceptions and observations, and collective intent of individuals is evaluated through gap analysis. As the vast majority of services in healthcare are delivered by a group of individuals, enabling the individuals to figure out the best decision for the patient beyond existing limitations is expected to improve the quality of care significantly.

Keywords: healthcare team, medical procedure, intent inferencing, decision making, gap analysis

1. INTRODUCTION

When a set of planned activities in healthcare practices fails to achieve its original goals, we consider it a medical error. Medical errors have led to a significant number of injuries or patient deaths and have become a topic of much concern (Taib, McIntosh, Caponecchia, & Baysari, 2011). Injuries caused by medical management rather than the underlying disease of the patient are identified as adverse events (Bucknall, 2010). Medical errors and adverse events are not rare outcomes nowadays. It is widely acknowledged that many modern healthcare practices are so complex that they can foster unsafe and insecure conditions for patient care. Of the many causes behind these undesirable events, poor teamwork is a significant contributor. Better integration of individuals into teams and optimization of team performance is a promising strategy to increase the quality of care.

Modeling and simulating the complex, dynamic healthcare practices are important in training healthcare team members, and boosting team performance by enhancing individual and team competencies. An effective team performance can be realized if individual team members have a profound understanding, beyond their own limited perceptions and observations, of surrounding environments and teammates. To realize this, it is necessary to provide sufficient information to healthcare professionals when they need to make critical decisions under diverse requests and conflicting demands. Previously, state of the art computational technologies have been adopted to support safe and secure healthcare practices (Adler-Milstein & Bates, 2010). In the same line of research, we propose a methodology to model and simulate dynamic medical situations in healthcare practices by integrating gap analysis with intent inferencing.

Intent inferencing is a branch of knowledge engineering, in which individuals' goals are deduced from their perceptions and observations. Beyond individuals, collective intent of individuals is addressed by gap analysis in this paper. With the information provided by our computational tool, we hope to enable individuals to make the best decisions for the patient in a given situation

Previously, studies to improve team performance have been conducted through developing measures or indicators of team performance, which were commonly based on clinical surveys, direct observation or video-based analysis of real medical performance (Jeffcott & Mackenzie, 2008). These measures were useful to help train and assess real-life team performance. However, these measures have often overlooked patient conditions, which are dynamically changing over time in real-life healthcare practices, and the fact that individuals (healthcare providers) are likely biased when making decisions in complex situations (Brockopp, Downey, Powers, Vanderveer, & Warden, 2004). Furthermore, the conflict between these individuals during patient management is often the key to deteriorating team performance (Coombs, 2003).

Organizational behaviors can be realized by each individual's discrete efforts to accomplish their roles, plans and goals, while team dynamics depend heavily on all individuals' collaboration, coordination, and communication. Thus, team performance in multidisciplinary practices cannot be measured by simply collecting individuals' movements. Even if we trace all of these behaviors, the available information is still incomplete and insufficient to describe overall team performance. Therefore, we employ not only a computational technique to infer individuals' intents from their observables but also a strategic methodology to evaluate collective intent of individuals through integrating gap analysis with intent inferencing.

In our earlier studies (Santos, et al., 2010), we applied our methodology to model and simulate primarily static instances of real-life medical cases in which adverse events occurred due to the lack of communication between surgeons. In this work, we present an advanced methodology to model and simulate more dynamic situations over longer periods of time in real-life healthcare practices. Through the simulation, we address the impact of differences between individuals in making clinical decisions by investigating a post-op panniculectomy case, in which two surgeons made conflicting decisions for the same patient.

In the next section, we will describe BKBs and other background studies. Then we will introduce our idea to measure team performance with gap analysis. After that, we will present a real-life post-op panniculectomy case and show how we simulate the case with our methodology. Finally, we will end this paper with a conclusion and future directions for this research.

2. BACKGROUND

To ensure patient safety in real-life healthcare practices, individual team members must perform their roles and tasks with continual understanding of dynamic situations and of other team members. In this section, we review three fundamental ideas associated with intent inferencing as a part of our research: (1) representing the information relevant to clinical decision makings, (2) aggregating new information into existing knowledge while properly managing potential inconsistencies, and (3) inferring intents of individual team members from the information observed, perceived and acknowledged by those individuals.

2.1. Bayesian Knowledge Bases (BKBs)

The information available in healthcare practices can be represented by BKBs, which are generalizations of Bayesian Networks (BNs) that allow context-specific independence and cyclic relationships among knowledge. BKBs are rule-based probabilistic models to represent knowledge using graphs and probabilistic theory. The graphs are composed of nodes and arcs, where arcs denote causal relationships between knowledge and nodes contain the content of the knowledge. Unlike BNs, there are two types of nodes in BKBs: the i-node, representing a state of a random variables (i.e. how random variables are instantiated) and the s-node, denoting a conditional probability of the causal relationship as shown in Fig 1, where the knowledge that "if body temperature is high, then a surgeon determines hospitalization as a potential care with the probability of 0.8." is contained.

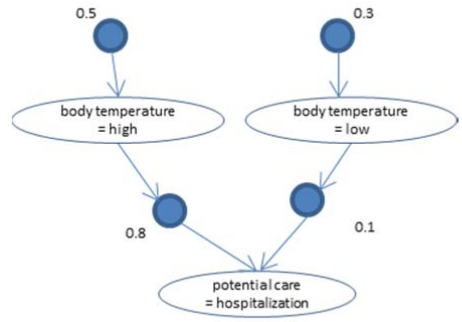


Figure 1: BKB fragment

BKBs are known to be simpler and more concise than BNs in representing knowledge since they can accommodate incomplete knowledge and perform reasoning with less complexity. BKBs have been extensively studied with highly efficient algorithms for reasoning (Rosen, Shimony, & Santos, 2004).

2.2. Bayesian Knowledge Fusion

In order to model dynamic changes in healthcare practices, the information represented in BKBs must be updated accordingly. Bayesian knowledge fusion is an algorithm designed to fuse multiple BKBs into a single, large BKB that preserves the information contained in all input sources. Originally, the fusion algorithm was devised to aggregate information provided by multiple experts (E. Santos, J.T. Wilkinson, & Santos, 2009). In order to handle potential disagreement among different experts, two special nodes are added to original BKBs when fused: the source node and the reliability index. Source nodes say which rules in the knowledge base come from which fragments, while the reliability index denotes the trustworthiness of the knowledge contained in the particular fragment. With these additional nodes, the inference process on the fused BKB can consider information from multiple sources and construct an explanation for any evidence observed without violating the basic rules of BKBs.We apply the algorithm to deal with dynamic situations in healthcare practices where updated information of patient condition must be accounted for and added to existing knowledge bases.

2.3. Intent inference

Healthcare team members' decision making processes can be simulated through individual intent inferencing based on BKBs. Intent, which can be deduced by individuals' actions, can be defined as a combination of goals that are being pursued by individuals. We typically construct a behavioral model by optimizing individuals' behavioral patterns. Thus, we collect data through observing individuals' actions and environments, and deliver them to the model.

BKBs have been applied successfully in various domains, such as adversary intent inferencing and war-gaming, in which human intent was inferred through reasoning with BKBs (Pioch, Melhuish, Seidel, Santos, & Li, 2009) (Santos, McQueary, & Krause, 2008) (Santos, et al., 2007). The instantiation of random variables is represented by i-nodes, which are classified into the four types: axioms, beliefs, goals and actions. These are essential components associated with human intent. Axioms represent what a person believes about himself; beliefs represent what a person believes about others (including other people and surrounding situations); goals represent what results a person wants to achieve; and actions represent what a person will do to realize his goal. Axioms and beliefs may influence themselves or each other, and both can contribute to goals (mostly sub-goals) (E. Santos, 2003).

The intent inferencing can function for three purposes: description of personal insights, prediction of future events and diagnosis of current outcomes. It can describe an insight that motivated individuals and anticipate future actions. In addition, it can assess earlier predictions by contrasting them with current outcomes. It also can provide an explanation of current outcomes.

3. EVALUATING TEAM PERFORMANCE

In addition to individual intent inferencing, we address the collective intent of individuals in teams. Teams and their performances are a crucial and integral part of healthcare practices. To ensure patient safety, teams must be well coordinated and communicate well. As a part of a computational methodology to model and simulate dynamic situations in healthcare practices, we use gap analysis as a way to construct a collective intent of a team and integrate it into the surgical intent inferencing. With this integrated approach, we can simulate real-life medical cases and analyze team performance.

3.1. Surgical Intent Modeling

Surgical intent modeling was proposed to model and simulate the clinical decision-making processes of healthcare professionals. Through this, we aim to improve the healthcare team members' understanding of surrounding environments and other team members' intents (Santos, et al., 2010). Considering the fact that healthcare services involve multiple operations and a wide range of people who must make discrete efforts to accomplish their common goals, tailoring intent models for each healthcare team member is necessary. Surgical intent models are naturally expected to include the entire process of healthcare service from diagnosing to discharging the patient. However, it is intractable to encompass every detail of the entire process even if all of them are necessary to infer intentions completely and accurately. Therefore, we select the most relevant elements with the appropriate level of detail when building the models. For example, the elements we choose for intent models of surgeons are beliefs about the condition of the patient, axioms about the surgeon's own capability in performing the medical procedure, goals regarding choice of procedures, and actions that are taken to fulfill the procedure. In general, surgeons' intent models are the most sophisticated since they have the greatest authority in clinical decision making.

3.2. Individual Differences

It is necessary to understand individual differences and similarities for modeling an individual's decision making processes in healthcare practices. We classify individual differences as either professional or personal. Both of these influence individual competence.

3.2.1. Professional Differences

Individuals are different due to their educational background, malpractice experience, complexity of procedures to take during patient care, etc. Therefore, their roles in the clinical decision-making process and in delivering healthcare services are varied. In general, surgeons have the greatest authority in overall clinical decision-making processes, while nurses have more limited authority to manage patient pain.

3.2.2. Personal Differences

Individuals with the same professional background can be very different in their personalities. In general, individual personalities change over time very slowly. However, some attributes are transient and do not last long. For example, extremely fatigued individuals do not remain in the same state for a long time since the level of fatigue can change relatively quickly. On the other hand, the self-interest level of individuals is more stable, though changes can occur over time.

3.3. Gap Analysis

A medical situation is composed of various individuals and medical devices; medical errors occur when any of these elements does not function appropriately. In medical studies, gap analysis has been used by Calhoun as a way to assess individuals' self-appraisal in communication (Calhoun, Ride, Peterson, & Meyer, 2010). In our research, we use gap analysis to evaluate the performance of a team delivering healthcare services. Based on the probabilistic knowledge representation system used for our research, we compute gap values by comparing probability distributions of individual team members belonging to the same team. Since we believe individuals' intents are well coordinated with the collective intent of the team in an effective team, we consider the team with the smallest gap value as the safest team with respect to medical errors. However, when some individuals make decisions which are in conflict with others' and the collective intent of the team, this leads to deterioration in team performance. By comparing gap values obtained from different teams under the same situation, we can identify which team is more vulnerable to medical errors than others. The formulation to compute gap values can be described as

$$g(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \left| P(i) - P(j) \right|$$

where $g(x)$ denotes the gap value of team x composed by n individual members in an arbitrary situation, and $P(i)$ denotes the likelihood of the world of an individual i in the same situation. The gap value can be computed and interpreted in various ways, but we interpret the gap value as a measure of team performance to deliver healthcare service in a safe and secure manner. Thus, a team having a large gap means that individual team members have a significant discrepancy and low team performance.

4. CASE DESCRIPTION

A patient had a circumferential panniculectomy performed by a general surgeon and a plastic surgeon. The general surgeon was in charge of the mesh work and the plastic surgeon was in charge of the rest of the surgery. During the pre-op, the nurse prepped one side of the patient at the beginning and cleaned the other side a while later, rather than clean both sides at once. The Foley catheter, which is commonly implanted at the beginning of the prep, was implanted in the middle of operation in this case.

After the surgery, the patient was discharged and received home care. After a few days, the visiting nurse reported that the drainage came open and the patient had a lot of pain. The general surgeon suggested admitting the patient to the hospital but the plastic surgeon insisted on home care for a few more days. The disagreement between the general surgeon and the plastic surgeon was never resolved. After a few days, another plastic surgeon took over the case since the original plastic surgeon was out of town. The new plastic surgeon decided to admit the patient immediately and pursue a follow-up procedure. By that time, the patient had already experienced a lot of pain in the past few days. During the follow-up procedure, it was confirmed that the wound had been infected. The original plastic surgeon should have admitted the patient immediately after the wound opened.

4.1. Panniculectomy Case during 5 days after OR

For modeling and simulating the case dynamically, we shortened the duration of the care from 2 weeks to 5 days after the patient had the panniculectomy operation and was discharged from the hospital. We simplified this case because the patient condition did not change so dramatically that we needed to model each actual day. In addition, we are using a discrete representation of information. As shown in Table 1, we assume the patient condition worsened from Day 1 to Day 4 (as shown by the numbers from -2 to -10) and recovered on Day 5 after both surgeons (general surgeon and the new plastic surgeon) agreed on readmitting the patient to the hospital ("Home" denotes the surgeon's decision to discharge the patient from the hospital and take care of him at home while "Hosp" represents the surgeon's decision to readmit the patient to the hospital).

Table 1: Change of Patient Condition and Surgeons' Decisions

	Day1	Day2	Day3	Day4	Day5
Patient Status	-2	-5	-7	-10	-5
General Surgeon	Home	Hosp	Hosp	Hosp	Hosp
Plastic Surgeon	Home	Home	Home	Hosp	Hosp

4.2. Possible Cases depending on Personalities

In order to validate our approach, we modeled four possible medical situations, where the major differences were in the surgeons' different types of interests. In each case, we assumed a healthcare team composed of four individuals: general surgeon, plastic surgeon, nurse, and patient. For the panniculectomy case, we speculated on the role of the plastic surgeon in delivering the healthcare service and varied his self-interest while fixing other members' best-interests to patient-health as the highest priority. While varying the plastic surgeon's best-interest, we addressed four categories: patient preference, patient health, surgeon liability, and surgeon cost. If the plastic surgeon considers a patient's preference as his first priority, he will make a decision that conforms to the patient desires. If a surgeon considers a patient's health to be the highest concern, he makes a decision that can improve a patient's health most. When surgeons seek to reduce liability as their primary interest, they make decisions that help reduce their future liability in case any incidents happen. Pursuing surgeon cost as a primary interest refers to the situation in which a surgeon makes a decision to maximize his individual or organizational income. In a real situation, a surgeon tends to pursue a mix of these four best-interests rather than only one. Thus, we hypothesize four possible cases, each of which represents a different type of best-interests. Each type of best-interest can contribute to individual's best-interest proportionately and the weights used for each case are presented inside the parentheses. The weights that are not specified explicitly are set at 0%. Except for the plastic surgeon, we assume the best-interest of other team members is patient-health at 100%.

4.2.1. Case 1

A plastic surgeon focuses on both satisfying a patient's preference (refers to his preference on the care he will receive based on his economic situation, physical and mental condition and so forth) and the patient's health during a decision-making process. The weights for two types of best interests are roughly equivalent (patient preference=100%, patient health=80%).

4.2.2. Case 2:

A plastic surgeon considers a patient's health to be the most important factor when making a decision (patient health =100%).

4.2.3. Case 3:

A plastic surgeon focuses on reducing his/her liability while improving a patient's health. The weights for these two types of best-interests are roughly equivalent (patient health=80%, surgeon liability=100%).

4.2.4. Case 4:

A plastic surgeon focuses on reducing his/her liability and improving a patient's health. The weight of liability is considerably larger than that of patient health (patient health=50%, surgeon liability=100%).

4.2.5. Experimental Results

We used the BKB fusion algorithm to simulate the dynamic situations in the panniculectomy case. The generic BKBs for two surgeons are similar in most parts of their decision-making processes and have minor differences due to their different roles. In addition, we consider the visiting nurse and the patient as separate BKBs as well. Even though they are not active decision-makers in the patient's care, we assume they both play some roles through providing supplementary information to the surgeons.

In order to simulate the dynamics of the surgeons' decision making processes, which is based on the patient condition that changes over time, we used the BKB fusion algorithm (E. Santos, J.T. Wilkinson, & Santos, 2009). Through the experiments conducted, we validate that BKBs can represent the dynamics in medical decision making when the patient conditions are changed.The fragments of BKBs, which refer to the input BKBs in the fusion process, are relatively small and contain only the information representing the changes through new i-nodes that influence the distribution of pre-existing i-nodes.

With the BKBs specified above, we conducted two sets of experiments to examine whether the BKBs and their fusion approach can provide a true representation of knowledge and correlations among them. In static validation, we tested the BKBs on Day 1, with varying professional and personal differences. In dynamic validation, we tested if the fused BKBs accurately represent the changes made in decision-making processes with regard to the dynamic patient condition.

4.3. Static Validation

The purpose of our static validation is to test if the BKBs constructed to represent individuals in a healthcare team can truly represent a wide range of individuals and their decision making processes. Since the professional and personal attributes of individuals do not change over a short time period in general, we assume these attributes are static during the time period under our consideration. For example, surgeons' experience does not change during a 5-day or 2-week period. In addition, personal self-interest does not change within a limited time, although it may change smoothly over a longer time period (years or decades).

4.3.1. Professional Differences vs. Error Probability

As for professional differences, we considered experience, complexity and malpractices. One of our general assumptions is that less experienced individuals make mistakes with a higher probability than highly experienced individuals. Table 2 represents the results of experiments obtained through the surgeon's BKB. In addition, the malpractice, experience and complexity are denoted as M, E and C, respectively and the two levels of malpractice, experience and complexity are presented as Low (L) and High (H). As shown in Table 2, when the complexity of the procedure is high, the surgeon is highly likely to change his decision from home care to hospitalization when his level of malpractice and experience is low since the surgeon would like to ensure patient safety by keeping him and the medical equipment more readily accessible. However, the patient can be taken care of well through home care if the surgeon is highly experienced. If the surgeon has a high malpractice history, he would be more risk-averse and would likely change his decision from home care to hospitalization when the procedure is highly complex even if he is experienced enough with the procedure.

Although we confirmed that all individual BKBs follow this tendency, we present here only the experimental results obtained from the general surgeon's BKB.

Table 2: Professional Differences vs. Error Probability

Evidence				Target (Planned Procedure)		
Potential Procedure	M	E	C	1st rank	2nd rank	1st rank prob.
Home	L	L	L	Home	Hosp	1.81e-05
Home	L	L	H	Hosp	Home	1.09e-05
Home	L	H	L	Home	Hosp	2.71e-05
Home	L	H	H	Home	Hosp	1.92e-05
Home	H	L	L	Home	Hosp	2.13e-06
Home	H	L	H	Hosp	Home	1.83e-06
Home	H	H	L	Home	Hosp	2.01e-06
Home	H	H	H	Hosp	Home	1.22e-06

4.3.2. Personal Differences vs. Error Probability

As personal differences, we address the self-interest of surgeons and nurses. Table 3 demonstrates a few examples of how different types of interests influence the final decision when the patient's condition is normal. As evidence for best-interest, PP, PH, SL and SC represent patient preference, patient health, surgeon liability and surgeon cost respectively, as explained in

Section 4.2. Each row represents how a surgeon determines his procedure when his best-interest is set as evidence. For example, the first row represents how a surgeon determines home care as the best procedure when his best-interest is patient preference, which is home care in this example.

Table 3: Personal Differences vs. Error Probability

Evidence	Target (Planned Procedure)		
Best-interest	1st rank	2nd rank	1st rank prob.
PP (Home)	Home	Hosp	0.0013
PP (Hosp)	Hosp	Home	0.0013
PH	Home	Hosp	0.0025
SL	Home	Hosp	0.0027
SC	Hosp	Home	0.0023

4.4. Dynamic Validation

Based on the static validation, we expanded the simulation into a 5-day period to validate that the fused BKBs represented the dynamics of the panniculectomy case accurately. To this end, we conducted an additional set of experiments and computed gap values over time for each case we addressed earlier.

4.4.1. Dynamics of Potential Procedure

In the panniculectomy case, the only source of dynamics is the change in patient condition, such as the wound opening and drainage fall. The potential procedure must cope with this change of patient condition. Therefore, we conducted a set of experiments to test if the procedure predicted by inferencing with the BKB changes according to the patient condition, as shown by Table 4.

Table 4: Dynamics of Potential Care

		Target (Potential Procedure)		
Case	Day	1st rank	2nd rank	1st rank prob.
1	1	Hosp	Home	0.00134
	2	Hosp	Home	1.32E-06
	3	Hosp	Home	8.23E-07
	4	Hosp	Home	2.22E-08
	5	Hosp	Home	1.49E-08
2	1	Home	Hosp	0.002489
	2	Hosp	Home	2.27E-06
	3	Hosp	Home	1.88E-06
	4	Hosp	Home	5.54E-08
	5	Hosp	Home	2.49E-08
3	1	Home	Hosp	0.00268
	2	Home	Hosp	2.64E-06
	3	Home	Hosp	1.65E-06
	4	Hosp	Home	4.44E-08
	5	Hosp	Home	2.49E-08
4	1	Home	Hosp	0.00268
	2	Home	Hosp	2.64E-06
	3	Home	Hosp	1.65E-06
	4	Home	Hosp	3.92E-08
	5	Home	Hosp	2.99E-08

4.4.2. Gap Analysis in Panniculectomy Case

Gap values were computed using Equation (1) for each case mentioned in Section 4. As shown in Figure 2, we obtained the lowest gap value from case 2 during the 5 days since the both the general and plastic surgeons placed their best-interests towards patient health. In case 1, the plastic surgeon's best interest is set towards patient preference and health, and he insists on readmitting the patient to the hospital from day 1. Although his motivation is not ideal, his decision turns out to be good for the patient health from day 2 since the patient condition gets worse. In case 3, the plastic surgeon insists on home care since he cares about his liability in addition to the patient health. However, since the patient condition gets worse, he changes his decision to readmit the patient to the hospital at day 4. The gap value becomes negligible after the patient was re-hospitalized. Case 4 is a more severe case with respect to the patient safety since this plastic surgeon is more biased to his liability issue and insists on home care until day 5. However, the gap value becomes smaller as time goes on since the plastic surgeon would become skeptical of his decision when the patient condition worsened.

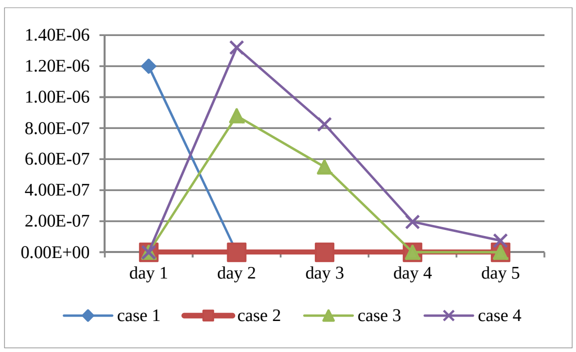


Figure 2: Gap Analysis with four cases

5. CONCLUSION AND FUTURE WORKS

In this paper, we proposed a new computational framework to simulate dynamic healthcare practices by employing the Bayesian knowledge fusion method developed to aggregate the information from multiple sources. By modeling and simulating complex real-life situations, we expect to contribute to training healthcare professionals and ensuring patient care. We also addressed team performance through gap analysis by integrating gap analysis with individual intent inferencing. Consequently, we hope to supply healthcare practitioners with information of complex situations and help them make the best decision for the patient.

ACKNOWLEDGMENTS

The work was supported in part by grants from the Office of Naval Research and Air Force Office of Scientific Research.

REFERENCES

Adler-Milstein, J., & Bates, D. W. (2010). Paperless healthcare: Progress and challenges of an IT-enabled healthcare system. *Business Horizons*, 119-130.

Brockopp, D. Y., Downey, E., Powers, P., Vanderveer, B., & Warden, S. (2004). Nurses' clinical decision-making regarding the management of pain. *International Journal of Nursing Studies*, 631-636.

Bucknall, T. K. (2010). Medical error and decision making: Learning from the past and present in intensive care. *Australian Critical Care*, 150-156.

Calhoun, A. W., Ride, E. A., Peterson, E., & Meyer, E. C. (2010). Multi-rater feedback with gap analysis: An innovative means to assess communication still and self-insight. *Patient Education and Counseling*, 321-326.

Coombs, M. (2003). Power and conflict in intensive care clinical decision making. *Intensive and Critical Care Nursing*, 125-135.

Santos, E. Jr., *et al*. (2010). a, "Promoting Team Performance in the Operating Room by Analyzing Gaps among Surgical Care Members. In *Modeling and Simulation in the Medical and Health Sciences.*

Santos, E. Jr., *et al*. (2010). A method to improve team performance in the OR through intent inferencing. *7th EUROSIM Congress on Modeling and Simulation.* Prague, Czech Republic.

Santos, E. Jr. (2003). A Cognitive Architecture for Adversary Intent Inferencing: Knowledge Structure and Computation. *SPIE 17th Annual International Symposium on Aerospace/Defense Sensing and Controls AeroSense 2003*. Orlando, FL.

Santos, E. Jr., Wilkinson, J. T., & Santos, E. (2009). Bayesian knowledge fusion. *22nd International FLAIRS Conference.* AAAI Press.

Jeffcott, S. A., & Mackenzie, C. F. (2008). Measuring team performance in healthcare: Review of research and implications for patient safety. *Journal of Critical Care*, 188-196.

Pioch, N., Melhuish, J., Seidel, A., Santos, E. Jr., & Li, D. (2009). Adversarial Intent Modeling Using Embedded Simulation and Temporal Bayesian Knowledge Bases. *SPIE Defense and Security Symposium*, *7348*, pp. 1-12. Orlando, FL.

Rosen, T., Shimony, S., & Santos, E. Jr. (2004). Reasoning with BKBs - Algorithms and Complexity. *Annals of Mathematics and Artificial Intelligence, 40*(3-4), 403-425.

Santos, E. Jr., McQueary, B., & Krause, L. (2008). Modeling Adversarial Intent fot Interactive Simulation and Gaming: The Fused Intent System. *SPIE: Defense and Security Symposium, 6965*. Orlando, FL.

Santos, E. Jr., Zhao, Q., Pratto, F., Pearson, A., Mcqueary, B., Breeden, A., et al. (2007). Modeling Multiple Communities of Interest for Interactive Simulation and Gaming: The Dynamic Adversarial Gaming Algorithm Project. *SPIE: Defense and Security Symposium, 6564.* Orlando, FL.
Santos, E. Jr., *et al*. (2010). A method to improve team performance in the OR through intent inferencing. *7th EUROSIM Congress on Modeling and Simulation.* Prague, Czech Republic.

Santos, E. Jr., *et al*. (2010). a, "Promoting Team Performance in the Operating Room by Analyzing Gaps among Surgical Care Members. In *Modeling and Simulation in the Medical and Health Sciences.*
Jeffcott, S., & Mackenzie, C. F. (2008). Measuring team performance in healthcare: Review of research and implications for patient safety. *Journal of Critical Care*, 188-196.

Taib, I. A., McIntosh, A. S., Caponecchia, C., & Baysari, M. T. (2011). A review of medical error taxonomies: A human factors perspective. *Safety Science*, 607-615.

AUTHORS BIOGRAPHY

Eugene Santos Jr., Ph.D., is currently Professor of Engineering in the Thayer School of Engineering at Dartmouth College, Hanover, NH. He is currently Editor-in-Chief for the *IEEE Transactions on Systems, Man, and Cybernetics: Part B*, an associate editor for the *International Journal of Image and Graphics*, and is also on the editorial advisor board for *System and Information Sciences Notes* and on the editorial boards for *Journal of Intelligent Information Systems* and *Journal of Experimental and Theoretical Artificial Intelligence*.

Keum Joo Kim, Ph.D., is currently working for Thayer School of Engineering at Dartmouth College as research associate. She has been a member of *Sigma Xi* and *IEEE*. She has served for several professional communities as reviewer such as *the IEEE Transactions on Systems, Man and Cybernetics: Part B, International*

Journal of Image and Graphics, the Data and Knowledge Engineering Journal, the International Conference on Parallel Processing, Journal of Supercomputing, etc.

Fei Yu is currently a Ph.D. candidate at the Thayer Engineering School, Dartmouth College. She has published papers in *the Proceedings of WI-IAT Conference*, *the Proceedings of the UMAP Conference*, and *the Proceedings of the EUROSIM Conference*. She also has published a book chapter in the *E-Government Diffusion, Policy, and Impact: Advanced Issues and Practices*. Her research interest is in intent and cultural modeling, user modeling, and natural language processing.

Deqing Li is currently working toward the Ph.D. degree at the Thayer Engineering School, Dartmouth College. She has published papers in *the Proceedings of SPIE* and *the Proceedings of the IEEE Conference on Systems Man and Cybernetics*, and a book chapter in the *E-Government Diffusion, Policy, and Impact: Advanced Issues and Practices*. Her research interest is in decision theory, intent modeling, and related applications.

Phoebe Arbogast received her B.A. from Amherst College in 2010. She is a project manager, research assistant, and teaching intern for Joseph Rosen, M.D. and is currently enrolled in Northwestern University Feinberg School of Medicine.

Elizabeth Jacob received a B.S. in Biology with a minor in English from Dartmouth College in June of 2011. She is currently a research intern with the Distributed Information and Intelligence Analysis Group (DI2AG) under Professor Eugene Santos Jr. at Thayer Engineering School, Dartmouth College. She has published a paper in the *EUROSIM Conference* and a book chapter in the text, "Patient Care: Modeling and Simulation in the Medical and Health Sciences" (Eds. J. A. Sokolowski and C. M. Banks).

Joseph M. Rosen, M.D., is Professor of Plastic Surgery at Dartmouth-Hitchcock Medical Center, an Adjunct Professor at the Thayer School of Engineering at Dartmouth, and serves on the staff of the White River Junction VA. He is currently active in policy and doctrine advising for the Department of Defense, chairing a Sub-Panel on Improvised Explosive Devices (IED) under the Defense Science Board in 2008. He has studied nerve repair for over 20 years and his other research interests include Human Machine Interfaces, Microsurgery and Transplantation of Limbs, Computer-Aided Surgery, Virtual Reality Simulators and methods of Surgical Education, Telemedicine and Informatics, Healthcare Delivery for Medical Disasters, and countermeasures for Bio-Terrorism.

EXPLORATORY SEQUENTIAL DATA ANALYSIS OF A CYBER DEFENCE EXERCISE

Dennis Andersson[(a)], Magdalena Granåsen[(a)], Thomas Sundmark[(a)], Hannes Holm[(b)], Jonas Hallberg[(a)]

[(a)]Swedish Defence Research Agency, Box 1175, SE-581 11 Linköping, Sweden
[(b)]KTH Royal Institute of Technology, Kungl Tekniska Högskolan, SE-100 44 Stockholm, Sweden

[(a)]{dennis.andersson,magdalena.granasen,thomas.sundmark,jonas.hallberg}@foi.se, [(b)]hannes.holm@ics.kth.se

ABSTRACT
Baltic Cyber Shield 2010 (BCS), a multi-national civil-military cyber defence exercise (CDX), aimed to improve the capability of performing a CDX and investigate how IT attacks and defence of critical infrastructure can be studied. The exercise resulted in a massive dataset to be analyzed and many lessons learned in planning and executing a large-scale multi-national CDX. A reconstruction & exploration (R&E) approach was used to capture incidents such as attacks and defensive counter-measures during the exercise. This paper introduces the usage of R&E combined with exploratory sequential data analysis (ESDA) and discusses benefits and limitations of using these methods for analyzing multi-national cyber defence exercises.

Using ESDA we were able to generate statistical data on attacks from BCS, such as number of reported attacks by the attackers and the defenders on different type of services. Initial results from these explorations will be analyzed and discussed.

Keywords: cyber defence exercise, data analysis, experimentation, reconstruction and exploration

1. INTRODUCTION

Most organizations and services are critically dependent on reliable and secure information systems. Thereby, cyber warfare and terrorism is becoming a significant threat to recognize in today's society. Incidents such as the cyber attacks on Estonia in 2007 and the attacks on U.K., U.S., German and French resources in 2005 (Greenemeier 2007) are frequently cited and evidences of that the threat is real. However, the amount of publicly available data from such incidents is limited, which makes it difficult to study the associated phenomena. Hence, there is a need for data that conceptualize the phenomena of cyber warfare and terrorism, which thereby motivates cyber defense exercises (CDX) simulating such attacks and training teams in how to defend critical information systems.

In May 2010, the Cooperative Cyber Defense Centre of Excellence and the Swedish National Defense College hosted the Baltic Cyber Shield (BCS) international cyber defense exercise (CDX). For two days, six Blue Teams from northern European government, military and academic institutions defended simulated power generation companies against a Red Team of 20 computer hackers. The scenario described a volatile geopolitical environment in which a hired-gun Rapid Response Team of network security personnel defended Critical Information Infrastructure (CII) from cyber attacks sponsored by a non-state terrorist group. (Geers 2010)

The technical infrastructure was designed and implemented in a computer cluster located at, and hosted by, the Swedish Defense Research Agency (FOI). Each blue team network consisted of a number of virtual computers on the cluster, containing vulnerabilities to be exploited by the red team. The network connections were established through Virtual Private Networks (VPNs) enabling the teams to be physically distributed. Moreover, the networks were connected to the Programmable Logic Controllers (PLCs) of a power infrastructure model, including steam engines, solar panels, a simulated distribution network and factories with butane flames that could be detonated by the red team. Thus, a mixed-reality supervisory control and data acquisition (SCADA) network was created. (Hammervik, Andersson and Hallberg 2010)

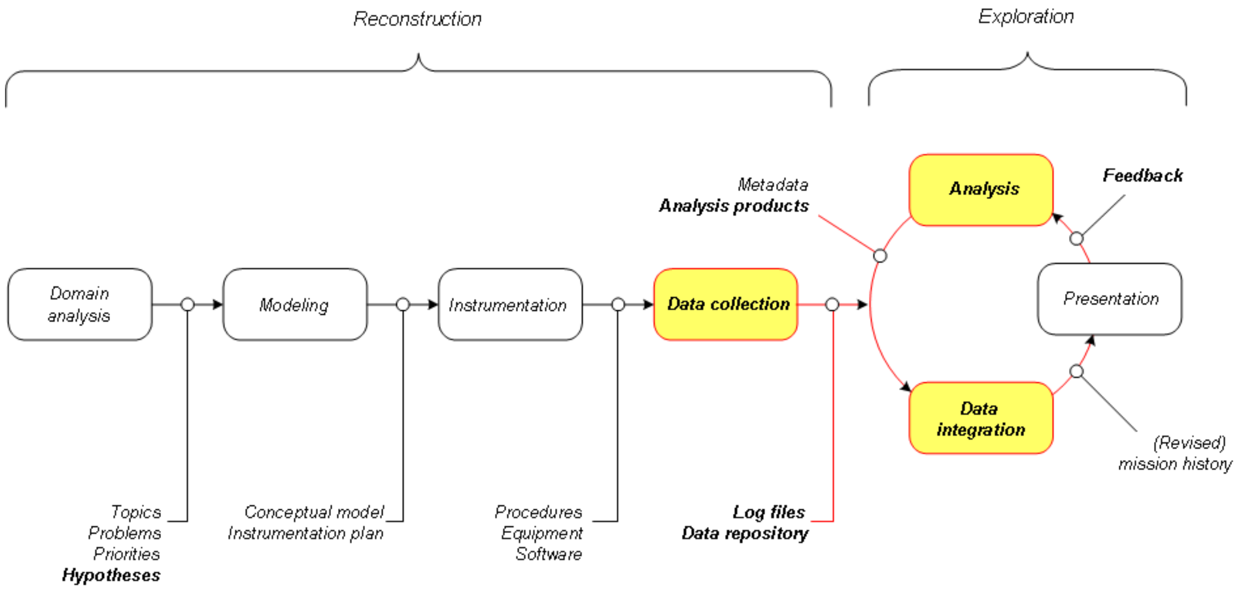


Figure 1: The Process of Reconstruction & Exploration (Andersson 2009)

The BCS CDX had three main goals: training of the Blue Teams, highlighting the international aspects of cyber defense, and improving the knowledge on how to perform CDXs (Geers 2010). To accomplish those goals, the outcome of the CDX needs to be carefully studied in terms of teamwork, collaboration models, scenario validity, attack patterns and C2 structures.

As the teams were given a large degree of freedom in how to organize and perform their work – there was little a priori knowledge from the experiment team on how the events would unfold and what phenomenon to monitor. A consequence of this is that there are too many unknown variables to adequately model the teams and their processes as would be desired. Instead, we chose to collect a massive heterogeneous dataset, containing both qualitative and quantitative data.

The resulting model has shown great potential for creating an understanding, or *situation awareness*, when studying the course of events after the exercise, and therefore enabling discovery experimentation (Alberts and Hayes 2002) using the exercise data. The objective of this paper is to describe the actual data collection, the analysis process, and discuss initial findings.

2. METHOD

During planning of the exercise, it was soon recognized that a structured way of organizing data collection was needed to be able to handle the multitude of available data sources and enable the analysis required to fulfill the goals of the BCS CDX. The Reconstruction & Exploration approach (R&E) (Figure 1) was selected due to its capacity to deal with large and complex data sets as well as being well-known by the analyst team (Andersson 2009). R&E was originally designed for use

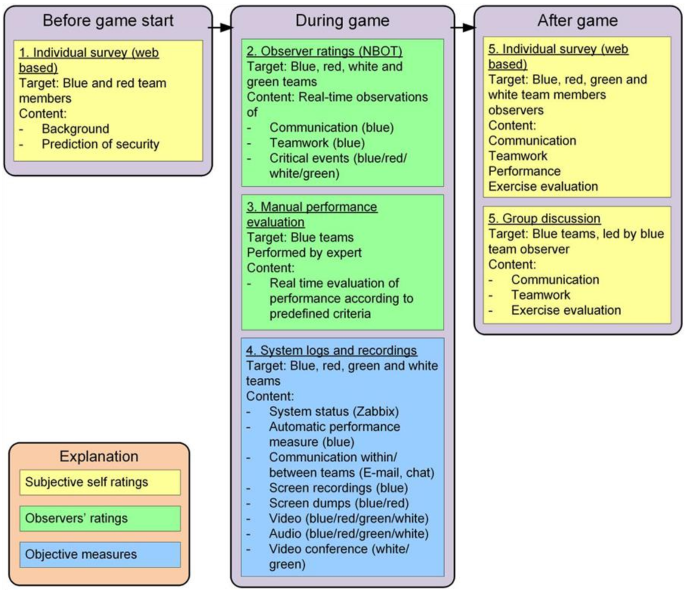


Figure 2: Prioritized Data Collection Nodes

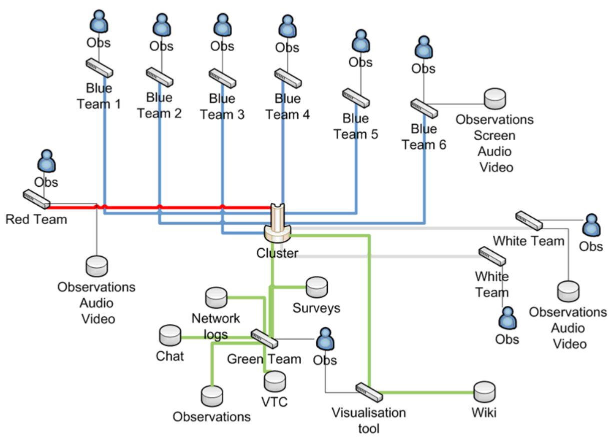


Figure 3: Logical Distribution of Teams, Observers and Data Collection Nodes during BCS 2010 (Hammervik, Andersson and Hallberg 2010)

with distributed tactical operations (DTOs) like military or crisis management operations (Pilemalm, Andersson and Hallberg 2008) and as such had never before been applied in its entirety to the IT security domain.

R&E consists of 7 steps, domain analysis, modeling, instrumentation, data collection, data integration, presentation and analysis. The output of the minimalistic domain analysis and modeling showed the need for collection of both quantitative and qualitative data (Figure 2) to enable reconstruction of situation awareness for the analyst, with the main focus on data from the blue teams. However, data collection also included red team activities as a reference for understanding blue team actions. Objective data in terms of system logs was assumed to provide results on the teams' activity in the system, but in order to understand why the teams chose the actions they did, there was a need to collect also the participants' views of what was happening and the reasoning within the teams. Therefore, it was decided that observers would be placed within each team, and that questionnaires would be used as a means to collect the subjective estimations of what they were experiencing.

It was decided that video cameras, audio recorders, screen capture tools and human observers should be placed in each team and surveys were to be distributed among the training audience (Figure 3) to try to capture the behavioral aspects of the teams. Observers were equipped with Network-Based Observation Tool (NBOT) (Thorstensson 2008) to enable quick and intuitive reporting of interesting events. Data collection for the objective measures included e-mails, chat sessions, keyboard interactions, network traffic and utilization of memory, processors and hard disk space on each node in the virtual network. In order to capture screen video and keyboard interactions, custom made scripts had to be installed on every machine used in the network. Because it was decided that some of the teams should use their own computers, the analyst team had to rely on participants' willingness to cooperate and install these scripts on their respective machines. For the teams that were supplied workstations by the exercise organizers, however, it was easier to setup and control this logging. For the supplied Windows computers a custom-made screen capture program was used, while on Linux the participants were recommended to use xvidcap, but any other appropriate application was allowed. To capture the terminal I/O a script to be executed by the participants was supplied as part of the team packages. Some data, such as e-mails, video feeds and NBOT reports were also available in real-time for the exercise judges (the White team, WT) who used that information to score blue team performance (Geers 2010).

Data was collected throughout the whole two-day exercise, in total 3 TB of data was collected. F-REX (Andersson 2009) was used for the exploration part of R&E. F-REX is a completely configurable tool

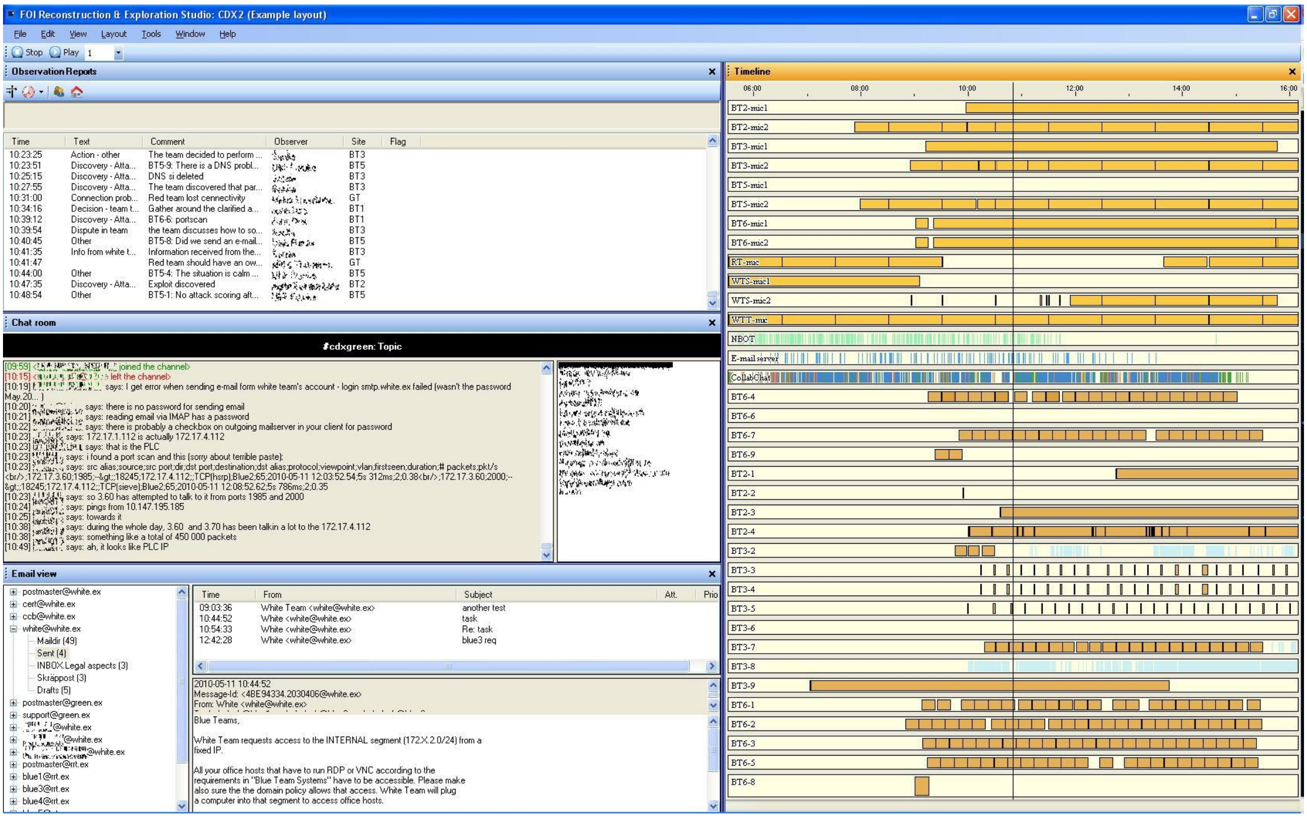


Figure 4: F-REX screenshot showing analysis in the first cycle. The layout shows observer reports, chat room log, e-mail to the left and a timeline of events currently in the mission history, separated by source, to the right. (Note: names are scrambled to preserve anonymity)

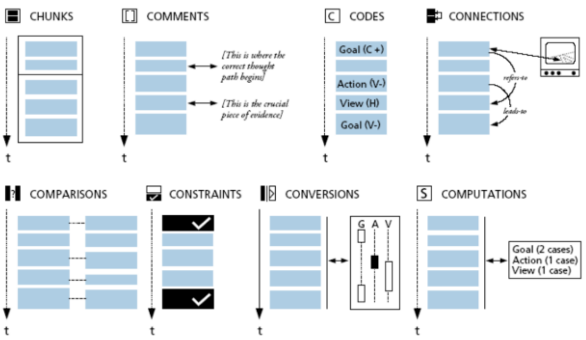


Figure 5: 8C's (Sanderson 1991)

allowing users to view a large and heterogeneous data set in a uniform and synchronized manner, much like playing a DVD back and forth (Figure 4). Its features include quick and easy timeline-based navigation in data based on timestamps from the data collection tools. At any time the analyst can shift focus in F-REX by applying a new layout with any views he or she prefers.

The captured data could be imported to an F-REX project, known in R&E as a mission history, for further synchronization, presentation and analysis according to the Exploratory Sequential Data Analysis (ESDA) method (Sanderson 1991). ESDA is an empirical exploratory approach, as opposed to confirmatory, in which the analyst uses temporal ordering of data to try to make sense of a dataset. Sanderson describes ESDA as a family of observational methodologies that are used when the objective is to observe what people do over time. Sanderson & Fisher (Sanderson and Fisher 1994) outlines the 8 C's (Figure 5) as the main operations needed in ESDA, the reader is advised to read their article for a thorough explanation of the 8 C's. They do not claim that all of the 8 steps must be performed at every study, nor do they have to be performed in a particular order. However, the way that they present them seems to create a fairly logical work flow that can easily be followed using F-REX and the exploration part of R&E.

In R&E, the exploration phase is cyclic, with analysis results and presentation comments being fed back into the model to create revised mission histories. In this study, the first reconstruction cycle used only chat logs, e-mail communication and observer reports - simply because they were estimated to generate the most value for least effort into the analysis. It is easy to assume that neglecting a large portion of the data set as this will impact results. This is, however, merely a case of "data guilt" as Fisher & Sanderson so accurately defines "unless there is a formal commitment to analyzing all the data to meet sampling assumptions, it may not be necessary. Thoroughly analyzing a subset of the data may be more informative". (Fisher and Sanderson 1996)

3. RESULTS

A CDX differs from the type of operations that are typically analyzed by the team in the sense that almost all data is in the virtual domain, and there is little real action to observe and analyze in the physical world. As such, the R&E approach had much to prove. Setting up the data capture was indeed a journey into uncharted waters as, to the authors' knowledge, this type of comprehensive data capture had never before been tried in a CDX.

Since most of the action happens in the virtual domain, in a CDX data capture is mostly a matter of running software that log system-system and human-system interactions. It turned out that it is not always easy to capture these interactions in a easily quantifiable format, as many programs, protocols and data formats being used are proprietary. The chosen fallback solution was to capture screen videos, keyboard interactions and network traffic for systems that could not be tapped in to in any other way. This data capture is not optimal because it is very crude and hard to interpret as needed to do the chunks, connections and codes. Still, the work proved possible, although very time consuming. The process became especially cumbersome since some of the blue teams were allowed to use their private laptops, and the data capturing was therefore dependent on their willingness to install and run special software and scripts to capture these interactions. Getting them to do so proved difficult, probably because of lack of understanding of the importance of the evaluation process. As a result the data set is missing some interactions that could potentially be vital for the detailed analysis of the work that was going on in these teams, and as mentioned before this introduces too many unknown variables to enable a satisfactory analysis of these interactions.

We found however, that the data discussed above can be used by analysts to acquire an increased understanding of the *what*, *when*, *where*, *who* and *how* (Whetten 1989) of the events that unfolded in the exercise. The remaining question, and the most fruitful one, *why*, is less straight-forward, and we found that there are too many unknowns within the setup of the exercise to fully answer why the teams took certain actions with only that data.

Perhaps the answer to the *why* lies in the multimedia data that was captured, i.e. video cameras and audio recorders in each team that were employed to capture human-human interactions. Analysis of these interactions is typically very time consuming, but resembles more traditional R&E work (and ESDA for that matter). This analysis has yet to be performed, but is essential not only to understand the *why*, but also to answer questions such as how human-human interaction affects the team collaboration or performance in a CDX.

Table 1: Compromised Services as Reported by Attacking vs. Defending Teams

Service	# reports by attacking team (s_a)	# reports by defending teams (s_d)	s_d/s_a
Operator	2	1	0.500
Fileserver	5	1	0.200
External firewall	4	3	0.750
Historian	8	3	0.375
Mail server	6	9	1.500
News server	4	5	1.250
DNS/NTP	1	3	3.000
Database	3	3	1.000
Intranet	3	2	0.667
Public web server	11	12	1.091
Portal	6	7	1.167
Other	7	13	1.857

The third type of data capturing that took place during the BCS CDX was surveying. The background survey showed that the teams consisted of highly experienced personnel on both a technical and strategic level, most of which worked with IT security on a daily basis. Having this in mind, the participants perceived scenario complexity and realism as perfectly sufficient and were highly motivated throughout the exercise. Teamwork was experienced as smooth, probably due to that most team members were familiar to each other. In some teams, the members reported lacking technical competencies within fields experienced as crucial, which could be a possible explanation to differences in performance between the teams. Another aspect which was captured using surveys was the participants' prior assumptions regarding the probability of successful compromise of hosts with specific properties. These data were not included in the mission history, but are expected to be useful for separate studies, such as comparing experts' expectations with actual results to measure the accuracy of expert assessment as a metric for IT security.

As mentioned before there was not enough control within the exercise to a priori generate variables to measure. Instead the analyst team put together data from the exercise to try to make sense of what actually happened from both a technical and teamwork point of view. The first version of the mission history enabled finding an initial classification of the targets for all discovered compromises, as reported by the red and blue teams respectively (Table 1). The table does not yield any strong interpretations, however it hints that the most frequently attacked services during the BCS CDX were the historian, the public web server and the customer portal. The defending teams seem to have reported most of the incidents on the public web servers and the customer portals, while the attacks on the historians would be more likely to have passed undetected.

Our experience from this work is that ESDA is a very useful complement to R&E when analyzing massive multimedia-heavy datasets such as the one collected during the BCS CDX. While one can argue that any analysis made with the assistance of R&E could be categorized as ESDA, it is the structured way of working through the data set, as outlined by the 8C's, that makes ESDA so powerful. From our experience, the 8C's should be considered as guidelines that help structuring the analysis process.

4. CONCLUSION

This study has shown the successful use of R&E and F-REX for analyzing cyber defense exercises (CDXs). In order to perform the actual analysis, Exploratory Sequential Data Analysis was applied in the exploration phase. R&E with ESDA has shown great potential for analyzing CDXs.

It can be argued that any analysis with R&E is automatically ESDA and that would indeed be the case according to the definition of Fisher & Sanderson, since they do not enforce usage of all C's or enforce a certain ordering between the steps. Being aware of ESDA and the 8C's when performing the analysis helps with structuring the analysis and as such ESDA should be regarded as a useful technique to know for R&E analysts.

Capturing human-human interactions in a CDX is not very different from any DTO, although it is reasonable to assume that more of the communication will use digital foras, as opposed to a DTO which typically uses radio as the primary means of communication. A CDX does however, focus more on human-system interactions, which are not always easy to capture. To successfully do so, the analysts must carefully plan their instrumentation. Moreover, it is important to work closely together with the exercise organizers to make sure they understand the need for capturing the necessary data.

For the data collection part it could be concluded that having an observer tool with predefined coding schemas was very helpful for the observers and the analysts, but that the coding schema needs to be tested and verified in advance to avoid having to change schema during the exercise. The observer reports and the different teams' self-reporting via e-mail seem to be the most valuable resources for analyzing the data. From the reports it seems that the historians, the portals and the public web servers were the most frequently attacked targets during the BCS CDX.

Although the CDX does not primarily serve as an experiment, we have shown that the data set acquired from it can be used for discovery experimentation

(Alberts and Hayes 2002). To enable more detailed studies on specific research questions, e.g. relating to IT security or teamwork, a higher level of experiment control than was used in BCS CDX 2010 is desired.

ACKNOWLEDGMENTS

This paper was originally published in proceedings of the 16th International Command and Control Research and Technology Symposium (ICCRTS) in Québec City, Canada (Andersson, Granåsen, Sundmark, Holm and Hallberg 2011). The authors want to thank the organizers of the Cyberspace management track, the reviewers and the audience of ICCRTS who contributed with valuable comments on the work. This version of the paper contains minor modifications from the original version, with the objective of addressing some of those comments. In addition, the comments from the anonymous reviewers of DHSS 2011 have been taken into account to further improve the contribution of the paper.

REFERENCES

Alberts, D. and Hayes, R., 2002. *Code of Best Practice for Experimentation*, Washington: CCRP Publication Series.

Andersson, D., 2009. F-REX: Event-Driven Synchronized Multimedia Model Visualization, *Proceedings of the 15th International Conference on Distributed Multimedia Systems*, pp. 140-145, September 10-12, Redwood City (California, USA).

Andersson, D., Granåsen, M., Sundmark, T., Holm, H. and Hallberg, J., 2011. Analysis of a Cyber Defense Exercise using Exploratory Sequential Data Analysis, *Proceedings of the 16th International Command and Control Research and Technology Symposium*, June 21-23, Québec City (Québec, Canada).

Fisher, C. and Sanderson, P., 1996. Exploratory sequential data analysis: exploring continuous observational data, *Interactions*, 3(2), pp. 25-34, New York (New York, USA).

Geers, K., 2010. Live Fire Exercise: Preparing for Cyber War, *Journal of Homeland Security and Emergency Management*, 7(1), article 74.

Greenemeier, L., 2007. China's Cyber Attacks Signal New Battlefield Is Online, *Scientific American*, September 18.

Hammervik, M., Andersson, D. and Hallberg, J., 2010. Capturing a Cyber Defence Exercise (extended abstract), *Proceedings of the 1st National symposium on Technology and Methodology for Security and Crisis Management*, pp. 30, October 27-28, Linköping (Sweden). (Note: Hammervik, M. is now publishing under Granåsen, M.)

Pilemalm, S., Andersson, D., and Hallberg, N., 2008. Reconstruction and exploration of large-scale distributed operations—Multimedia tools for evaluation of emergency management response, *Journal of Emergency Management*, 6(4), pp. 31-47.

Sanderson, P. 1991. Exploratory sequential data analysis, *Engineering Psychology Research Laboratory Technical Report EPRL-91-04*, Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, Urbana (Illinois, USA).

Sanderson, P. and Fisher, C., 1994. Exploratory Sequential Data-Analysis: Foundations, *Human-computer interaction*, 9(3-4), pp. 251-317.

Thorstensson, M., 2008. *Using Observers for Model Based Data Collection in Distributed Tactical Operations*, Linköping Studies in Science and Technology, Thesis No. 1386, Linköpings Universitet, Linköping, (Sweden).

Whetten, D., 1989. What Constitutes a Theoretical Contribution?, *Academy of Management Review*, 14(4), pp. 490-495.

AUTHORS BIOGRAPHIES

Mr. **Dennis Andersson** is a researcher at the Swedish Defence Research Agency, Division of Information Systems. He holds a master in Computer Science at Linköping University, where he is now also a PhD student in Information Science. His research is focused on capturing and sharing experiences obtained from complex exercises and operations.

Ms. **Magdalena Granåsen** is a researcher at the Swedish Defence Research Agency, Division of Information Systems. She holds a master in Cognitive Science at Linköping University, where she is also a PhD student in Information Science. Her research is focused team cognition and social media.

Mr. **Thomas Sundmark** is a software developer at the Swedish Defence Research Agency, Division of Information Systems. He holds a master in Computer Science at Linköping University.

Mr. **Hannes Holm** is a PhD student at the Royal Institute of Technology, Department of Industrial Information and Control Systems, in Stockholm, Sweden. His research is focused on methods for measuring and assessing security in computer networks.

Dr. **Jonas Hallberg** is a senior researcher at the Swedish Defence Research Agency, Division of Information Systems. He holds a PhD in Computer Systems at Linköping University. His research is focused on methods for assessment of IT security.

REALTIME CLUSTERING OF UNLABELLED SENSORY DATA FOR TRAINEE STATE ASSESSMENT

Keith W Brawner[(a)] and Avelino Gonzalez[(b)]

[(a)]Army Research Laboratory – Human Research & Engineering Directorate, Orlando, FL 32826
[(b)]University of Central Florida, Orlando, FL 32826
[(a)]Keith.W.Brawner@us.army.mil
[(b)]Gonzalez@ucf.edu

ABSTRACT

The grand challenge of Intelligent Tutoring Systems (ITS) development is that of creating a computer tutor as good as a human tutor. This difficult task may be broken into several parts. The first task real instructors perform prior to making instructional decisions is assessing the state of the trainee. Thus, the first consideration in the construction of an ITS is obtaining meaningful data from sensors and interpreting them in order to assess trainee emotional state. The interpretation of sensor data is the significant problem in this area, with the problem of sensor data mostly having been reduced to sensor selection. The machine learning methods for interpreting unlabelled sensor data are significantly more sparse than the sensors available, and their selection is far from straightforward. In this paper, Growing Neural Gas (GNG) methods, two types of incremental clustering, and Adaptive Resonance Theory (ART) will be evaluated against each other on fabricated and realtime data streams of trainees' state in order to determine the best selection of methods to accomplish this task.

Keywords: Intelligent Tutoring, Realtime Data Streams, Trainee State Assessment, Machine Learning

1. INTRODUCTION

Decisions by instructors regarding the trainee can be divided into two basic questions: "How well is the trainee doing?", and "What should be done about it?". In this regard, a computer instructor is no different than a human one. As computer-based instruction gains popularity, there is high demand for algorithms and methods to autonomously make these types of decisions. If a computer instructor can make these types of assessments as optimally as a human instructor, computation algorithms can be constructed to optimize learning goals more effectively than humans. Each of these are stepping stones for projects in self-directed, computer-enabled, automated, learning.

Computers hold an advantage over human instructors as they can make decisions based on sensory data not available to the human. For instance, humans cannot see in the Infrared spectrum, measure heart rate, measure galvanic skin response, or read Electroencephalography (EEG) activity levels. While humans still hold the advantage when processing visual and sound data, designers of ITSs should consider the advantages they have in order to design more effective instructional machines. The sensors chosen for automated trainee assessment should perform their individual tasks according to their ability, taking advantage of the computer's strengths.

This data is not without challenges. Sensory input to a computer instructor is likely to be usable in only a very narrow time window. Sensory data varies significantly from individual to individual and from day to day. Additionally, the data which is processed by sensors comes with no interpreted meaning. Because of the high variation in the data, a constructed model is not likely to be useful long term, and models constructed with this data must be built quickly and not reused. The difficulty in the creation of a unified, emotional model for affective sensing is a large part of why emotional sensors have not been integrated into the ITS domain (Arroyo, Cooper, Burleson, Woolf, Muldner & Christopherson, 2009).

Two traditional methods for dealing with this problem are reprocessing by incremental rebaselining, in the case of 'microclusters' (Aggarwal & Yu, 2008), or averaging over a Hamming window (Papadimitriou, Brockwell & Faloutsos, 2004). Reprocessing based on a new, changed, data stream appears to be a simple solution, but only creates more problems. Issues of reprocessing include: when to reprocess, determination of when data shift is too extreme to be usable, and the reprocessing window being a time when decisions cannot be made on that type of data. These issues point to the idea that reprocessing will not solve the general problem. Using data in a window presents its own problems, but it is generally accepted that usable conclusions can be conferred via windowed data (Hore, Hal & Goldgof, 2007).

The solution to the generalized problem lies with machine learning classifiers that are able to adapt to changes in a data stream in realtime. These methods build a model of the data seen so far, use it to make decisions, and discard, log, or internalize it after a training session is complete. Using this type of solution provides an accurate assessment of trainee state, but creates the problem of dealing with a realtime data stream.

2. REALTIME DATA STREAM PROBLEMS

Processing realtime sensory data is becoming increasingly easier as computation gains speed and takes advantage of multiple cores. Even though computers are becoming faster, the primary trouble with data streams does not lie in the speed of computation. Algorithms dealing with realtime data streams must deal with the critical issues of potentially infinite length, concept detection, concept evolution, and concept drift (Beringer & Hüllermeier, 2006). Each of these presents their own problems for classification.

Sensory data is inherently unlabelled. Sensor measurements, while accurate, do not natively imply anything about the item which they are measuring. Just as a nuclear engineer can use a temperature sensor to measure the temperature of a reactor core, the sensor does not tell him the implications of that temperature. An individual, or algorithm in place of an individual, must give meaning to the collected data. In the same fashion, measurements of heart rate variability in a trainee are not interpreted for their implications on learning and training. However, asking the trainee their state at each heartbeat, second, or minute is not only impractical, but a task that distracts from learning. Labels for state data must be computer-generated, infrequently polled from the user, or a combination of both.

2.1. Infinite Length

The fundamental problem when dealing with realtime data streams is the potentially infinite length. Any machine learning algorithm that relies on viewing all data to make a decision at current time will fail to process a potentially infinite stream. Although the data stream is not infinite in the real sense, the practical implication of infinite length is the presence of a point of diminishing returns. For any algorithm that depends on the presence all data to make a decision there is a point where there is too much data to process and make a decision before more data is presented. The implication at this point is that the algorithm can no longer perform in realtime. While certain algorithms may be used up to a hundred, thousand, or million data points the presence of the limiting number is not recoverable. The infinite length problem immediately discounts the use of traditional systems of machine-learning such as Bayesian Networks. Training algorithms that iterate over all previous inputs, such as backpropagation with Artificial Neural Networks (ANNs), also must be discarded.

As an example, in the domain of trainee state assessment from sensor streams, the duration of a given scenario-based training event is unknown and the sampling rate can be over 500 Hz with two dimensions.

2.2. Concept Detection

The second problem in dealing with realtime data streams is the presentation of a new, previously unknown, class of data. Because there is no fundamental limit to the types of observations, there must be a method for detection of the novelty. This makes many traditional, unmodified, machine learning methods impractical, such as Support Vector Machines and Decision Trees, which are unable to dynamically respond to the development of new, previously unseen, classes of data.

In the domain of interest, the number of moods or their representation is not known prior to observation. Even if the number of states is known *a priori*, there is no guarantee that any of the states will be represented in a given period, that one state will correspond to one location within the observational space, or that one state will correspond to only one location of the sampling space.

2.3. Concept Drift

The third issue when dealing with realtime data streams is the problem of the underlying concepts changing over time. In the domain of interest, a particular mood of a trainee can represent itself differently over time. For instance, in an unpublished Galvanic Skin Response (GSR) classification experiment, spikes were observed in a GSR data stream of people exposed to fear-based imagery. However, after the spike had been resolved and the image had been rescinded, both the next rest state and next image presented demonstrated elevated GSR measures. This is not indicative of the subject evidencing strong emotion in the rest state or stronger emotion in response to fear-based imagery, but instead evidence of a change in baseline observation. This further rules out ideas such as Hamming window baselining as a possible solution. The experiment that dealt with this datastream evidenced poor performance using NeuroEvolution of Augmenting Topologies, which has elements of ANNs and Genetic Algorithms, to attempt classification. This indicates that such methods would not be successful, even with realtime modifications. A 2010 study, pictured below, found similar observance patterns among GSR data.

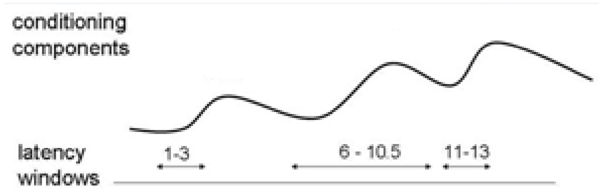


Figure 1: Example of concept drift changing baseline measurements in one-dimensional, time-aligned, GSR data (Gao, Raine, Venables, Dawson & Mednick, 2010)

2.4. Concept Evolution

The final problem present in realtime data streams is concept evolution. This is observed when an already evolved concept presents itself in a way different from before. In the domain of text mining, this is presented by using different words to express the same concept (Masud et. al. 2010). In the domain of trainee state modeling, this may represent itself as a changed baseline in one dimension and a different baseline in another; however, it is unknown what types of sensory measurements are likely to experience the evolution of one concept differently in the represented tested sampling space.

3. POTENTIAL SOLUTIONS

The nature of this problem lends itself to several branches of machine learning. Unsupervised methods can be used in order to cluster inherently unlabelled sensory information. Information can then be attached to the clusters in a semi-supervised manner to give meaning to the data. For instance, unsupervised methods can point out general data trends and use another algorithm, or infrequent survey data, to give these established categories various implications. Supervised or statistical Machine Learning can process clusters with attached meaning to recommend instructional decisions for learning goals or outcomes.

The components of this section will summarize algorithms that will be tested against the first portion of the adaptive tutoring problem; continuous, realtime, noisy, sensor data.

3.1. Incremental Clustering

Clustering is a type of unsupervised learning operating on the principle of determining the distance from an observed point in sample space to other points relative to it. If the observed point is within a threshold distance of similarity to other points, it is assumed to have a relationship to them and is clustered together. Examples of clustering include hierarchal clustering, graph clustering, and k-means (Jain, 2008).

Two types of incremental clustering algorithms are used for this paper as a baseline measurement of performance. Both of these are k-means variants adapted for incremental use on data streams of potentially infinite length. The performance of these algorithms is expected to flag on the problems of drift and evolution, but provides a reasonable baseline approach upon which to compare the other methods presented.

The first clustering algorithm used in this paper is adapted from http://gromgull.net, and closely follows the approach of online agglomerate clustering (Guedalia., London & Werman, 1998). This approach is summed up in their work as, for each point:

Move the closest centroid towards the datapoint
Merge the two closest centroids, if appropriate
    Creates one redundant centroid
Set redundant centroid equal to the datapoint

The second method is custom-developed, but follows the basic point recognition algorithm pseudo-code:

For each point
    Compare point to all known clusters
    If no cluster is within vigilance
        create new cluster here
    else
        move matched cluster up to <delta> in the direction of the recent point

3.2. Growing Neural Gas (GNG)

Neural Gas is a method of finding the optimal data representations based on feature vectors. It was inspired by the successes of ANNs and Self Organizing Maps, and has been used as an alternative to clustering. An incremental version has been created in the form of GNG (Holmstrom 2002), and is expected to outperform baseline clustering on the problems of interest. The general algorithm is below, for each new data point, although the specific implementation in that experiment may be slightly different, as the Modular toolkit for Data Processing (MDP) 2.6 implementation was used:

If appropriate (current point does not correspond to known information), create new reference arc, store error
Else increment age of all arcs in this area, move existing arcs towards new data, establish new ages for arcs
Remove Aged arcs
If any non-emanating arcs exist, remove them
If it is the time to add a new point (due to timing)
    Add a new reference point, halve the distances of the existing arcs to this point, scale the existing errors
Compute non-Hamiltonian path of all arcs (depth first)
For this point against each class:
    If there are few related nodes, compute the probability of the point belonging to the lowest error class
    Else determine the modified shape of the cluster it is most likely to belong to

3.3. Adaptive Resonance Theory (ART)

ART is a type of neural network architecture which classifies objects based on the activation of nodes in the structure. It was developed to classify data in a one-pass learning environment (Carpenter & Grossberg, 1995), and has performance roughly equivalent to neural networks with significantly reduced training time. In its most basic form ART draws n-dimensional hypercubes around similar input patterns, where n is the dimension of the input data. Matched data is the data that falls within the smallest hypercube or of the class of the closest available hypercube. Hypercubes are expanded to compensate for new data in accordance with parameter settings. Although sometimes viewed as a disadvantage, ART systems are capable of one-pass learning, and are consequentially sensitive to the input order of data. It is expected that ART systems will respond well to the problem of evolution, but poorly to the problem of drift.

3.4. Parameters used in this experiment

The parameters used in this experiment for online agglomerate clustering was a maximum cluster number of 15. The incremental k-means approach used a delta parameter, corresponding to a numeric value of how much to change in response to a new point, of 10%. It also used a vigilance parameter, corresponding to a maximum percentage error before creating a new cluster of 20%. These values were chosen based upon minor experimental sampling.

The ART algorithm used in these experiments was complement coded, with no category maximum, fast learning, a vigilance of .9, and a bias of .000001. More information about ART parameters can be found in the original paper (Carpenter & Grossberg, 1995) or about this specific implementation via the pyrobot users group.

The GNG algorithm used in this paper is implemented as part of the Open Source Modular Toolkit for Data Processing, with the following parameters: alpha = .1, lamda=20, eps_beta=.5, max_nodes=400, max_age=10, epilon_n=.06, d=0.995. These all all roughly recommended parameter settings. More information about these parameters can be found in the MDP 2.6 website: http://mdp-toolkit.sourceforge.net/, or in the published paper (Zito, Wilbert, Wiskott & Berkes, 2009).

4. PROBLEM SPACE

In order to test each algorithm, several datasets were obtained. The benchmark datasets include: a fabricated two dimensional set of predefined shapes, a fabricated two dimensional point-drawing of shapes representing movement from one class of data to the next, and a two dimensional real dataset of electrocardiology (ECG) and GSR data taken during an experiment, and a three dimensional real dataset of EEG-classified trainee state.

4.1. Ordered set of point-drawn shapes

A set of predefined shapes will be chosen to represent the problem space of generic, unsupervised, classification. The points are presented in order of shape appearance in order to establish how well each classification algorithm responds to the development of new data classes over time. Each shape is Gaussianly distributed among its boundaries with the exception of the outlined circle, which has a Gaussianly distributed ring. Points are drawn between shapes in order to simulate the gradual change of classes. This more accurately simulates the trainee state space by not providing drastically different shift. Although there is no correct answer to this dataset, the general classification of observed shapes should be a goal. While these are quickly apparent to human eyes, it is important to note that these patterns are presented to each algorithm with very little memory. This dataset tested 4250 unique points.

4.2. Randomized set of point-drawn shapes

This set uses the same set of datapoints as described in section 4.1, with the order of presentation randomized. This tests how well each algorithm remembers the states that it has previously seen. As in 4.1, this dataset tests 4250 unique points.

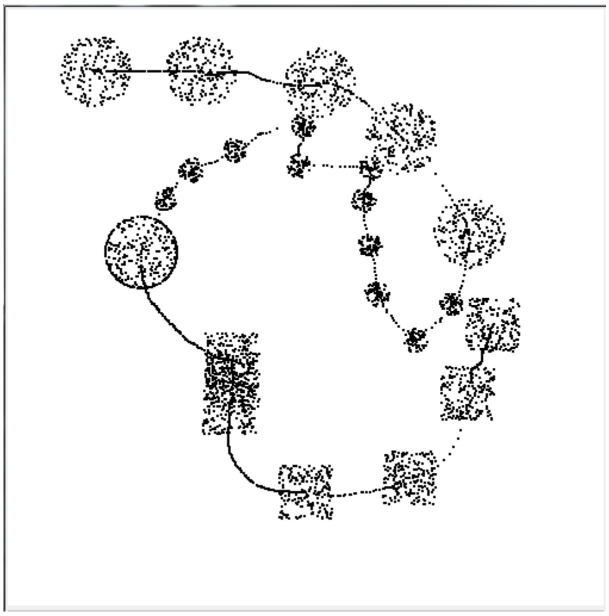


Figure 2: A set of predefined shapes, used in shape order for dataset 4.1, and used in randomized order for dataset 4.2

4.3. Feature-extracted EEG data

A set of ECR and GSR data captured in another, currently unpublished, experiment is available to the author. The experiment involved a user interacting with a digital character in well-, and ill-defined scenarios. One of the three scenarios was deliberately made frustrating for the user in order to evince particular emotional state. A random user was selected for the use of their data, in order to gauge an idea of performance on other, similar, datasets. Technologies of eMotive EEG, Galvanic Skin Response (GSR), and electrocardiology (ECG) data were captured in this experiment. It is expected that the user demonstrated various cognitive states during the various phases of the experiment. These were, in order, a rest period, a scenario which was chosen randomly among three possible, and a survey. The performance on this set of data can be used to gauge an idea of performance on other, similar, datasets. This dataset used 9,342 points, corresponding to 38 minutes of actual data collection time.

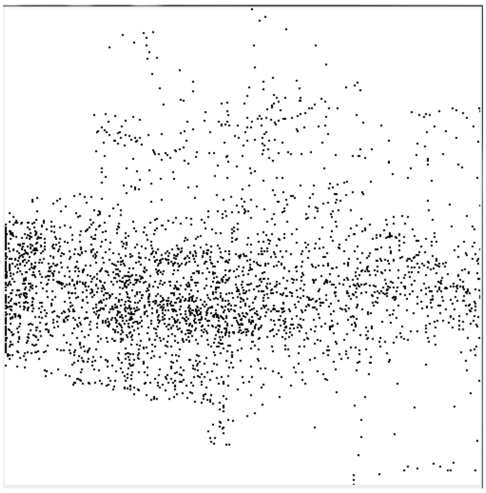


Figure 3: Classified EEG data. Short Term Excitement (STE) is along the x-axis, Boredom/Engagement is along the y-axis.

4.4. ECG and GSR data

In the experiment described above, the ECG and GSR data was collected as an additional measurement of trainee state. This dataset tested 400,000 datapoints, corresponding to 24 minutes of actual data collection time.

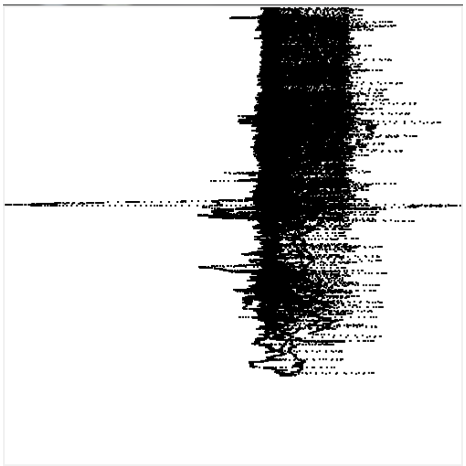


Figure 4: Unfiltered ECG and GSR data. ECG data is along the x-axis, GSR data is along the y-axis

5. RESULTS

5.1. Set of ordered, predefined shapes

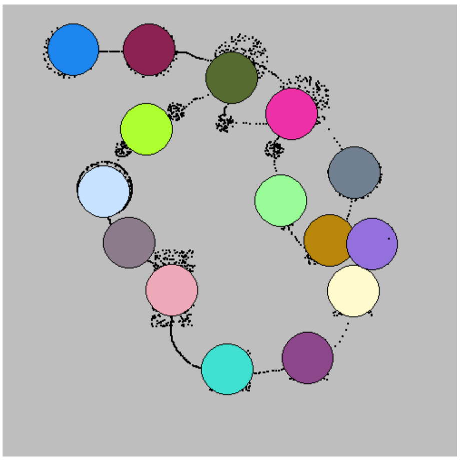


Figure 5: Online agglomerate shaped clustering

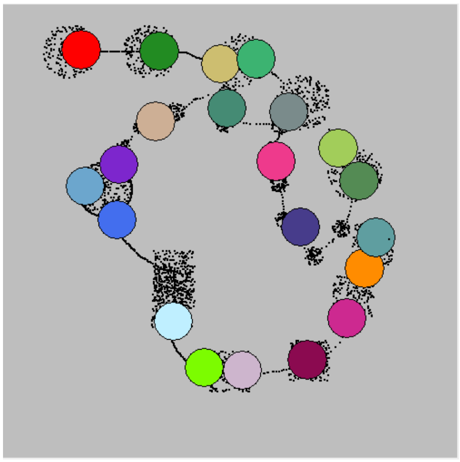


Figure 6: Incremental k-means shaped clustering

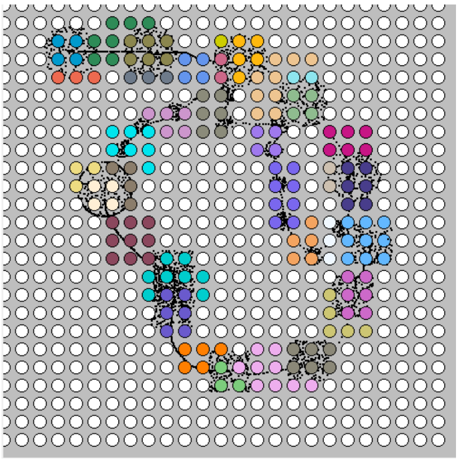


Figure 7: ART shaped clustering

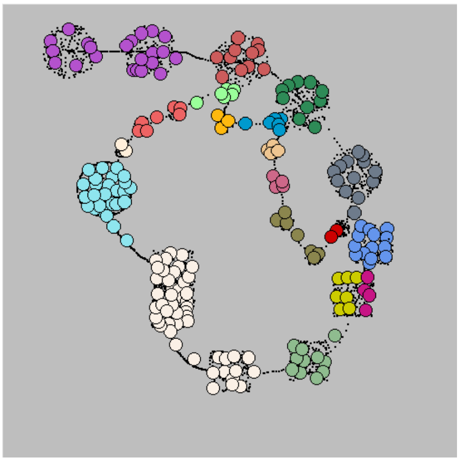


Figure 8: GNG shaped clustering

5.2. Set of random, predefined shapes

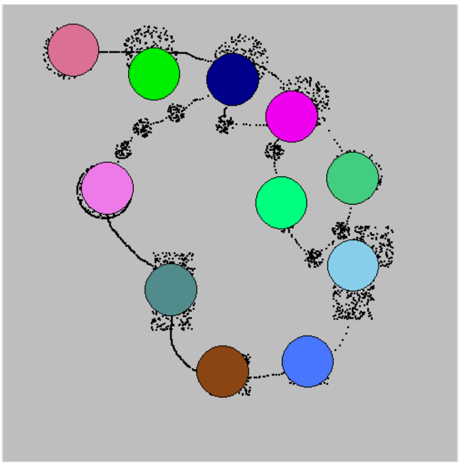


Figure 9: Online agglomerate random shaped clustering

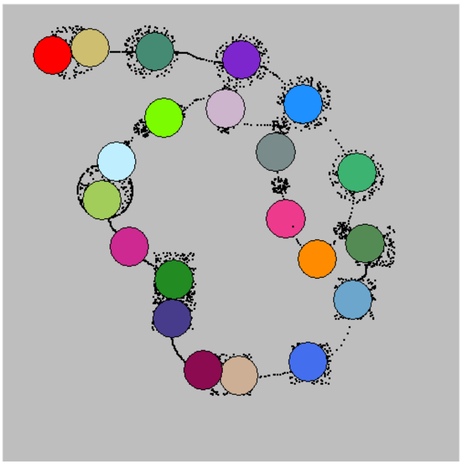


Figure 10: Incremental k-means random shaped clustering

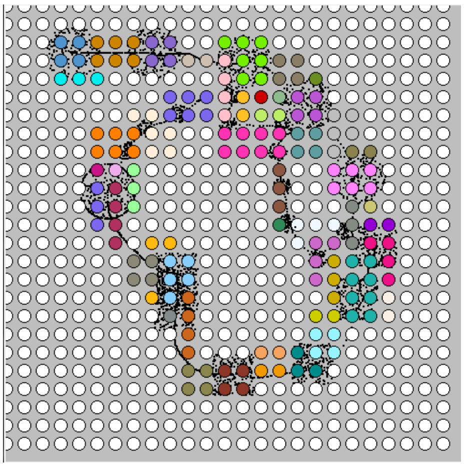


Figure 11: ART random shaped clustering

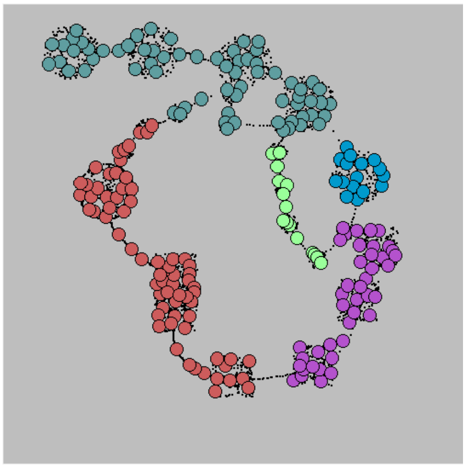


Figure 12: GNG random shaped clustering

5.3. Classified EEG data

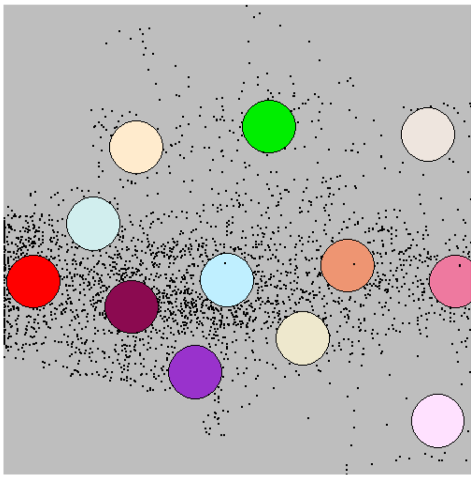


Figure 13: Online agglomerate EEG clustering

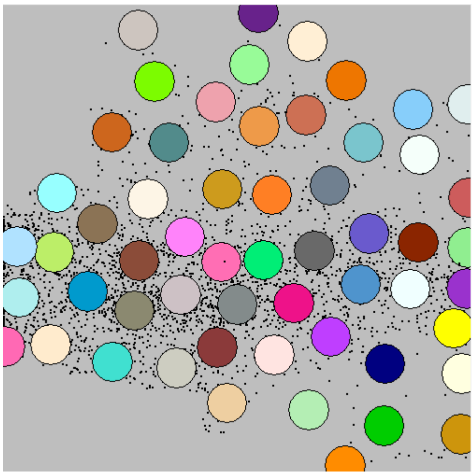


Figure 14: Incremental k-means EEG clustering

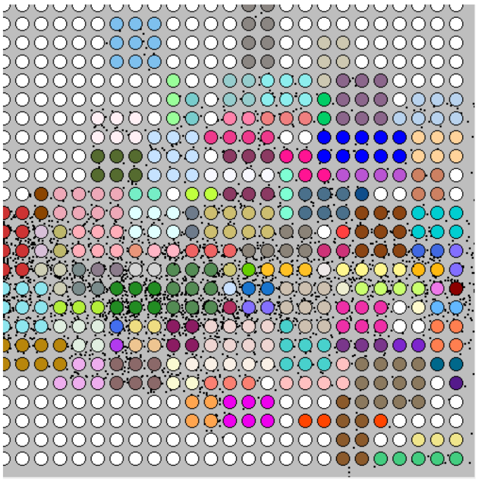


Figure 15: ART EEG clustering

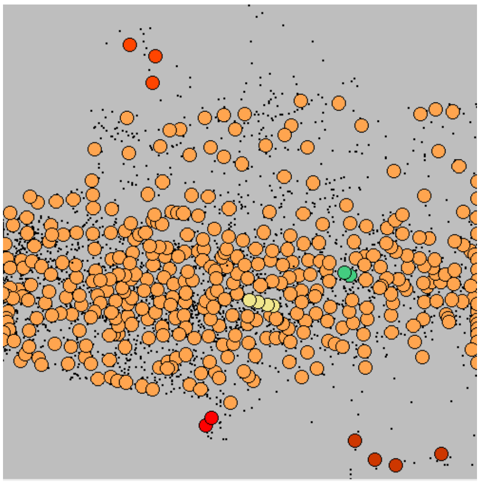


Figure 16: GNG EEG clustering

5.4. ECG and GSR data

Online agglomerate clustering did not complete this successfully within allotted time, and failed to deal with the problem of infinite length.

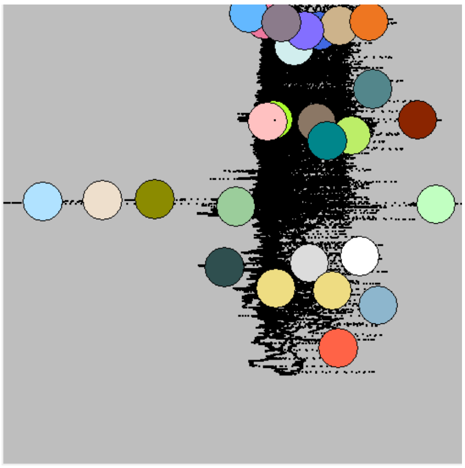


Figure 17: Incremental k-means ECG/GSR clustering

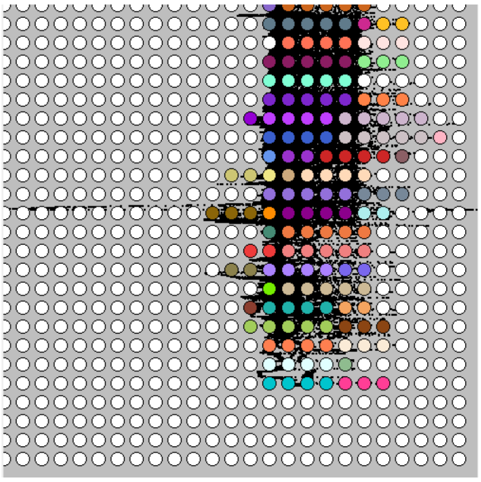


Figure 18: ART ECR/GSR clustering

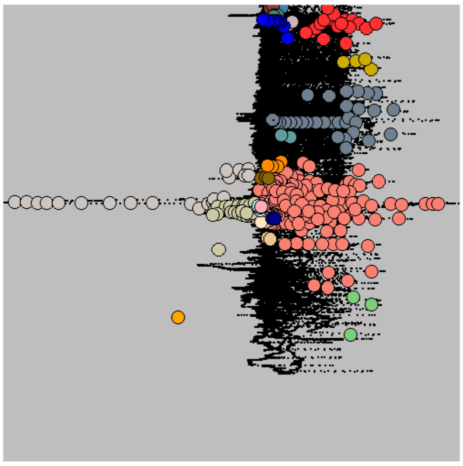


Figure 19: GNG ECR/GSR clustering

5.5. Time Analysis

All experiments were run, single-threaded and overnight with no interference on a 2.67 GHz processor with 4GB of RAM. Data was cached before clustered. Time was stopped for screen-drawing so as not to bias algorithms, such as ART and GNG, where data visualization is a lengthier. Python version 2.7 was used to handle garbage collection and other programming issues. Integrated DeveLopment Environment (IDLE), a part of Python, was run in a thread separate from processing. Given the high data processing rate of these algorithms, it was determined that the average time over multiple runs would not be required to be analyzed. However, these numbers should be taken as representative rather than guaranteed.

Table 1: Time analysis of various algorithms (in Hz)

	Online agglomerate clustering	Incre. k-means	ART	GNG
Ordered shapes	215	69672	11707	1396
Unordered Shapes	273	27243	4560	1403
EEG	189	15648	3803	801
ECG/GSR	-	45511	2035	472

6. DISCUSSION

6.1. Ordered Shapes

It is worth noting early that the size of the clusters in each of the first two algorithms are not a function of the true size, which is difficult for visualization purposes. However, the overall trend can be seen, so this is of little concern. In a time analysis, incremental k-means outperformed all other methods, with generally good responses from ART and GNG.

Agglomerate clustering does an excellent job of capturing the major data clusters, while missing the smaller clusters, and generally maintaining a fairly stable number of clusters. It hits the fifteen cluster limit, but at a generally acceptable time. Hitting this limit causes it to miss several of the smaller circles, and it classifies these as generally transitory states.

K-means, while correctly classifying many of the smaller clusters, and correctly classifying the larger clusters, misclassifies the edge-bound circle and two of the squares as a few different classes. It cannot merge these due to the deficiency of the algorithm.

ART does a successfully captures the meaning in the smaller clusters, but generally over-classifies the larger shapes. GNG, however, classifies the obvious shapes with very little error and acceptable time performance.

6.2. Randomized Shapes

In general, the performance of the incremental clustering algorithms did not degrade with the randomization of the data points. ART, however, narrows in the clusters that it is classifying, while GNG nearly completely fails to isolate individual classes of underlying structure. This behavior of the GNG algorithm is noted in other locations (Ancona, Ridella, Rovetta & Zunino, 1997).

6.3. EEG

The EEG dataset has no easily apparent patterns, but it does have locations of more data presentation instead of less. Online agglomerate clustering fails to find the patterns in the striated middle bands, as does k-means, and GNG. ART, however, classifies x-dimensional bands, which are likely to be the observed states. Additionally, outliers, which are likely to be noise, are classified into a classes of outliers in GNG and ART, but not in the online clustering algorithms.

6.4. GSR/ECG

The final dataset deals with a large amount of data to be processed, and is indicative of the worst-case, real-life situations where these algorithms may be used to classify trainee state. This data was collected at a sampling rate of 500 Hz, which puts a significant speed test on each of these algorithms. Online agglomerate clustering fails to finish classification in time. GNG also fails to complete in time, but it is worth noting that this is only by a slim margin of 472 Hz, rather than 500.

With additional speedups found in high performance computing, or simply a faster processor, it is possible that this algorithm would finish in time. However, it is worth noting that any real-life usage will likely be in more than two dimensions.

This is the second set of data with no readily determinable pattern. Incremental k-means expresses this as a series of unmerged clusters, while ART expresses this as roughly 20 striated bands. ART completely fails to capture the heartbeat, which is the most important part of the ECG data. Further experiments, which are not shown, show this to be true even at 5 times the imaging resolution.

7. CONCLUSION

The obvious conclusion, as demonstrated in the No Free Lunch Theorem (Wolpert & Macready, 1997), is that no one algorithm works best across all problems. The incremental clustering algorithms performed admirably across the domains of well-defined clustering, while performing quite poorly on real-world data. One of the important takeaways here is that cluster merging is too expensive of a proposition to allow in realtime.

Generally speaking, ART was able to classify well in realtime even when data patterns were not present or not apparent. The clusters formed by ART were of reasonable input size and classification. GNG was able to successfully classify provided that data was in the correct order and that a pattern was apparent, but was a bit slow on problems of high data input rate.

8. FUTURE WORK

Future work in this area will branch into two directions. The first direction will continue to look for algorithms that perform well on this type of data problem, or in the adaption of ART and GNG to classify fewer clusters, and operate with a quicker timeframe, respectively. Feature extraction of meaningful data will also be looked into, including the addition of time-stamp related data and its effect on formed clusters, which would appear to have an effect on other assortments of data (Beringer & Hüllermeier, 2006).

The second direction will look at the problems of how to model trainee state, state transitions, and future state prediction. Simultaneously, the author and others will look at how to present delayed reward to the machine learning algorithms which make decisions based on trainee states. It is expected that these studies will involve Markov Chain models, Markov Decision Processes, and Partially Observable Markov Decision Processes. Experiments of what constitutes a reward will also be conducted.

ACKNOWLEDGEMENTS:

The lead author would like to thank the Army's Simulation and Training Technology Center and University of Central Florida for continued support and development. Additionally, he would like to thank gromgull, D.S. Blank, Aaron Garrett, and Terry Stewart for parts of the software or software basis used in the above detailed algorithms.

REFERENCES:

Aggarwal, C., and Yu, P. S. Framework for clustering uncertain data streams. In *IEEE International Conference on Data Engineering*, 2008.

Ancona, F., Ridella, S., Rovetta, S., and Zunino, R., "On the importance of sorting in Neural Gas training of vector quantizers," in Proc. Int. Conf. Neural Networks'97, Houston, TX, June 1997, vol. 3, pp. 1804–1808.

Arroyo, I., Cooper, D.G., Burleson, W., Woolf, B.P., Muldner, K. and Christopherson, R. (2009a) 'Emotion sensors go to school', *Proceedings of the International Conference on Artificial Intelligence in Education*, IOS Press, July, pp. 17–24.

Beringer, J., Hüllermeier E., Online clustering of parallel data streams, Data Knowl. Eng. 58 (2) (2006) 180 – 204.

Carpenter, G.A., & Grossberg, S. (1995). Adaptive resonance theory (ART). In M. Arbib (Ed.), The handbook of brain theory and neural networks. (pp. 79-82). Cambridge, M.A.: MIT press.

Gao, Y., Raine, A., Venables, P.H., Dawson, M.E., Mednick, S.A., 2010. The development of skin conductance fear conditioning in children from ages 3 to 8 years. Dev. Sci. 13 (1), 201 –212.

Guedalia, I. D., London, M., Werman, M.. 1998. An on-line agglomerative clustering method for non-stationary data. Neural Computation. Citeseer.

Holmstrom, J. (2002). Growing neural gas. Unpublished Master's Thesis, Uppsala University.

Hore P, Hall LO, Goldgof DB. A fuzzy C means variant for clustering evolving data streams. IEEE-SMC, 2007

Jain AK (2008) Data clustering: 50 years beyond k-means. In: ECML PKDD '08: Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases - Part I, Springer-Verlag, Berlin, Heidelberg, pp 3–4

Masud M., Chen Q., Khan L., Aggarwal C., Gao J., Han J., Thuraisingham B., Proceeding of the 2010 IEEE International Conference on Data Mining, IEEE Computer Society, 2010

Papadimitriou, S., Brockwell, A., Faloutsos, Ch.. Adaptive, unsupervised stream mining. The VLDB Journal, 13(3), 2004.

Wolpert, D.H., Macready, W.G. (1997), "No Free Lunch Theorems for Optimization," *IEEE Transactions on Evolutionary Computation* 1, 67.

Zito, T., Wilbert, N., Wiskott, L., Berkes, P. (2009). Modular toolkit for Data Processing (MDP): a Python data processing frame work, Front. Neuroinform. (2008)

AUTHORS BIOGRAPHIES

Keith Brawner is a researcher for the Learning in Intelligent Tutoring Environments (LITE) Lab within the U. S. Army Research Laboratory, Simulation and Training Technology Center (ARL STTC). He has five years of experience within U.S. Army and Navy acquisition, development, and research agencies. He holds a M.S. in Computer Engineering with a focus on Intelligent Systems and Machine Learning from the University of Central Florida, and is currently in pursuit of a doctoral degree in the same field. The focus of his current research is in machine learning, adaptive training, trainee modeling, and knowledge transfer. Dr. Avelino Gonzalez is his advisor.

Dr. Avelino Gonzalez received his B.S. and M.S. degrees from the University of Miami, Coral Gables, FL, in 1973 and 1974, respectively, and his Ph.D. degree from the University of Pittsburgh, Pittsburgh, PA, in 1979, all in electrical engineering. He is a Professor in the Department of Electrical Engineering and Computer Science at the University of Central Florida, Orlando, specializing in artificial intelligence and simulation.

THE IMPACT OF STUDENT EXPECTATIONS AND TUTOR ACCEPTANCE ON COMPUTER-BASED LEARNING ENVIRONMENT ACCEPTANCE AND FUTURE USAGE INTENTIONS

Heather K. Holden [(a)] and Benjamin S. Goldberg [(b)]

U.S. Army Research Laboratory – Human Research and Engineering Directorate
Simulation and Training Technology Center
Learning in Intelligent Tutoring Environments (LITE) Laboratory

[(a)]heather.k.holden@us.army.mil [(b)]benjamin.s.goldberg@us.army.mil,

ABSTRACT

Computer-based tutoring systems should be designed to meet/exceed learners' expectations and maximize learners' technology acceptance to increase their satisfaction and future system usage intentions. Pedagogical Agents (PAs) are commonly incorporated into an Intelligent Tutoring System (ITS) to increase learners' satisfaction; however, empirical evidence on the impact of PAs on learner acceptance is limited. Additionally, few ITS researchers seek to understand learners' expectations and acceptance of both the agent and learning environment. This paper presents the results of a study that evaluates the relationships between learner expectations and acceptance of a PA and learning environment both before and after learners' system interactions.

Keywords: technology acceptance, technology expectations, Pedagogical Agents (PAs), Intelligent Tutoring Systems (ITS)

1. INTRODUCTION

Technology acceptance is a critical driver that influences user interactions with computer-based training platforms. The capability for a system to meet user expectations increases the probability of high user acceptance and promotes efficient knowledge transfer. Most research in the field of computer-based training lacks emphasis in understanding a trainee's expectations on the individual level and primarily focuses on performance outcomes.

ITSs are generally designed to produce significant learning gains; however, they are not designed with the intent to increase students' interest and motivation for future interactions. Although previous research shows assessment of users' perceptions and attitudes towards a technology, evaluation measurements of student expectations prior to ITS interaction is uncommon (Jackson, Graesser, and McNamera, 2009). This is important because previous work has shown initial technology expectations to significantly influence student perception of subsequent interactions (Jennings, 2000; Lindgaard, Ferrnandes, Dudek, and Brown, 2006; Jackson et al., 2009). Furthermore, students' expectations may increase as the perceived intelligence of a technology increases, and this relationship needs further awareness (Foner, 1997; Norman, 1994; Jackson et al., 2009).

In lieu of this issue, some ITS researchers incorporate PAs to the training interface in an effort to establish a virtual training companion to better facilitate the learning process. PAs provide a medium for delivering instructional content and feedback/support, and are incorporated to increase trainee satisfaction and acceptance of the learning experience.

Over the last decade, research has aimed to understand the different characteristics associated with a PA and how they impact training interactions. Empirical results have shown the presence of a PA alone can increase trainee self-efficacy, attitudes, and satisfaction with the learning environment (Junaidi 2007; Kim, Wei, Xu, Ko, & Ilieva, 2007; Rosenberg-Kima, Plant, Baylor, & Doeer, 2007; Fatahi and Ghasem-Aghaee 2010). However, the literature in this domain has shown an absence of evidence explaining the relationships between student acceptance, perceptions, and expectations of both the agent and learning environment.

This paper presents the results of a study evaluating expectations and technology acceptance associated with a PA and the learning environment it's embedded within. The experimental testbed trained participants on the rules and strategies for playing Sudoku. Measures were taken to assess system and agent expectations and acceptance both prior to system exposure and directly following interaction. Four experimental conditions were designed with the PA displaying varying degrees of competence and emotional support during the training experience. This study aims to answer what are the common user expectations when interacting with an instructional PA and how those expectations are met when a PA exhibits varying degrees of support. This is important because failing to meet minimal expectations for a user can have a negative effect on cognition and will reduce training effectiveness.

Additionally, this study determines if there is a significant relationship between affect (mood) to student expectations and acceptance. Investigation of the influences of students' expectations could enable ITS researchers and developers to potentially better understand: (1) differences in performance and system evaluations; (2) possible affective, cognitive, and motivational restrictions of ITS usage; (3) factors which attribute to future usage intentions; and (4) methods to appropriately design and adapt instructional strategies to increase user acceptance and exceed user expectations.

2. RELATED LITERATURE

2.1. Expectations and Acceptance of a Computer-Based Tutor

An ideal trainee-PA relationship emulates the same benefits as the human relationship seen in one-to-one tutoring (Bloom 1984). A central component of human-to-human tutoring is social interaction. Teaching and learning are highly social activities, which attribute to trainees' cognitive and affective development (Kim and Baylor 2006). Social interaction builds trust, thereby strengthening the relationship between the instructor and trainee, and in turn influences motivation for learning (Baylor 2000).

Social interaction in a tutoring environment is driven through communication, which is the primary medium a PA can utilize to establish trainee trust. Lee, Ahn, & Han (2006) defines trust as the product of three dimensions: ability, benevolence, and integrity. Core, Traum, Lane, Swartout, Gratch, van Lent, & Marsella (2006) considers trust as a linear combination of solidarity, credibility, and familiarity. Research has also found a significant connection of users' trust to their technology acceptance (Lee et al., 2006; Cho, Kwon, & Lee, 2007). It is hypothesized that the presence of a PA alone will increase trainees' self-efficacy, attitudes, and satisfaction of system interaction.

In addition to trust building, social interaction influences motivation and learning. A trainee who enjoys interaction with a PA will demonstrate a more positive perception of the overall learning experience and be more accepting to the learning environment (Johnson, Rickel, & Lester, 2000; Junaidi, 2007). Thus, theoretically the more trust a trainee has in the PA technology, the more trust he/she will have in the training experience. However, there is a lack of empirical evidence supporting this claim.

2.2. Expectations and Acceptance of an Intelligent Tutoring System

ITS researchers usually use measures of students' computer experiences and usage patterns to gauge students' attitudes and perceptions towards an ITS; however, research has found that such measures are insufficient indicators (Garland and Noyes, 2008). Research focusing on information technology acceptance typically uses the original or a revised Technology Acceptance Model (TAM) to gauge users' acceptance. The TAM is a theoretical model which predicts how a user comes to accept and use a given information technology by interpreting responses on cognitive, affective, and behavioral measures. It specifies causal relationships among external variables, belief and attitudinal constructs, and behaviors captured during initial system interactions (Hubona & Kennick, 1996). The model suggests that when users are presented with a particular information technology, a number of factors, notably *perceived usefulness* and *perceived ease of use*, influence their decision of how and when they will use the technology. These cognitive responses then influence the users' affect (attitude) towards using the technology, which ultimately drives their behavioral intentions (Burton-Jones & Hubona, 2005).

The key limitation to TAM is its inability to account for measures on system expectations prior to interaction with the system. Jackson et al. (2009) discovered that students' prior expectations of a technology's capability have a large effect on students' initial and post-levels of motivation and technology familiarity. Furthermore, meeting expectations can significantly increase the likelihood that students' will use the system again in the future. However, few if any, ITS researchers evaluate students' ITS acceptance.

3. METHODOLOGY

A 2x2 experiment was designed assessing the effect competency and emotional support has on meeting user expectations and system acceptance. The PA was given one of two definitions for each variable of interest: high and low. Participants were randomly assigned to one of four experimental conditions, each instantiating varying degrees of PA competency and PA emotional support. The PA was embedded into a testbed (i.e. learning environment) that teaches subjects the rules and strategies of Sudoku. The environment was designed with the ability to cater to any level of previous experience.

This study anticipates the following: (a) Students' assessment of a PA's qualities will have a strong, positive relationship to their acceptance of the learning environment; (b) The perceived notion that the PA is trusted by the learner will vary in relation to the participant's prior experience within the domain; and (c) The PA condition experienced by the participant will have a direct effect on reported Self-Assessment Mannequin (SAM) mood dimensions; and (d) PA's exhibiting either emotional support, competence, or both will produce significantly higher knowledge gains than a PA's that provides neither quality.

The experimental learning environment was developed with Microsoft Visual Basic.NET Express Edition. The PA embedded within the learning environment is the Microsoft Agent Audie, who is a representation of an animated computer. Figure 1 presents a screenshot of the experimental learning environment.

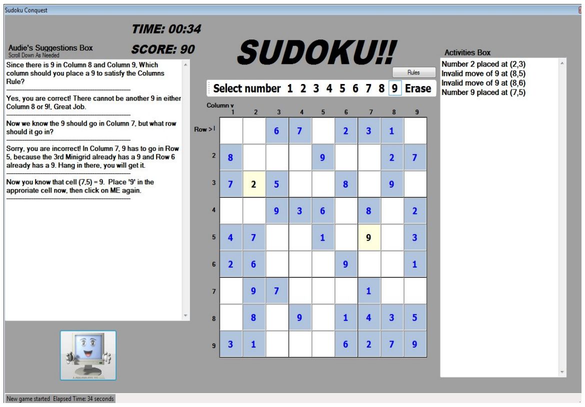


Figure 1: Sudoku Learning Environment Screenshot

3.1. Sample Population

Although participation in this experiment was open to the general community, the population for this study was a sample of convenience. Thirty-five volunteers participated in the study. The participants consisted of 22 males and 14 females between the ages of 19 and 63. Eighty-six percent of the sample reported having advanced computer experience, and ninety-one percent of the sample believed that computers can help them learn difficult concepts.

3.2. Procedure and Instrumentation:

After obtaining informed consent, participants completed a pre-experiment survey. The instrument collected information on demographics (i.e. age, gender, and education); mood (Self-Assessment Mannequin (SAM) Lang, 1985); expectations of a computer-based tutor; and expectations of a computer-based learning environment. Expectation measures associated with the PA were collected from the 12-item Attitude toward Tutoring Agent Scale (ATTAS: Adcock & Van Eck, 2005; Jackson et. al, 2009). The instrument assessing participant expectations of a computer-based learning environment consisted of 9 questions adapted from Davis (1989) and Holden & Rada (2011). This construct is composed of seven items on Perceived Ease of Use (PEU), one item on Perceived Usefulness (PU), and one item on future Usage Intentions (UI). For both expectation constructs, participants were asked to rate the importance of each item on a Likert scale from 1 (not at all important) to 9 (totally important) and to rank the order from least important to most important.

Following the pre-experiment survey, participant interaction with the learning environment was initiated. The instructional sequence was divided into four phases: a Sudoku tutorial, an interface tutorial, game 1, and game 2. All phases were guided by an assigned conditional PA reflecting the appropriate variance of competency and emotional support, except during the interface tutorial where the PA exhibited the same habits for all four experimental conditions. For novice and beginner Sudoku players, participants were given an easy Sudoku puzzle for the first game, while intermediate and experts were given a medium puzzle for the first game. A hard Sudoku puzzle was given for the second game across all groups. Participants had 15 minutes to complete as much of the 9x9 grid as possible.

Following, participants' completed a post-experiment survey. The survey inquired about participants' mood (SAM); perceptions of the PA tutor; and perceptions of the learning environment. The perceptions of the PA and learning environment constructs consisted of the same measures from the expectations constructs in the pre-experiment survey. However, participants were asked to rate their agreeableness for each item on a Likert scale from 1 (Strongly Disagree) to 9 (Strongly Agree). These constructs measure participants' acceptance of the PA and learning environment.

4. RESULTS

4.1. Pre-Interaction Analysis:

Survey data was collected prior to system interaction to gage experience with the domain and expectations towards computer-based tutoring. This data was analyzed as a whole to identify trends and similarities associated with PA implementation. In regards to previous Sudoku experience among participants, 31% reported no knowledge or familiarity of the domain, 31% reported basic experience initially, and 37% reported having advanced experience. Participant interest and motivation was also assessed. Of the sample, 86% were excited about learning new topics, 65% were interested in increasing their Sudoku knowledge, and 86% were motivated to participate in the experiment.

Additionally, participants were instructed to rate and rank the importance of their ideal computer-based tutor qualities. These items (derived from ATTAS) were used to gage initial expectations of a PA. Items were based on qualities/characteristics associated with human tutors. Table 1 presents the results of the overall sample.

Table 1: Initial Expectations of a PA's Qualities

Statement: A tutor that…	Min., Max.	Mean	Std. Dev.	Ranked: Most Imp.	Ranked: Least Imp.
…you would use again	4,9	7.91	1.463	40%	40%
…you would strongly recommend to others	4,9	7.71	1.447	29%	54%
…you would enjoy working with	6,9	8.31	0.963	34%	34%
…you feel motivated to work with	5,9	8.14	1.264	14%	51%
…helps you better understand the learning content	7,9	8.60	0.604	49%	26%
…lets you know how well you are doing	2,9	8.06	1.434	37%	20%
…keeps you updated on your progress	5,9	8.11	1.022	31%	23%
…understood how much you knew	6,9	8.43	0.815	31%	31%
…provided you helpful feedback	5,9	8.51	0.853	49%	17%
…Increases your interested in the learning content	5,9	7.91	1.380	31%	34%
…holds your interest	5,9	8.14	1.264	14%	34%
…you're satisfied with his performance	5,9	8.09	1.147	40%	34%

Participants were also asked to rate and rank the important attributes of their ideal learning environment. These items were used gauge the participants initial expectations of what is deemed necessary for computer-based instruction. Table 2 presents the results.

Table 2: Initial Expectations of a Learning Environment

Statement: A learning environment that…	Min., Max.	Mean	Std. Dev.	Ranked: Most Imp.	Ranked: Least Imp.
…is easy to use. (PEU)	5,9	8.14	1.115	57%	26%
…is controllable. (PEU)	2,9	7.43	1.668	17%	34%
…is enjoyable. (PEU)	4,9	7.54	1.482	17%	51%
…does not require a lot of mental effort. (PEU)	2,9	6.74	2.049	26%	51%
…is easy to learn how to use. (PEU)	6,9	8.14	1.089	40%	31%
…is ease to intuitively navigate through. (PEU.	4,9	8.17	1.224	26%	17%
…has good functionality (features). (PEU)	5,9	8.06	1.211	37%	20%
…is useful for learning content. (PU)	5,9	8.40	1.006	51%	17%
…is reusable for learning other content in the future. (UI)	4,9	7.43	1.720	29%	51%
Note: Perceived Ease of Use (PEU), Perceived Usefulness (PU), and Future Usage Intentions (UI)					

4.2. Post-Interaction Analysis:

Responses of post-interaction perceptions/acceptance are segmented by experimental condition and Sudoku experience. Both are compared against one another as well as analyzed as a whole. This section addresses the three hypotheses mentioned at the beginning of the methodology section. During the experiment, participants were subjected one of four versions of a PA: an Emotionally Supportive and Competent (ESC) agent (N = 9); an Emotionally Supportive Only (ESO) agent (N = 9); a Competent Only (CO) agent (N = 8); and a Neither Emotionally Supportive nor Competent (NESC) agent (N = 9). A between-condition analysis was conducted to assess the effect a PA's emotional support and knowledge level has on the learner's perception of the PA and learning environment.

The participants subjectively rated the PA's qualities/characteristics (same items assessed pre-interaction, see Table 1) and the learning environment based on their experience. Results convey the data had a significant positive relationship between acceptance of a PA and acceptance of the learning environment. Pearsons correlations showed the agent, regardless of condition, had a strong positive relationship to the learning environment's perceived ease of use ($r = .808$, $p<.001$), perceived usefulness ($r = .799$, $p<.001$), and future usage intentions ($r = .868$, $p<.001$). Although this relationship slightly varies across the independent experimental conditions, it still maintained its significance.

Next, we examined the influence prior experience has on perceived trust towards the PA. A univariate Analysis of Variance (ANOVA) was conducted and resulted in a main effect between an individual's Sudoku experience and their perceived trust of the tutor agent, $F(2,32) = 5.531$, $p<.05$. Upon further analysis, a multiple comparisons revealed a significant difference in overall perceived trust between participants with basic ($M = 7.672$, $SD = 1.401$) and advanced ($M = 5.123$, $SD = 2.069$), $p<.01$. We found that agent condition to not have a significant effect within this sample on a PA's characteristics used to evaluate trust.

Furthermore, we evaluated the impact that the PA condition had on the self-reported SAM mood dimensions. A univariate ANOVA showed the agent condition the learner interacted with had a main effect on reported levels of Arousal, $F(3,31) = 8.713$, $p<.01$. The other two dimensions (Pleasure, Dominance) were not significantly affected by the PA condition experienced across participants. A pairwise comparison identified learners in the Emotionally Supportive and Competent Tutor (ESC) environment reported significantly higher arousal levels ($M = 3.3$, $SD = .686$) over learners in the control condition with the neither emotionally supportive or competent tutor (NESC) ($M = 6.111$, $SD = .776$), $F(1, 33)$, $p = .012$. The pairwise comparison of the remaining agent conditions were found to be non-significant.

As well, a learner's perception of knowledge gained in the Sudoku domain following interaction was assessed. In the post-survey, participants were asked if they felt their knowledge/understanding of Sudoku increased following the experience. A univariate ANOVA shows the agent condition assigned to have a significant main effect on the reported knowledge increase item $F(3, 31) = 11.346$, $p<.001$. When comparing all conditions against one another in a pairwise comparison, results show significantly more learners reported increased Sudoku knowledge in the ESC ($M = 1.867$, $SD = .161$, $p<.05$) and ESO ($M = 1.917$, $SD = .153$, $p<.05$) tutor conditions when compared to the condition with a NESC tutor (*M* = 1.356, *SD* = .182, $p<.05$). The CO tutor (*M* = 1.889, *SD* = .185) is approaching significance, $p=.05$, with a larger sample believed to produce a reliable difference.

5. DISCUSSION

A result of this study is better understanding what attributes of a PA and learning environment is most important to a learner. According to the pre-interaction survey, participants most preferred a PA which helps them better understand content and provide helpful feedback. Particularly, participants with less Sudoku experience (e.g. none or beginner) thought these PA qualities to be of more importance than advanced Sudoku players. Additionally, there was an equal distribution in the reusability ranking of a computer-based tutor. A majority of participants with little or no Sudoku experience ranked the idea of using the tutor again as least important, while a majority of advanced Sudoku players rated this quality as most important. Furthermore, a PA that is enjoyable to work with varied between experience, with the advanced participants ranking it most important and the novice/beginners ranking it least important. Thus, a learner's prior knowledge of a domain might result in different desired expectations of a computer-based tutor. Interestingly, all participants reported having a tutor that they were motivated to work with or could strongly recommend to others as least important.

In regard to the learning environment, the sample most preferred an environment that is easy to use and useful for learning content, and reported the reusability of a learning environment as least important. Perhaps when learners enter training experiences, they focus on the current experience and do not consider how it can impact future learning interactions. Conversely, our sample may have developed a domain-specific (Sudoku) conceptual model of the learning environment at the time of this question and could not perceive it to be scalable to other domains. More research is needed to identify the appropriate conclusion. Sudoku experience did not have an impact on learners' preferences of the learning environment.

According to the post-interaction analysis, there is a significant relationship between acceptance of the PA and the learning environment they are interacting within. This supports the theoretical basis and importance of PA research. Thus, PAs have the ability to minimize trainee resistance to the utilization of intelligent tutoring systems.

Additionally, a learner's initial domain knowledge may influence their trust for using a PA. We found that beginner Sudoku players were significantly more trusting of the PA than advanced players, regardless of the agent condition given. This outcome could be based on the generalization that advanced participants heavily relied on their prior domain knowledge and used the PA only as a supplement to achieve the game objectives, while less experienced participants relied more on the PA to better understand the domain and complete the game. This notion also becomes more apparent as we observed the differences in initial expectations of each group.

In terms of affect, mood dimensions (e.g. Pleasure, Dominance, and Arousal) were found not to have an impact on learner's acceptance of a PA or the learning environment; however, learners interacting with the PA exhibiting emotional support and competency reported a higher increase in arousal when compared across conditions. Determining whether this type of arousal was based on engagement, excitement, fear, or anger is difficult to interpret from this data set. Correlations show pleasure to decrease among participants in this condition as arousal increases, but this does not hold true among the remaining groups.

Results also show characteristics of support and competency in a PA to impact a learners perceived knowledge gain. This suggests agents must provide relevant and supportive feedback to promote higher efficacy and commitment for using a PA. The results of the study support this assertion by showing all conditions exhibiting some level of competent or emotional support to report a higher perceived understanding of Sudoku than participants in the condition designed with neither.

6. CONCLUSIONS

This study supports the importance of understanding learners' initial expectations prior to system interaction. Future related research should consider expectations in addition to assessing the impact of different PA characteristics on learning outcomes. Our results demonstrate that learners' perceptions of a PA can have a direct impact on their acceptance of the learning environment the agent is embedded within. The study consisted of a small sample size. Several findings identified during data analysis were approaching significance and may be noteworthy with a larger sample population. In addition to increasing or sample for future studies, assessment questions to help explain mood outcomes will be incorporated.

REFERENCES

Adcock, A. and Van Eck, R., 2005. Reliability and Factor Structure of the Atttitude Toward Tutoring Agent Scale (ATTAS). *Journal of Interaction Learning Research*, 16, 195-212.

Baylor, A., 2000. Beyond Butlers: Intelligent Agents as Mentors. *Journal of Educational Computing Research*, 22 (4), 373-382.

Bloom, B, 1984. The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. *Educational Researcher*, 13 (6), 4-16.

Burton-Jones, A. and Hubona, G., 2005. Individual differences and usage behavior: Revisiting a Technology Acceptance Model Assumptions. *The DATABASE for Advances in Information Systems*, 36 (2), 58-77.

Core, M., Traum, D., Lane, H., Swartout, W., Gratch, J., van Lent, M., and Marsella, S., 2006. Teaching Negotiation Skills through Practice and Reflection with Virtual Humans. *SIMULATION*, 82(11), 685-701.

Cho, D., Kwon, H., Lee, H., 2007. Analysis of Trust in Internet and Mobile Commerce Adoption. *Proceedings of 40th Hawaii International Conference on System Sciences*, pp. 50-60. January 3-6, 2007, Waikoloa (Hawaii, USA).

Davis, F., 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13*(3), 319-339.

Fatahi, S. and Ghasem-Aghaee, N., 2010. Design and Implementation of an Intelligent Educational Model Based on Personality and Learner's Emotion. *International Journal of Computer Science and Information Society*, 7 (3), 423-434.

Foner, L., 1997. Entertaining Agents: A Social Case Study. *Proceedings of the First International Conference on Autonomous Agents*, pp. 122-129. February 5-8, 1997, Marina del Rey (California, USA).

Holden, H., and Rada, R., 2011. Understanding the Influence of Perceived Usability and Technology Self-Efficacy on Teachers' Technology Acceptance. *Journal of Research on Technology in Education*, 43(4), 343-369.

Hubona, G., and Kennick, E., 1996. The Impact of External Variables on Information Technology Usage Behavior. *Proceedings of Hawai'i International Conference on System Sciences*, pp.166-176. January 2-6, 1996, Maui (Hawaii USA).

Kim, Y. and Baylor, A., 2006. PAs as Learning Companions: The Role of Agent Competency and Type of Interaction. *Educational Technology Research and Development*, 54 (3), 223-243.

Kim, Y. and Baylor, A., 2006. A Social-Cognitiv Framework for PAs as Learning Companions. *Educational Technology Research and Development*, 54 (6), 569-596.

Jackson, T., Graesser, A., and McNamara, D., 2009. What Students Expect May Have More Impact Than What They Know or Feel. *Proceedings of the 2009 Conference on Artificial Intelligence in Education*, pp.73-80, July 6-9, 2009, Brighton (East Sussex, UK).

Jennings, M., 2000. Theory and Models for Creating Engaging and Immersive Ecommerce Websites. *Proceedings of the 2000 ACM SIGCPR Conference on Computer Personnel Research*, pp. 77-85, Chicago (Illinois, USA).

Johnson, W., Rickel, J., and Lester, J., 2000. Animated PAs: Face-to-Face Interaction in Interactive Learning Environments. *International Journal of Artificial Intelligence in Education*, 11, 47-78.

Junaidi, J., 2007. Assisting Learning Using Computers through the Application of Animated PAs (APA) or Intelligent Agents. *Proceeding of the 1st International Malaysian Educational Technology Convention*, pp. 30-34. November 2-5, 2007, Johor Bahru, Malaysia.

Lang, P.J., 1985. *The Cognitive Psychophysiology of Emotion: Anxiety and the Anxiety Disorders*. Lawrence Erlbaum Publishers, Hillsdale, NJ

Lee, H., Ahn, H., and Han, I., 2006. Analysis of Trust in the E-Commerce Adoption. *Proceedings of the 39th Hawaii International Conference on Systems Science*, pp.113-123. January 4-7, 2006, Kauai (Hawaii, USA).

Lindgaard, G., Fernandes, G., Dudek, C., and Brown, J., 2006. Attention Web Designers: You Have 50 Milliseconds to Make a Good First Impression. *Behaviour & Information Technology*, 25, 115-126.

Kim, Y., Wei, Q., Xu, B., Ko, Y., and Ilieva, V., 2007. Math Girls: Toward Developing Girls' Positive Attitude and Self-Efficacy through PAs. *Proceedings of the 2007 Artificial Intelligence In Education Conference*, pp. 119-126. July 9-13, 2007, Los Angeles (California, USA).

Norman, D., 1994. How Might People Interact With Agents. *Communications of the ACM*, 37, 68-71.

Rosenberg-Kima, R., Plant, E., Baylor, A., Doeer, C., 2007. Changing Attitudes and Performance with Computer-Generated Models. *Proceedings of 2007 Artificial Intelligence In Education Conference,* pp. 51-58. July 9-13, 2007, Los Angeles (California, USA).

AUTHORS BIOGRAPHY

Dr. Heather Holden is a researcher in the Learning in Intelligent Tutoring Environments Lab within the U.S. Army Research Laboratory – Human Research and Engineering Directorate Simulation Training and Technology Center. The focus of her research is in artificial intelligence and its application to education and training; technology acceptance and Human-Computer Interaction. Dr. Holden's doctoral research evaluated the relationship between teachers' technology acceptance and usage behaviors to better understand the perceived usability and utilization of job-related technologies. Her work has been published in the Journal of Research on Technology in Education, the International Journal of Mobile Learning and Organization, the Interactive Technology and Smart Education Journal, and several relevant conference proceedings. Her Ph.D. and M.S. were earned in Information Systems from the University of Maryland, Baltimore County. Dr. Holden also possesses a B.S. in Computer Science from the University of Maryland, Eastern Shore.

Mr. Benjamin Goldberg is a member of the Learning in Intelligent Tutoring Environments Lab at the U.S. Army Research Laboratory's (ARL) Simulation and Training Technology Center (STTC) in Orlando, FL. He has been conducting research in the Modeling and Simulation community for the past three years with a focus on adaptive learning and how to leverage Artificial Intelligence tools and methods for adaptive computer-based instruction. Mr. Goldberg is a Ph.D. student at the University of Central Florida and holds an M.S. in Modeling & Simulation. Prior to employment with ARL, he held a Graduate Research Assistant position for two years in the Applied Cognition and Training in Immersive Virtual Environments Lab at the Institute for Simulation and Training. Recent publications include a proceedings paper in the 2010 International Conference on Applied Human Factors and Ergonomics and a poster paper presented at the 2010 Spring Simulation Multiconference.

A LOW-COST SIMULATION ENVIRONMENT FOR PORT SECURITY EXPERIMENTATION AND TRAINING

Tania Randall[a] and Allan Gillis[b]

Defence Research & Development Canada – Atlantic
PO Box 1012, Dartmouth, NS B2Y 3Z7

[a]tania.randall@drdc-rddc.gc.ca , [b]allan.gillis@drdc-rddc.gc.ca

ABSTRACT
An immersive simulation environment has been developed to support the exploration of new technologies and concepts in a port security operations center, where operations focus on the monitoring and protection of people and property in a harbour (e.g., investigating boats or passengers portraying unusual behaviour). The simulation environment includes four main components: an operations center, response boats, a naval vessel, and potential waterborne threats, each requiring input and control from two or more human operators. Each component was constructed as simply and inexpensively as possible, while still providing operators with information, communication, and control capabilities consistent with their expectations. Virtual BattleSpace2 (a commercial-off-the-shelf video game), High Level Architecture federates, and real-world tracking software have been integrated and paired with consumer-level electronics to produce an inexpensive experimentation environment. Components of this flexible and adaptable simulation are also being exploited as a tactical training capability for port security boat operators.

Keywords: port security, game-based simulation, human-in-the-loop simulation, simulation-based experimentation

1. INTRODUCTION

Port security operations centers revolve around incoming and outgoing information. Information comes in through various mechanisms: cameras, radar, radio, phone, online communication tools, etc. If issues of concern are identified, they are communicated up the chain of command and decisions are relayed back down to the relevant stakeholders. As years pass, more and more opportunities to collect and disseminate data become available (e.g., consumer-level availability of GPS systems, cell phones and smartphones), and the management of the breadth of information becomes more complicated. It is not immediately clear that the addition of a new data source, a new display for multiple data sources, or a new communication pathway will improve the performance of operations overall.

The motivation for developing the simulation environment discussed in this paper came following the observation of port security exercises, and identifying the potential for the use of incident management system (IMS) software to manage and track the details of incidents and resources. Since IMS systems typically run independently of other software and data sources found within an operations center, they can be easily added to a baseline simulation without integration concerns. By running through a selection of scenarios with the baseline capability alone (i.e., no software added) and alternatively with the addition of the proposed software solution, experimenters can capture data on the performance of the system as a whole as a result of the augmented setup. While the exploration of IMS systems was the impetus for developing the baseline simulation, the setup is easily adapted to explore other capabilities or ideas that could likewise improve operations.

This simulation environment discussed in this paper is referred to as Virtual Harbour, or simply vHarbour. vHarbour is comprised of a combination of commercial gaming software and input devices, off-the-shelf computers and displays, custom-built software, simulation-based communications, and real military kit. Basing the environment on extendable gaming software and consumer electronics limits the overall cost of the setup.

2. HARBOUR PROTECTION

The operations center focussed on in this paper is a military operations center, responsible for the protection of people and property in a harbour. The team in the operations center is supported by multiple small boat teams on small, fast, rigid-hulled inflated boats (RHIBs). The RHIBs work in teams, monitoring designated portions of the harbour and relaying issues back to the operations center. RHIBs may be required to investigate suspicious boat behaviour, protect vessels entering the harbour, monitor controlled access zones, etc. In addition to receiving reports from the RHIBs, the operations center team monitors cameras and other sensors.

In order to guide the focus of simulation development, it was necessary to identify some scenarios early on that the simulation would be able to handle. All scenarios are presumed to take place in Halifax Harbour (Nova Scotia, Canada) on a sunny, summer day when active pleasure-based traffic in the harbour is at its peak. Three scenarios are fictitiously built upon prior conditions that have caused the Canadian Forces Base (CFB) Halifax to adopt a Force Protection (FP) state of yellow (i.e., a medium threat level). Each scenario involves five FP RHIBs, where two of the five RHIBs are escorting a foreign vessel into the harbour for a port visit. There are three Canadian Navy ships already in the harbour; which ship is an active part of the scenario will depend on the location of the potential incident described in the scenario.

The first scenario involves two small high speed boats approaching the escorted foreign vessel on a direct path. The second involves a fishing boat that stops briefly alongside a barge and leaves again with two people on its deck, getting into diving gear. When the boat nears one of the naval ships, the two divers jump in the water and head for the ship. The third scenario involves a small powerboat with two men on board that continually shadows the escorted warship; as a force protection boom is opened to allow the warship to be escorted inside, the small boat speeds up and attempts to enter the boomed area.

Figure 1 provides a visual of the entities involved in each scenario and their anticipated interactions (data sharing and/or communication).

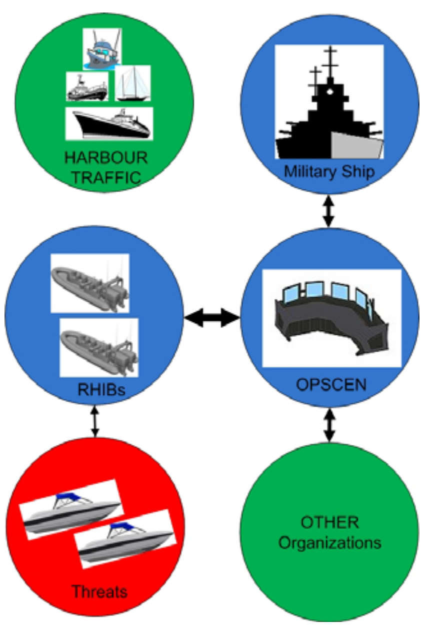


Figure 1: Scenario Entities

The heavier lines indicate that more interaction is anticipated between these elements. The absence of connecting lines does not indicate that entities could not conceivably communicate or share data with each other. Rather, it means that such interactions are not expected to directly influence the outcome of experiments focussed on the operations center.

The simulation was developed to handle (at a minimum) the three scenarios that have been described. Some scenario-specific elements influenced the development efforts (e.g., the third scenario requires that a force protection boom be in place and that there be an opening for the warship to enter through), but the simulation is flexible and can be adapted to handle alternative scenarios as well.

3. vHarbour SIMULATION COMPONENTS

The simulation environment focuses on the operations center since this is where new technologies or concepts will be trialed. All other simulation components are included simply to enhance the realism of the experience of those in the operations center. The other human-controlled components include two RHIB response boats, a naval vessel with a sentry and force protection officer, and up to two human-controlled (potential) waterborne threat boats. A naval vessel is included in each scenario primarily to stimulate the need to share and track information between multiple locations, creating a better testing environment for new software.

This section briefly describes the requirements for the core components of the simulation environment (i.e., the ‘what’) as determined by real-world observation of port security exercises; as much as possible the simulation environment mimics what was observed. The following section (section 4) will describe the software and electronics used to create the environment to handle these requirements.

3.1. Operations Center

The operations center includes all of the information feeds available in an actual operations center: video feeds of numerous cameras on the harbour, each adjustable in direction and zoom, radar and Automatic Identification System (AIS) tracks of ships/boats in the harbour (when available), and a map showing response boat positioning and communication options. This ‘map’ is an Electronic Chart Precise Integrated Navigation System (ECPINS) chart with an Asset Control and Tracking (ACT) add-on for RHIB position-tracking and communication between the RHIBs and the operations center.

Other tools are also available within the center, such as a whiteboard, phone lists, and Microsoft Office software. Communication occurs via radio, chat, e-mail and phone to military ships in the harbour as well as any other ships/boats and organizations that would typically be engaged with during harbour monitoring and protection.

3.2. Response Boats

RHIB (response) boats need to communicate their position to the operations center. To some extent, their position may be observed through cameras (or eyes) on the harbour, however, verbal reports are also called in over radio at fixed intervals. In addition, some of the boats on duty are fitted with a commercial ACT RHIB-based unit which provides automated position reports to the operations center's ECPINS display at regular intervals using a secure AIS channel. RHIBs with ACT have a small monitor mounted near the boat coxswain (driver) that shows the location of other ACT-fitted RHIBs as well as AIS and radar tracks on a harbour chart. ACT can also receive textual messages as well as map overlays from the operations center (e.g., exclusion areas, or points of interest) and can be used to return primarily preset messages. In addition to ACT, RHIB boat operators can (obviously) use their vision to see what is happening on the harbour, and this is a key source of day-time data collection which must be replicated in the simulation environment.

3.3. Red Team

While a variety of threats can exist in a harbour environment, the scenarios designed for this simulation focus on small boat threats. The boats are assumed to have no sensor feeds beyond operator vision. They have access to radios, but would be unlikely to use the radio to communicate ill-intended plans; they may use cell phones to speak with each other or communicate simply through verbal cues.

3.4. Ship

All military vessels in the harbour that are of relevance to any of the prescribed scenarios are stationary. With a force protection state of yellow, a Force Protection officer (FPO) and upper deck sentries can be expected on each ship. FPOs have chat, phone and/or radio communication pathways to the operations center, other organizations and to boats on the harbour. The sentries would have visuals on the harbour and radio communication with the FPO.

3.5. Other Ships and Agencies

The operations center staff may need to communicate with other ships/boats in the harbour (e.g., a small boat that is too close to a commercial shipping lane) or other military or non-military agencies via phone, e-mail or radio. It is difficult to predict which other agencies might be engaged; it depends on the scenario and the decision maker's interpretation of the events. However, likely candidates are the military and civilian police, local fire and emergency services, Halifax Port Authority (HPA), etc.

4. SIMULATION COMPONENTS

At this point the simulation requirements have been identified. This section focuses on the software and hardware components used or developed in order to meet those requirements, providing simulation fidelity sufficient for experimentation with new technologies or concepts.

4.1. Virtual BattleSpace 2 (VBS2)

VBS2, developed by Bohemia Interactive (www.bisimulations.com), is an immersive 1st person virtual environment (which is also referred to as a 'serious game') and is at the heart of vHarbour. It is used for scenario generation and editing, exercise control, and after-action review of operator decisions. All operator-controlled entities (RHIBs, threat boats, the sentry and also the cameras in the operations center) in the simulation are created through VBS2 at scenario design time and controlled through VBS2 instances at run time. One instance of VBS2 is used to simulate each of the cameras in the operations center; four are run on each of two computers and the windows are tiled 2x2 onto two 52" TV monitors. Each camera provides a different view of the harbour, and allows the operator to do basic panning and zooming as required. VBS2 also provides the visual environment (i.e., what the operator sees in the harbour) for each of the operator-controlled RHIBs and threat boats, as well as a visual display for the sentry onboard the ship. VBS2 terrain for Halifax harbour with geo-specific (e.g., key buildings, the dockyard, and bridges) and geo-typical elements was built for vHarbour. As well, it was necessary to build additional VBS2 models for a variety of vessel types including jet skis, fishing boats and other small water craft.

4.2. High Level Architecture (HLA) federates

Defence Research & Development Canada – Atlantic (DRDC Atlantic) has been developing HLA-based federations for over eight years, with most work based on the Virtual Maritime Systems Architecture (VMSA) (Canney 2002). Thus, core federates related to simulation control already existed. Other federates needed to be developed, including the following:

- A Scenario Editor and Visualizer (ScEr-V) for generating and directing harbour background traffic (Gaudet 2009),
- A radar federate that produces VMSA radar tracks for all entities in the simulation (generated by ScEr-V and/or VBS2) within size and range limitations and not blocked by obstacles or masked by shadow zones (Hackett 2009),
- An Automatic Identification System (AIS) federate that simulates the behaviour of AIS transponders, producing VMSA AIS tracks (and necessitating a custom addition to the VMSA Federation Object Model (FOM)) (Gaudet 2009),
- A National Marine Electronics Association (NMEA) federate that receives radar and AIS data from the simulation, formats it according NMEA standards and passes it over a serial connection to NMEA-compliant listening devices (thus bridging the simulation to real-world equipment) (Campaigne 2009), and

- A software bridge for exchanging data between simulation components using the High Level Architecture (HLA) and others using the Distributed Interactive Simulation (DIS) protocol; this DIS-HLA bridge (named 'Charon') supports the passage of simulation data between VBS2 (which supports DIS) and VMSA (which is based on HLA).

ScEr-V was used to generate the background traffic in Halifax Harbour for each scenario, including commercial shipping vessels, fishing boats, pleasure craft, and the Halifax-Dartmouth ferry (which follows the real-world schedule). It was also used to create three simulation-controlled RHIBs for each scenario. The radar and AIS federates produced tracks for both ScEr-V-generated and VBS2-generated entities, as appropriate. These tracks were picked up by the NMEA federate and passed to chart displays on the RHIB and in the operations center. Charon allowed VBS2 to 'see' the VMSA entities and vice versa.

4.2.1. Scenario Editor and Visualizer (ScEr-V)

ScEr-V provides a graphical interface (Figure 2) for building scenarios by dragging and dropping entities onto a map image, adding waypoints, paths, areas of operation (e.g., fishing grounds) or exclusion (e.g., controlled access zones), and specifying the types of behaviour each entity should perform. Behaviour options include: moving to an absolute or relative point, operating within an area (fishing, sailing, etc.), following a path, and drifting. It allows for specification of entity speed and duration of each behaviour, and supports switching from one behaviour to another after a pre-set amount of time or the completion of a task (e.g., after fishing an entire area, return to port).

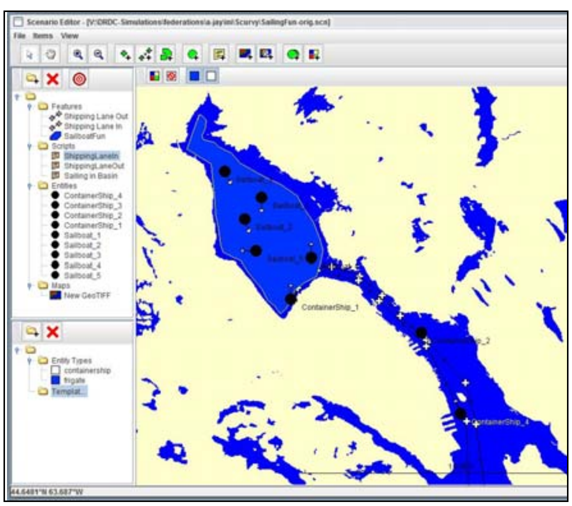


Figure 2: ScEr-V Graphical Interface

The ScEr-V scenarios are recorded as script files that are exported to an agent-based navigation federate that ultimately controls entity movement at run-time. The navigation federate also handles the priority of behaviours for each entity. For example, a fishing behaviour will be suspended if a munitions detonation occurs near that entity since escaping from a threat is more important than fishing. A ScEr-V viewer is also available at run-time to monitor entities and take control of them as needed. For vHarbour, an additional 'white cell' player was assigned to monitor the radio for requests issued to neutral traffic and to modify behaviours as necessary. In short, ScEr-V creates computer generated forces (CGFs) capable of controlling their own behaviours. In contrast, VBS-2 is used in vHarbour to create entities requiring/supporting human input throughout scenario execution.

4.2.2. DIS-HLA Bridge (Charon)

VBS2 and VMSA need to be able to exchange entity-related information with each other (type of boat, location, speed, etc.) so that VBS2 can properly display entities created and controlled by ScEr-V and the VMSA sensor federates can produce tracks for VBS2-based RHIBs. The VBS2 licence purchased by DRDC included LVC-Game by Calytrix Technologies (www.calytrix.com) that allows VBS2 to participate in DIS exercises. VMSA is based on HLA, so a DIS-HLA software bridge is required to exchange data between VBS2 and VMSA.

While it would have been possible to purchase a commercial product to perform this translation between DIS and HLA, the choice was made to develop this bridge in-house. A commercial solution would have required considerable configuration, been less flexible, and have taken additional time and money to procure. The bridge built in-house, Charon, was programmed using the Open Source DIS project as well as a DRDC-written HLA toolkit known as "Polka" and took approximately four weeks to build. Charon's core responsibility is to translate identification and spatial information for all VMSA entities into information that can be understood by VBS2 and vice versa.

4.3. Real Military Kit

ECPINS software was purchased for the operations room and ACT software was likewise purchased for one simulated RHIB station, both from OSI Geospatial Inc. (www.osigeospatial.com). This is the actual software used in the operations center and onboard the RHIBs and can run on any modern Windows-based computer. It provides blue force tracking of all ACT-fitted RHIBs (i.e., knowledge of where they are) to the operations center and the RHIBs themselves. It also allows the operations room staff to send instructions to the ACT-fitted boat, such as new exclusion zones drawn on a map, or textual instructions and warnings. An operator on the RHIB can return similar messages, typically by selecting phrases from a preset list rather than typing from scratch.

In the real world, positions and messages are passed via an encrypted AIS channel. In the simulated world, there are two challenges: passing data from the simulation to the ECPINS and ACT software and exchanging information between ACT (on the RHIB) and ECPINS (in the operations room). The latter was straightforward as an AIS Network Simulation came with the ECPINS and ACT software for exactly this purpose and it was simply a matter of configuring it according to the documentation. The former requires the passing of AIS and radar tracks from the VMSA simulation to the ECPINS and ACT software, as well as truth-based GPS data to both ECPINS and ACT so that they would know where their host (i.e., the operations center and the RHIB, respectively) is located.

One instance of the NMEA federate and two computers are required for each NMEA-compliant listener since data are passed via a serial connection. Thus, pushing VMSA tracks to both the ECPINS and ACT software requires two instances of the NMEA federate, each running on different computers, and two additional computers to run the ECPINS and ACT software. In the future, the design of this federate could be modified to work via TCP/IP and remove the device dependency.

Any VBS2 data required by ECPINS and ACT is routed through the bridge and the VMSA simulation, so there is no direct connection between VBS2 and ECPINS and ACT.

Given the use of the NMEA standard to pass data to a NMEA-compliant listener (such as ACT), it is now possible to share VMSA simulation data with *any* real-world device that is NMEA-compliant (e.g., a GPS system).

4.4. Radio Simulator

The idea of using real radios to support the simulation was explored, but difficulties in obtaining an adequate number, their inflexibility, and limitations on where they could be used led to the selection of a simulated radio network setup. SimSpeak, developed by the Canadian Forces (CF) was used for this purpose, and is similar to other Voice Over Internet Protocol (VOIP) systems used by gamers. Microsoft Life-Chat gaming headsets were used for monitoring and broadcasting on a single radio channel via SimSpeak.

4.5. Telephone Simulator

A commercially available telephone line simulator (Teletone TLS-5 (www.teltone.com)) was used to program the phone lines for the operations center, the ship, and 'all other organizations'. At simulation execution time, a single human actor must take responsibility for acting as any organization/agency that the operations center staff or military ship decide to contact (via phone or radio). A single phone line is assigned to represent all such organizations for a couple of reasons. First, the telephone simulator can only handle four simulated lines, and there are more than two potential agencies that could be called. Secondly, even if adding more lines was possible, the actor would need to answer all of those phone lines. Instead, there is a single line 'to all', and callers are instructed in their pre-experiment briefing to always identify themselves and who they are calling so that the actor knows what organization he/she is representing at any given time.

4.6. E-mail/Chat Service

A Linux-based virtual appliance was freely downloaded from VMWare and used for the e-mail server. E-mail addresses could then be provided to the operations center staff, the FPO on the ship, and the multiple organizations handled by the actor. A similar process is expected to be used to provide a chat service, although this has not been thoroughly explored to date.

5. RHIB Station

From an experimentation point of view, the operations center is the focus of vHarbour. However, the heart of vHarbour from a simulation perspective is the small boat station set-up.

There are four of these stations in the vHarbour set-up. Two stations are used for RHIBs that work together under the instruction of the operations center; one of these is fitted with ACT. The other two stations are used for the red team. They are identically configured (with no ACT) in terms of their hardware and software; however, settings in VBS2 are modified according to the type of threat boat(s) in the scenario.

The technical integration for the RHIB/small boat set-up has already been discussed – VBS2 supplies the visuals and control of the RHIBs and a headset is used with SimSpeak to supply radio communications.

Figure 3 shows a RHIB station that is configured with ACT.

Figure 3: RHIB station with ACT

5.1. Hardware and Structural Components

A small Anthrocart desk was configured to make the driving podium. A 52" TV was used to display the operator's view of the harbour via VBS2 and was mounted such that the average operator's sight would center on it while in a standing position. Its large size means that it is capable of filling most of the operator's field of view and gives a sense of immersion in the environment.

A Track-IR (www.naturalpoint.com/trackir) gaming peripheral was used to control the operator's point-of-view in VBS2. It achieves this by amplifying head movements such that an operator can look all the way over one shoulder by turning the head slightly away from centre. It uses an Infrared camera on top of the TV and a reflective target attached to the operator's hat to detect head position and orientation. The setup also uses a Saitek throttle quadrant and a Logitech force-feedback steering wheel (both found commercially in the gaming sector), so that the control of the boat is achieved in a manner similar to the real world. The ACT software is displayed on a small monitor and controlled by a joystick (both identical to those used on actual RHIBs) which are mounted next to the driver providing blue force tracking and communication with the operations center.

5.2. Motivation for Human-Controlled RHIBs

The choice to develop immersive RHIB stations may seem extreme, given that the focus of experimentation was meant to be in the operations room. However, the information detected visually by the RHIB operators and communicated to the operations center is critical to operator decision making. Thus, it is important to make these inputs as realistic as possible.

In past simulation-based experiments, DRDC scientists have attempted to give white cell players a god's-eye-view of the simulation (showing the position and identity of all entities) and lists of rules about what information they could release to the rest of the team (in this case, the operations center) based on certain conditions. For example, as they got closer to an entity, they could reveal more and more details about what they saw (e.g., a small boat, or, a small fishing boat with two people onboard). Obviously this is simple to implement in terms of simulation requirements; in this case, the hardest part is creating reasonable rules and scripts for the white cell players to use. Yet, even with well developed instructions, the white cell players had difficulty not releasing too much information too soon, and in some cases missed information that they could have provided. It was clear that for future studies it would be better to eliminate this uncertainty by providing a first-person viewpoint through a visual display that would only allow the individual to learn the appropriate level of detail at the appropriate time. This is exactly what VBS2 does for the RHIB drivers.

It is also possible to have scaled back the hardware set-up, but given the use of readily available consumer-level components, the additional expense is easily justified by the increase in realism for the RHIB and small boat drivers. Further, it's a logical assumption that a better, more realistic experience for the boat operators will translate to increased realism for the simulation overall.

Figure 4 provides a high level conceptual overview of the vHarbour simulation components and the links between them.

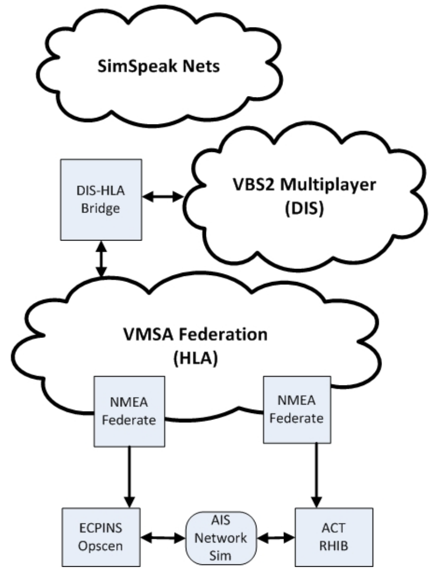


Figure 4: The vHarbour Simulation

6. vHarbour-ASSOCIATED COSTS

In 2010, the Canadian Army bought an Enterprise license of VBS2 so there was little cost associated with DRDC's use of VBS2 licenses. Contract support was used to assist with VBS2 model and terrain development as well as the development of the ScEr-V, radar, AIS and NMEA VMSA federates. These are now Crown IP and are reusable and applicable to future work efforts. ECPINS and ACT are COTS products that were purchased at a reduced price for simulation use only. SimSpeak is 100% Canadian Crown IP and available to DND users at no cost. The telephone line simulation is an inexpensive, commercially available product. The e-mail server was downloaded freely from VMWare. Headsets and all RHIB station components are commercially available and aimed at the entertainment industry thus making them affordable to the average person. Not including the initial costs of software development and integration, it is estimated that all of the components (including three computers) of a RHIB station without ACT can now be purchased for approximately $3500, including the VBS2 licences for those without access to an enterprise agreement. The remainder of the vHarbour setup was run on basic modern computers with desktop displays and keyboards.

7. INCIDENT MANAGEMENT SYSTEM (IMS) EXPERIMENTATION

vHarbour was developed over approximately a one-year period with three computer scientists working on it part-time in addition to the aforementioned contractor support. Given the lead time needed to prepare the simulation environment, development began based on the early premise that the port security operations center could benefit from having an IMS system. After looking at a number of commercial systems and further considering their applicability to this environment in particular, it was felt that these systems offered capabilities exceeding what was really needed. Also, many IMS systems require considerable training and frequent practice to maintain proficiency. Evaluating system usability as part of the software selection process can minimize this problem (Randall 2011). However, to date, the authors have not identified a system that they believe is appropriate enough for this operational setting to take on the effort of organizing participants and engaging in the experimental process. While the simulation environment is near completion, plans to trial IMS systems have been reconsidered.

8. vRHIB

In July 2010, Halifax hosted an International Fleet Review (IFR) in celebration of the Canadian Navy's 100th birthday. In addition to many visiting foreign naval vessels, Her Majesty Queen Elizabeth II sailed in the harbour during the review. In preparation for this visit, a major force protection exercise was conducted in early June. Naval Reserve (NAVRES) port security unit (PSU) and CF boat operators would be patrolling the harbour on RHIB boats during the IFR and lead-up exercises. Since the ACT systems were a relatively new purchase for PSUs and indeed would not be shipped to Halifax until late in the exercise period, DRDC offered to assist the operators with ACT familiarization through the provision of a reduced version of vHarbour, focusing on the RHIBs themselves. Since the two threat vessel stations were identical to non-ACT RHIB station set-ups, four RHIB stations could be offered, one with ACT. The offer was accepted and with tight deadlines, vHarbour was stripped of the then unnecessary phone simulator and ship, and the operations center was reduced to a single station running ECPINS and a radio. All of the simulation control was reduced to run on two computers: one for the VMSA federation and another to run VBS2 in administrative mode. All of the scenario generation was moved from VMSA to VBS2 alone and the four RHIB stations were configured to be blue forces (rather than two blue and two red). This entire set up was then moved to HMCS SCOTIAN, the NAVRES unit where the pre-IFR exercise was being conducted, to make the simulation more accessible to the boat operators. It was set up in two rooms, one for the operations center (i.e., ECPINS) and one for the four RHIB stations. The originally envisioned plan of training boat operators within the full simulation environment became unnecessary when the actual ACT units arrived early and became available for real-world training. However, the trainers did end up making use of the simulation in an entirely different manner. They turned one of the RHIB stations around so that operator could not see the other operators and designated that RHIB as red force. The three remaining RHIB operators practiced working together as a team to deal with the threat vessel. New scenarios were requested on-the-fly and were implemented quickly within the VBS2 environment. Thus, the simulation was used by NAVRES for training in their own way, ultimately demonstrating the flexibility of this environment now that the core pieces are in place.

As a result of the initial success with using vRHIB, a contract has been let to refine the vRHIB environment such that it can be easily maintained and used for training within NAVRES units across Canada. This contract effort will also address issues related to integrating a new training system into existing training plans.

9. OFF-THE-SHELF SOFTWARE FOR SIMULATION ENVIRONMENTS

Low cost COTS simulations, games, and peripherals continue to increase in capability while decreasing in cost. DRDC Atlantic is currently using another COTS product, Dangerous Waters, as the core simulation engine for a human factors experimentation environment for the Victoria Class submarine's operations center. Many other COTS products are available and cover numerous simulation domains. Often these products can be leveraged for experimentation or training simulations aimed at concept or tactical development, where a 'close enough' look and feel is sufficient to meet the end goals of the simulation. Investigating the low-cost and serious games marketplace before engaging in the development of a completely custom simulation environment may indeed be time well-invested.

10. CONCLUDING REMARKS

With only a year's effort from a small part-time team, a simulation environment has been created to fully simulate all relevant inputs to a port security operations center. It was created inexpensively, using VBS2 as its backbone, supported with consumer-level electronics and devices, existing software components, and a modest amount of new software development. This environment can now be used for investigating the use of new software or procedures and plans in the operations center. A simplified version of this flexible environment focusing on a team of RHIB drivers was able to support real-world tactical team training efforts. In the future, a refined version of the RHIB team simulation will be made available for training use by all NAVRES port security units in Canada.

11. REFERENCES

Canney, S.A., 2002. *Virtual Maritime Systems Architecture Description Document Version 2.00*, Defence Science and Technology Organisation, Document Number 00034, Edinburgh, Australia.

Gaudet, B., 2009. *Scenario Editor and Visualizer (ScEr-V): User Guide and Technical Description*, DRDC Atlantic Contract Report prepared by Greenlight Software for MacDonald Dettwiler & Associates DRDC Atlantic CR 2009-005, Dartmouth, Nova Scotia, Canada.

Hackett, D., 2009. *RADAR Federate: User Guide and Technical Description*, DRDC Atlantic Contract Report prepared by Greenlight Software for MacDonald Dettwiler & Associates, DRDC Atlantic CR 2009-044, Dartmouth, Nova Scotia, Canada.

.

Gaudet, B., 2009. *AIS Federate: User Guide and Technical Description*, DRDC Atlantic Contract Report prepared by Greenlight Software for MacDonald Dettwiler & Associates, DRDC Atlantic CR 2009-006, Dartmouth, Nova Scotia, Canada. .

Campaigne, F., 2009. *NMEA Federate: User Guide and Technical Description*, DRDC Atlantic Contract Report prepared by Greenlight Software for MacDonald Dettwiler & Associates, DRDC Atlantic CR 2009-054, Dartmouth, Nova Scotia, Canada.

Randall, T., 2011. Incident Management Systems Evaluation and Usability Assessment, *Proceedings of the 16th ICCRTs*, June 2011, Quebec City, PQ, Canada, DRDC Atlantic SL 2011-026, Dartmouth, Nova Scotia, Canada.

VISUAL ANALYTICS APPLICATIONS FOR CYBER SECURITY AND INTELLIGENCE

Valérie Lavigne[(a)], Denis Gouin[(b)]

[(a) (b)]Defence R&D Canada – Valcartier
2459 Pie-XI Blvd. North
Quebec (Quebec), Canada, G3J 1X5

[(a)]valerie.lavigne@drdc-rddc.gc.ca [(b)]denis.gouin@drdc-rddc.gc.ca

ABSTRACT

In the context of modern defence and security operations, analysts are faced with a continuously growing set of information of different nature which causes significant information overload problems and prevent developing good situation awareness. Fortunately, Visual Analytics (VA) has emerged as an efficient way of handling and making sense of massive data sets by exploiting interactive visualization technologies and human cognitive abilities. Defence R&D Canada has conducted a review of the applicability of VA to support military and security operations. This paper is meant to provide someone new to this area with a quick overview of the current state of the art in visual analytics. First, we introduce the important scientific visualization, interaction and reasoning concepts supporting VA. Then, we present some visual analytics advanced techniques. The VA requirements are described for the cyber security and intelligence for counterterrorism application domains, along with promising research projects and commercial software. Finally, we identify relevant VA resources.

Keywords: visual analytics, cyber security, intelligence for counterterrorism, interactive information visualization.

1. INTRODUCTION

In the era of the information age, decision makers and first responders in Defence and Security are faced with increasing amounts of dynamic information originating from a wide variety of sources and in a wide variety of formats, which they need to analyse in order to understand a situation and react promptly. This is the case for example, in situation management following a natural or man-made disaster, a terrorist attack, a military conflict, a pandemic flu or a criminal series of activities. In developing Situation Awareness (SA), analysts must understand how a situation has developed and how it may develop. They need to identify trends and patterns. They need to work in collaboration with representatives from different organizations.

Although information fusion and rule-based systems have shown their value in helping making sense of information and providing situation awareness, during the last decade, a new science and technology called Visual Analytics (VA) has rapidly emerged in helping users understand a situation by cleverly representing the information and providing mechanisms to interact with the information.

This paper is meant to provide someone new to this area with a quick overview of the current state of the art in visual analytics. We begin with an introduction of the important scientific visualization, interaction and reasoning concepts supporting VA and we present some visual analytics advanced techniques. Then the VA requirements are described for the cyber security and intelligence for counterterrorism application domains, along with promising research projects and commercial software. Finally, we identify relevant VA resources.

2. VISUAL ANALYTICS

"Visual analytics is the science of analytical reasoning facilitated by interactive visual interfaces." This is the most widespread definition of VA and it comes from the US research agenda that launched the field: Illuminating the Path (Thomas and Cook 2005).

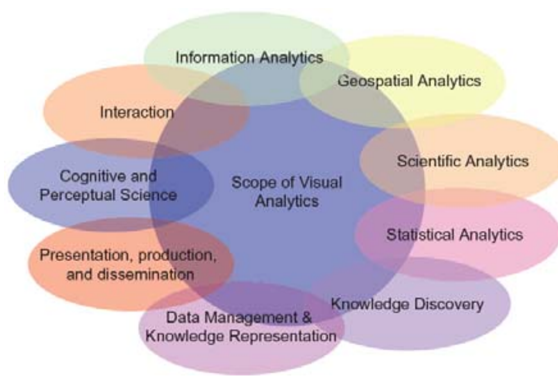


Figure 1: Visual analytics as a highly interdisciplinary field of research (Thomas 2009).

VA is a multidisciplinary field that combines various related research areas where much valuable prior work has been done. Figure 1 shows a non-exhaustive list of scientific disciplines that are related to VA (Thomas 2009). The challenge of making visual analysis effective calls for advancement in a variety of scientific fields. Visualization, interaction science and analytical reasoning are three research domains that bring highly important scientific concepts to VA and these are described in the following sections.

2.1. Visualization

Humans discovered a long time ago that they could enhance their cognitive abilities by using external representation aids (Card et al. 1999). The use of visualization to present information is not a new phenomenon. It has been used in maps, scientific drawings and data plots for over a thousand years.

The intent of visualization is not merely to display information using pictures. The visual representation should be designed in a meaningful way in order to provide insight to the user. The optimal choice is highly dependent on the data involved and the task to be performed.

Information can be presented visually using points, lines, shapes, colors, intensity, textures, motion, etc. To select effective visual cues for data representation, we can use results obtained from the study of human perception. Preattentive features form a set of visual properties that are detected very rapidly and accurately by the low-level visual system. These properties were initially called preattentive since their detection seems to precede focused attention. This process is effortless, meaning that it does not demand attentional resources for a human. In each case presented in Figure 2, a unique visual property in the target allows it to "pop out" of the display. Examples of tasks that can be performed using preattentive features are: target detection, boundary detection, region tacking; and counting and estimation (Healey 2009).

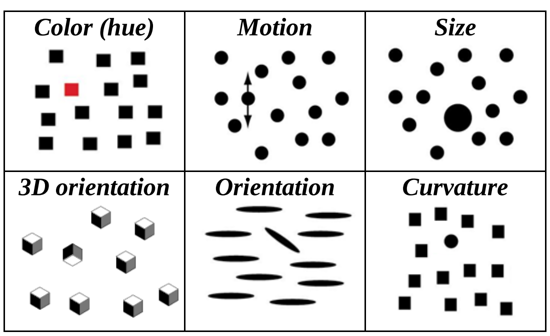


Figure 2: Examples of preattentive visual features (adapted from Healey 2009).

In Figure 3, the use of semantic depth of field makes some chess pieces more salient to guide user's attention (Kosara et al. 2001).

Color in information presentation is mostly used to distinguish one element from another (Stone 2006). Contrasting colors are different and draws attention, while analogous colors are similar and groups elements (see Figure 4). As Tufte (1990) puts it: "avoiding catastrophe becomes the first principle in bringing color to information: Above all, do no harm." Chosen poorly, colors can obscure the meaning of information.


Figure 3: Semantic depth of field relies on preattentive features to offer a focus + context view. Focusing effects can highlight information (Kosara et al. 2001).

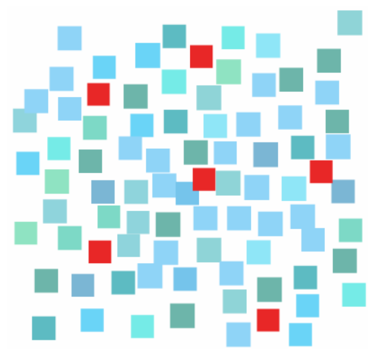


Figure 4: Contrast and analogy (Stone 2006).

The Gestalt laws of organization describe how people perceive visual components as organized patterns or wholes, instead of many different parts (Koffka 1935; Wertheimer 1938). According to this theory, there are six main factors that determine how we group things according to visual perception: closure, similarity, proximity, symmetry, continuity and common fate.

Inattentional blindness, also known as perceptual blindness, refers to the inability to perceive features in a visual scene when the observer is not expecting them. Salient features within the visual field will not be observed if not processed by attention because the amount of information processed at any particular time is limited. In an experiment, 50% of subjects that were asked to watch a short video in which two groups of people pass a basket ball around and asked to count the number of passes failed to notice that a woman wearing a gorilla suit had walked through the scene (Simons and Chabris 1999).

2.2. Interaction

Interaction enables the user to explore the data, try out hypotheses, drill into data, gain insight and collect knowledge. Human computer interaction has been an active research field for many years and is the study of interaction between people and computers. Visualization is considered interactive if control of some aspect of the information presented is available through a human input and the changes made by the user are incorporated in a timely manner.

Three categories of responsiveness (0.1s, 1s, and 10s) have been suggested to give an order of magnitude of the required response time for interactivity (Miller 1968, Card et al. 1991). 0.1s is the upper limit for the system response to feel instantaneous. After more than 1s, the user's flow of thought is interrupted and the user loses the feeling of operating directly on the data. For delays longer than 10s, users will want to perform other tasks while waiting for the computer calculation to complete.

The famous visual information seeking mantra for designing advanced graphical user interfaces "overview first, zoom/filter, details on demand" comes from the interaction taxonomy from Shneiderman (1996). More recently, Soo Yi et al. (2007) proposed seven general categories of interaction techniques in information visualization:

- Select: mark something as interesting;
- Explore: show me something else;
- Reconfigure: show me a different arrangement;
- Encode: show me a different representation;
- Abstract/Elaborate: show me more or less detail;
- Filter: show me something conditionally;
- Connect: show me related items.

These categories are organized around the user's intent while interacting with the system rather than the low-level interaction techniques provided by the system. Soo Yi et al. (2007) also point out that "for different representation techniques, different interaction techniques are used to perform a similar task or achieve a similar goal".

2.3. Analytical Reasoning

Visual analytics is intended to be an active, engaging exploratory process of discovery. This human-information discourse is between the analyst and his data. It supports three goals: assessment (understand current situation and explain past events), forecasting (estimate future capabilities, threats, vulnerabilities and opportunities) and planning (develop options, create possible scenarios, prepare reactions to potential events). Analysts apply reasoning techniques in order to achieve these goals. Visual analytics is meant to facilitate high quality analysis with a limited quantity of user's time. Six basic ways were identified in how information visualization can expand human cognition (Card et al. 1999, Ware 2000).

- Increased resources: high-bandwidth hierarchical interaction, parallel conceptual processing, offload of work from cognitive to perceptual system, expanded working memory and expanded storage of information.
- Reduced search: locality of processing, high data density and spatially-indexed addressing.
- Enhanced recognition of patterns: recognition instead of recall, abstraction and aggregation, visual schemata for organization, and enhanced patterns and trends.
- Perceptual inference: visual representations make some problems obvious, and complex specialized, graphical computations can be enabled.
- Perceptual monitoring: visualizations can allow monitoring of a large number of potential events.
- Manipulation medium: visualizations can allow exploration of a space of parameter values and amplify user operations.

3. ADVANCED VISUAL ANALYTICS CONCEPTS AND TECHNIQUES

This section presents some VA concepts applied to various information types. Research results are presented to illustrate the use of VA techniques.

3.1. Graph, Link and Network Visualization

In network information, be it a graph or a hierarchical representation, the relevant aspect is mainly the links between the data elements. However, in many situations, the number of links can quickly grow and result in an undecipherable mesh of data relationships. Innovative visualizations are required to untangle these spider webs of links and make sense of the information presented.

NodeTrix is a hybrid approach to social network visualization (Henry et al. 2007). Node-link diagrams are used to show the global structure of a network, while arbitrary portions of the network can be shown as adjacency matrices to better support the analysis of communities. It is especially useful in the case of globally sparse but locally dense social networks (Figure 5).

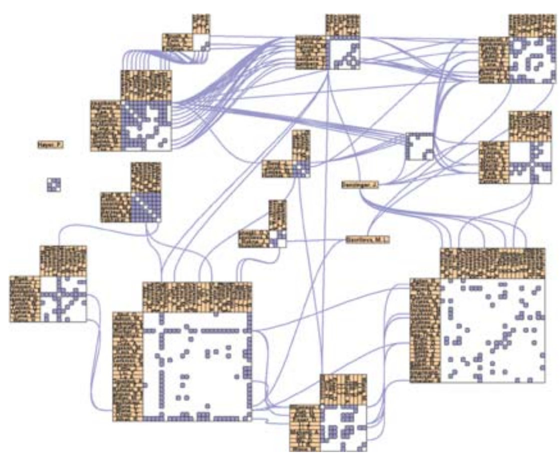


Figure 5: NodeTrix social network visualization (Henry et al., 2007).

3.2. Temporal Visualization

Temporal data analysis is useful to detect trends and recurring events over time. The study of event sequences also enables the identification of links between individual observations and possible causes for some events. The use of timelines provides an overview of what happened in a given time interval while a time slider can be used to show unfolding events in ascending or descending time order. Time sliders are especially effective when we are faced with multiple dimensions that must be represented, as is the case when combining geospatial and temporal data, for example.

Lifelines 2 (Wang 2010) is an interactive visualization tool that organizes electronic record information in a temporal display to allow the discovery of patterns across multiple records, hypothesis generation and finding cause-and-effect relationships in a population (Figure 6).

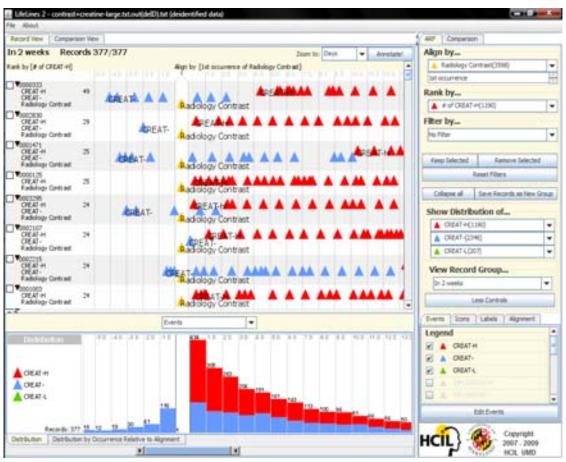


Figure 6: Lifelines2 visualization of multiple electronic health records (Wang 2010).

3.3. Hierarchical Display

Many hierarchical datasets can be displayed with the techniques used for graph and network data. However, some visualizations such as treemaps, take advantage of the specific characteristics of hierarchical datasets.

Treemaps show data as a set of rectangles where the smaller rectangles contained inside a larger rectangle are subranches. In Figure 7, a market tree map visualization shows the change is various financial areas for over 500 stocks. It is a tool that can be used to spot trends and investment opportunities. The rectangle size is proportional to the company's market cap while the color shows price performance. The overview that this treemap visualization offers makes it obvious that on November 23, 2010, the economy was not going very well in the US. Each region can be selected to view more details about it. For example, the very light green rectangle in the bottom left part is the New York Times stock and it increased by 6.7%.While most treemaps are based on rectangular regions, other space division schemes were explored such as circular treemaps (inefficient use of space) and Voronoi treemaps.

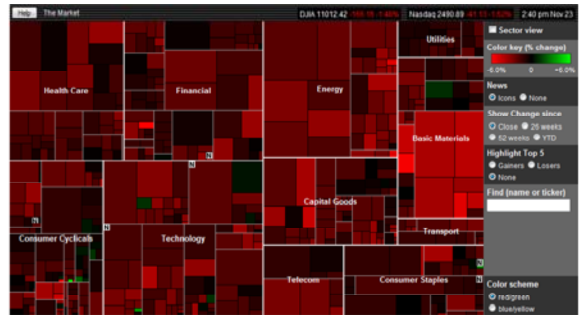


Figure 7: SmartMoney's treemap view of markets on November 23, 2010 (SmartMoney 2010).

3.4. Multivariate Visualization

Datasets with more than 3 dimensions can be very difficult to represent. A few techniques were developed in order to reduce the dimensionality of the data while being able to show the important characteristics of the datasets.

The Dust & Magnets metaphor represents individual cases as particles of iron dust, and dataset variables are represented as magnets (Soo Yi et al. 2005). This enables the user to manipulate the magnets and see the dust particles move accordingly. When a magnet is dragged, individual dust particles are attracted to the magnet based on the value of the attribute corresponding to the magnet. The dataset characteristics are exposed through the interaction with the magnets, enabling the user to get a feel of the importance of each variable and of the relations between them (Figure 8).

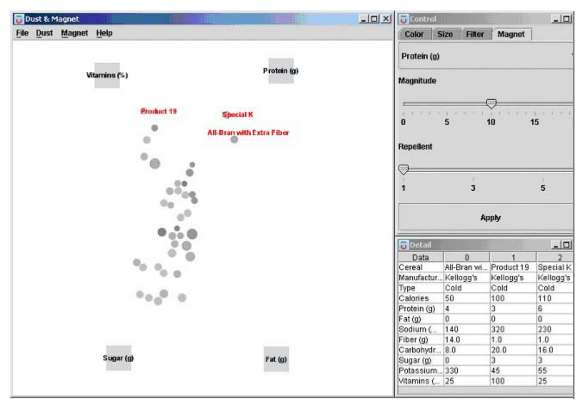


Figure 8: Dust & Magnet example using a cereal dataset (Soo Yi et al. 2005).

3.5. Validation and Evaluation

Evaluation of user interfaces is very challenging. Relatively little effort has been devoted to this aspect up to now in comparison to the work done in the development of tools. As more and more visual analytics applications appear, researchers are looking for a better assessment of their effectiveness and utility. The ultimate goal is the creation of a set of metrics that could predict the efficiency of tools for given tasks.

While waiting for these metrics to be defined, we can use a number of alternative methodologies and approaches. It is possible to classify them in two categories: analytic and empirical evaluations. Analytic evaluations such as heuristic inspection and cognitive walkthrough are based on formal analysis models and conducted by experts. Empirical evaluations consist of studies where measurements take place in controlled experiments and qualitative studies where data is gathered with focus groups or interviews.

The Scientific Evaluation Methods for Visual Analytics Science and Technology (SEMVAST) project focuses on two activities: making benchmarck data sets available through the Visual Analytics Benchmarks Repository and seeding an infrastructure for evaluation. The VAST Challenge serves as a testbed for these activities (see section 5.3).

4. DEFENCE AND SECURITY DOMAINS

4.1. Cyber Security

As the modern world is relying heavily on computers and networks to conduct day-to-day activities, these computers and networks have become increasingly a target of choice for countries conducting spying or disruptive operations, terrorist and criminal organizations or simply hackers. The impact of cyber attacks on a country, an organization or individuals can be severe and costly. Moreover, network attacks are increasingly sophisticated and unpredictable.

Good security tools are necessary to properly manage computer networks, prevent and detect intrusions. This includes tools to analyze service usage in a network, detect a distributed attack, and investigate hosts in a network that communicate with suspect external IPs. One key requirement for these tools is the ability to process and filter massive amounts of information.

Visual analytics techniques have been explored and put in service to counter cyber security. “Visualization is often appropriate when human intelligence and domain knowledge must be combined with automated methods. This is certainly the situation with monitoring and exploring network traffic patterns. The sheer number of alerts and the sophistication of attacks requires a symbiosis of Intrusion Detection Systems (IDS) algorithms and human analysis to fight new adversaries” (Mansmann et al. 2009).

D’Amico et al. identified a list of analysis tasks related to cyber security (Figure 9). All of these tasks can involve VA. Globally, VA can improve cyber security with capabilities to (Wolf 2009):

- Recognize risks and protect against cyber threats, allowing for more effective attack prevention and faster isolation and mitigation of attacks;
- Enable key aspects of the digital forensic process, including data collection, discovery, investigation, examination, analysis and reporting capabilities for information discovery;
- Allow information discovery, processing and visualization (tactics which apply across many applications for computer security and forensics).

“Incorporating visual analytics into an organization's best practices allows computer security professionals to quickly identify threats to their own organizations. By doing so earlier and more comprehensively than their competitors, this leads to significant competitive advantage in the face of increasing threats and daily attacks” (Wolf 2009).

Triage analysis	• Weed out false positives • Escalate suspicious activity for further analysis
Escalation analysis	• Analyze data over longer time than Triage • Incorporate multiple data sources (more than Triage)
Correlation analysis	• Look for patterns and trends • Assess similarity to related incidents – internal & external
Incident response	• Recommend, implement Courses of Action • Support law enforcement investigation
Malware analysis	• Reverse-engineer malware • Develop defenses against malware
Forensic analysis	• Collect and preserve evidence • Support law enforcement investigation
Threat analysis	• Characterize attackers: identification, modus operandi, motivation, location
Vulnerability analysis	• Identify and prioritize vulnerabilities • Manage remediation of vulnerabilities
Sensor management	• Develop signatures, tune sensors • Modify placement of sensors

(from 2005 D'Amico & Whitley CTA, and other Secure Decisions decision analyses)

Figure 9: Cyber security analysis (D’Amico 2011).

The visual analytics applications used in cyber security are mostly related to network analysis. The NFlowVis Network visualization tool provides a number of views used to perform large-scale network traffic monitoring, to detect distributed attacks and to analyse intrusion detection events. For example, as shown on Figure 10, a TreeMap is used to represent compromised hosts in the center of the display and selecting attacking hosts arranged at the borders of the display. Figure 11, is a graph visualization showing communication flows between source and destinations hosts.

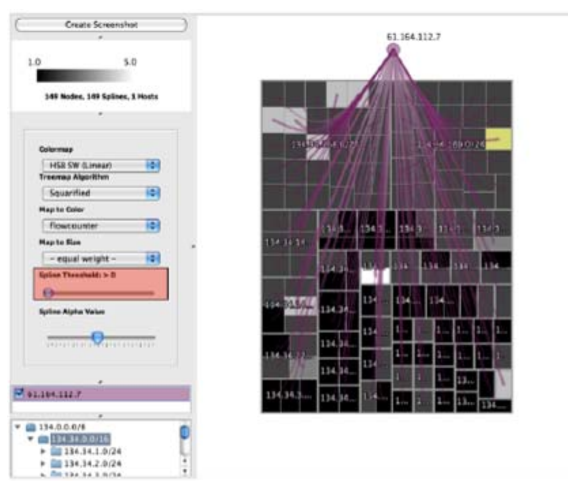


Figure 10: Example of NFlowVis showing the identification of compromised hosts using threshold adjustments (Mansmann et al. 2009).

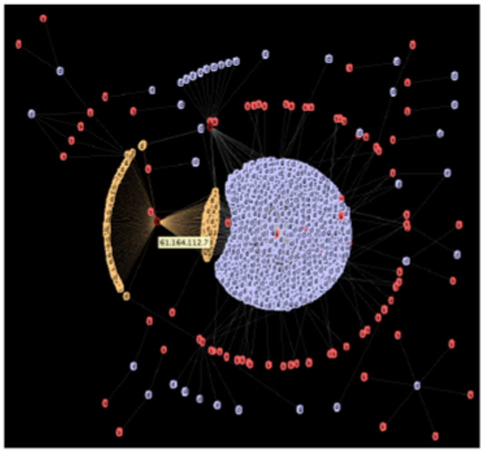


Figure 11: Example of NFlowVis showing communication flows between source (red) and destinations hosts (blue) (Mansmann et al. 2009).

ManyNets is a network visualization tool with tabular interface designed to visualize up to several thousand network overviews at once (Figure 12). This allows networks to be compared and large networks to be explored using a divide-and-conquer approach. A collection of networks is presented in a table, where each row represents a single network. Columns represent statistics, such as link count, degree distribution, or clustering coefficients. Networks can also be subdivided and compared based on motifs (small patterns of connectivity), clusters, or network-specific attributes (Sopan et al. 2010).

Figure 12: ManyNets is a tool for the simultaneous visualization of many networks (Sopan et al. 2010).

When it comes to forensic activities, history trees can be useful for providing traceability and the history of the analyst's workflow (Figure 13).

Figure 13: History trees (Endert et al. 2009).

Starlight Visual Information System is a visual analytics platform where viewers can interactively move among multiple representations of the information. Starlight can be used for cyber security and computer forensics and the use of Starlight for cyber network analysis is depicted in Figure 14.

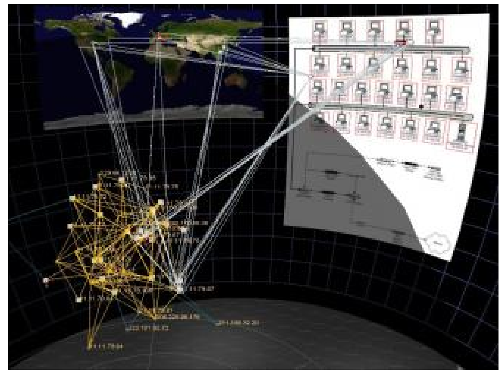


Figure 14: Use of Starlight for cyber analysis (Future Point Systems 2011).

4.2. Intelligence and Counterterrorism

The purpose of intelligence is to provide commanders and staffs with timely, relevant, accurate, predictive, and tailored intelligence about the enemy and other aspects of the area of operations. Intelligence supports the planning, preparing, execution, and assessment of operations (FM 2-0 2010).

In modern conflicts, Intelligence activities are more and more concerned with Counter-Insurgency (COIN). COIN is defined as "a part of a wider set of irregular activities and threats to a secure and stable environment", where an irregular activity may be defined as: "behaviour that attempts to effect or prevent change through the illegal use, or threat, of violence, conducted by ideologically or criminally motivated non-regular forces, groups or individuals, as a challenge to authority" (DND/CF 2008).

"Terrorism is a global threat influencing the attitude and behaviour of a target group by threatening, or carrying out, devastating actions. These actions, as we have seen, can and do include the use of conventional weapons, biological, chemical, or nuclear agents. Today's terrorism also threatens our economic and information resources. Our vulnerability to terrorist attacks expands with our growing reliance on information technologies. Increased access to information and the centralization of vital components of local, national, and global infrastructure threaten both local and national security" (Visual Analytics Inc 2011).

Inter-agency communication and collaboration is essential to investigate terrorist groups. Law enforcement and intelligence agencies need to work together to collect and analyze data from multiple data sources in order to monitor, penetrate, infiltrate, and prevent terrorist activity. A particular attention is needed on preparation activities such as money transfers, material purchases and personnel movement.

"Actionable intelligence is essential for preventing acts of terrorism. The timely and thorough analysis and dissemination of information about terrorists and their activities will improve the government's ability to disrupt and prevent terrorist acts and to provide useful warning to the private sector and our population" (Bush 2002).

In order to solve complex, multifaceted, real-world problems, intelligence analysts need to develop an understanding of various collections of data and link together information of different types. The use of Starlight (see section 4.1) for intelligence analysis is depicted in Figure 15. This platform enables the visualization of multiple data collections simultaneously in order to uncover correlations that may span multiple relationship types, including networks, geographical data and textual information.

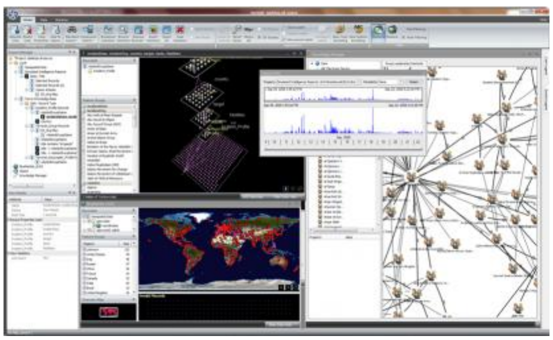


Figure 15: Use of Starlight for intelligence analysis (Future Point Systems 2011).

Analysts are also often faced with large collections of unformatted text documents. IN-SPIRE is a text analysis and visualization software that can quickly reveal important information from these datasets and accelerate subsequent investigation and discovery. IN-SPIRE's two main visualizations display representations of the documents in which those with similar or related topics appear closer together (Figure 16). In the Galaxy visualization (upper right), dots represent documents and cluster around center points that represent central topics or themes. In the ThemeView visualization (lower right), users see a relief map where the highest peaks represent the most prevalent topics in the collection.

Oculus nSpace is a web browser-based system of systems for intelligence analysis meant to support multiple analytical styles and workflows (Wright et al. 2006). It is the combination of two capabilities called: TRIST and Sandbox (see Figure 17). TRIST's multi-dimensional linked views help users find relevant documents (unstructured text, images, videos, etc) from web services. Queries can be saved and scheduled to be executed repeatedly. The Sandbox is a space where relevant information can be dropped and where analytical sense making happens. Elements can be grouped and collapsed. Graphs and networks are supported and matrices allow analysis of competing hypotheses using groups of evidence. The Sandbox supports flexible visual cognition through spatial arrangement.

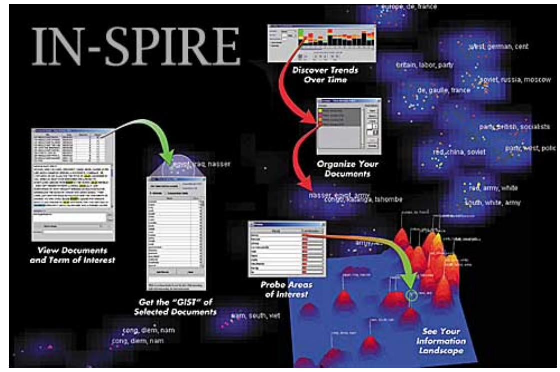


Figure 16: The IN-SPIRE discovery tool integrates information visualization with query and other interactive capabilities (PNNL 2011).

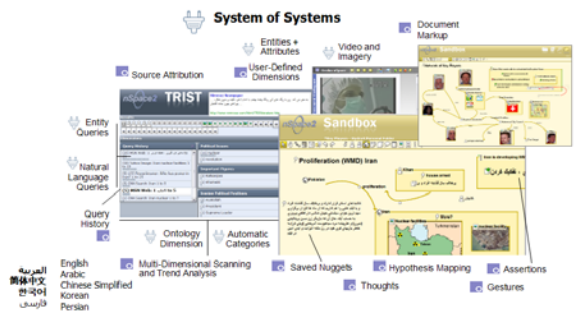


Figure 17: Oculus nSpace TRIST and Sandbox capabilities (Wright et al. 2006).

As part of the National Consortium for the Study of Terrorism and Responses to Terrorism (START), tools have been developed to consult the Global Terrorism Database. This database holds information about successful and missed terrorism attacks across the world since the 1970s.

Figure 18 shows a Theme River representation of terrorism attacks in the world over time. Stripes of different colors show how many terrorist attacks occurred in the corresponding country each year. The number of attacks is represented by the thickness of the stripe at the year on the x-axis The black stripe (within the yellow stripes) has been selected, corresponding to terrorist attacks in India (Lee 2008).

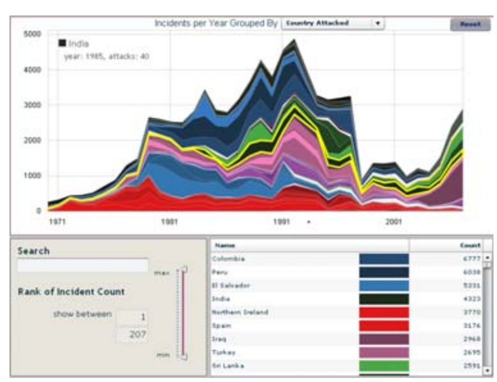


Figure 18: Theme River representation of terrorism attacks in the world over time (Lee 2008).

The Basic Ordinance Observational Management System (BOOMsys), a prototype geovisual application that depicts spatiotemporal data about Improvised Explosive Device (IED) attacks in Iraq (Figure 19). BOOMsys shows the number of IED attacks aggregated by province using circles scaled to the data. Each symbol can be probed to display background information for the province. It also includes tools to explore the data by various time frames, including totals by day, week, month, and a composite week by day.

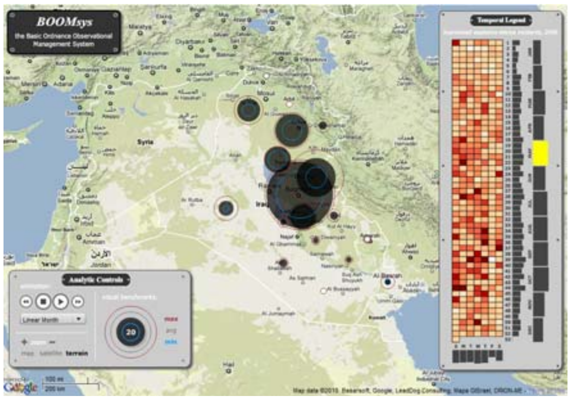


Figure 19: Analysis of IEDs in Iraq with BOOMsys (GeoVISTA 2011).

5. RESOURCES

5.1. Visual Analytics Community Resources

Visual analytics is not only a research field, it is also a community. The Visual Analytics Community (VAC) is comprised of people and organizations from around the world who are stakeholders in the advancement and application of visual analytics technology. The VAC brings together diverse individuals and organizations from academia, government, and industry. At the core of this community is the VAC Consortium.

It is an international consortium of individuals, institutions, and government agencies that acts as focal point and champion for addressing the diverse needs of the Visual Analytics Community, including user, research and application development, business, and educational needs. A consortium meeting is held annually. The event includes talks from government and industry experts, research highlights, product demonstrations, and discussions with leaders in the field of visual analytics.

VAC Views is a biannual publication that provides information about different visual analytics applications, outreach efforts including recent and upcoming conferences and workshops, and educational highlights.

The visual analytics community social website, vacommunity.org, is meant to act as a coordination point where each community of research and practice has access to dedicated blogs, forums, wikis, calendars, RSS feeds for information distribution, and other social media services. The website theiVAC.org provides formal news, information, and articles on visual analytics to broaden the outreach of the Integrated Visualization and Analytics Community (iVAC).

5.2. Learning More

Two very useful resources for learning more about visual analytics are the InfoVis:Wiki project (InfoVis:Wiki 2011) and the Visual Analytics Digital Library (VADL 2011).

5.3. VAST Challenges

The VAST Challenge is an original mean of gathering visual analytics developers and researchers around shared benchmark datasets and comparing new visual analytics tools. It is proposed every year to the international VA community as part of the IEEE VAST Symposium. Participants can demonstrate the visual analytics capabilities of their tools against an invented scenario and synthetic datasets. The challenge scenario and datasets then remain available to anyone interested in testing their VA tools with representative tasks and datasets. Figure 20 shows visualizations that were submitted to the VAST 2008 Challenge.

The VAST Challenge 2011 includes a cyber security mini-challenge involving situational awareness in computer networks.

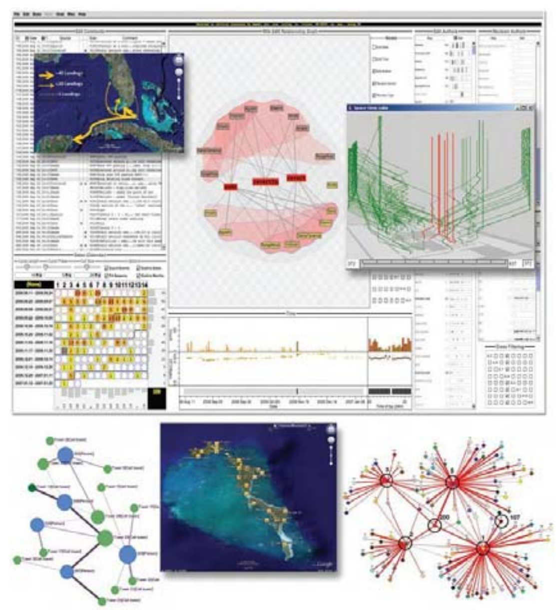


Figure 20: Visualizations submitted to the VAST Challenge 2008, from (clockwise, from upper left): SPANDAC, NEVAC, Fraunhofer Institute, Oculus Info Inc., Palantir, Oculus Info Inc. (VAC Views, 2009).

5.4. VizSec Conference

The International Symposium on Visualization for Cyber Security (VizSec) is a forum that brings together researchers and practitioners from academia, government, and industry to address the needs of the cyber security community through new and insightful visualization techniques. VA tools and techniques have also been presented at this conference over the last few years. Research projects that incorporate multiple data sources, such as network packet captures, firewall rulesets and logs, DNS logs, web server logs, and/or intrusion detection system logs, are a key focus of this conference.

6. CONCLUSION

In the context of modern defence and security operations, analysts are faced with significant data overload problems which prevent them from understanding a situation at hand and anticipating how this situation may develop. Fortunately, VA has emerged as a significant multi-disciplinary research field that leverages the human cognitive abilities to comprehend information when presented in a proper way and combined with suitable interaction. To describe VA, this paper has presented a number of information visualization, interaction and analytical reasoning techniques that allow making the relevant information more salient in order to help detect patterns, trends and anomalies.

VA is making its way into defence and security applications such as cyberspace management and intelligence for counterterrorism. Examples of this are: performing large-scale network traffic monitoring and intrusion detection in the cyberspace domain; and making sense out of large collections of unformatted text documents for counterterrorism intelligence analysis. VA has a significant momentum and VA research and applications have been growing exponentially over the last years. Several VA applications have become available commercially, various key communities have been stood up and significant initiatives have been undertaken.

ACKNOWLEDGMENTS

The authors wish to thank DRDC CSS (Centre for Security Sciences), in particular Mr Andrew Vallerand and Mr Jack Pagotto, for their indefectible interest in R&D work in Visual Analytics and for allowing the authors to engage with the VA community.

REFERENCES

Bush, President G. W., 2002. The Department of Homeland Security, June 2002, http://www.dhs.gov/xlibrary/assets/book.pdf, retrieved February 2011.

Card, S.K., Mackinlay, J.D. and Shneiderman, B., 1999. Readings in Information Visualization: Using Vision to Think, Morgan Kaufmann Publishers, San Francisco, 1999.

Card, S. K., Robertson, G. G. and Mackinlay, J. D., 1991. The Information Visualizer: An Information Workspace, In Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'91), pp. 181-186, ACM Press, New York.

D'Amico, A., 2011. Visual Analytics for Cyber Defense Decision-Making, VAC Meeting 2011, University of Maryland, Inn and Conference Center, 3-4 May, 2011.

D'Amico, A., Whitley, K., Tesone, D., O'Brien, B. and Roth, E., 2005. Achieving Cyber Defense Situational Awareness: A Cognitive Task Analysis of Information Assurance Analysts, Proc. of the Human Factors and Ergonomics Society 49th Annual Meeting, 2005, pp. 229-233.

DND/CF, 2008. Counter-Insurgency Operations B-GL-323-004/FP-003, 2008.

Endert, A., Fink, G. A. and North, C. 2009. Visual Analytics for Cyber Security: Observations and Opportunities for Large Display Workspaces, Usable Security, CS 6204, Fall 2009, Virginia Tech.

FM 2-0, 2010. Intelligence, Headquarters Department of the Army, Washington, DC, 23 March 2010.

Future Point Systems, Inc, 2011. http://www.futurepointsystems.com/, retrieved May 2011.

GeoVISTA, 2011. http://www.geovista.psu.edu/BOOMsys/, retrieved May 2011.

Healey, C. G., 2009. Perception in Visualization, last updated 11 May 2009, Christopher Haley's web site: http://www.csc.ncsu.edu/faculty/healey/PP/index.html, retrieved September 2009.

Henry, N., Fedeke, J.-D. and McGuffin, M., 2007. NodeTrix: a Hybrid Visualization, IEEE TCGV, Proceedings of Information Visualization 2007.

InfoVis:Wiki, 2011. http://www.infovis-wiki.net/index.php/Visual_Analytics, retrieved January 2011

Koffka, K., 1935. Principles of Gestalt Psychology, Harcourt Brace, New York.

Kosara, R., Miksch, S. and Hauser, H., 2001. Semantic Depth of Field, Proceedings of IEEE Symposium on Information Visualization (InfoVis 2001), IEEE, p. 97-104.

Lee, J., 2008. Exploring Global Terrorism Data: A Web-based Visualization of Temporal Data, ACM Crossroads, 15, 2, p. 7-16, 2008, http://www.start.umd.edu/datarivers/JoonghoonLee.pdf, retrieved February 2001.

Mansmann, F., Fisher, F., Keim, D.A. and North, S. C., 2009. Visual Support for Analysing Network Traffic and Intrusion Detection Events using TreeMap and Graph Representations, CHiMiT '09 Proceedings of the Symposium on Computer Human Interaction for the Management of Information Technology.

Miller, R. B., 1968. Response Time in Man-Computer Conversational Transactions. Proc. AFIPS Fall Joint Computer Conference Vol. 33, pp. 267-277.

PNNL, 2011. http://in-spire.pnnl.gov/, retrieved May 2011.

Shneiderman, B., 1996. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations, Proceedings of the IEEE Symposium on Visual Languages, IEEE Computer Society Press, pp. 336-343.

Simons, D. J. and Chabris, C. F., 1999. Gorillas in Our Midst: Sustained Inattentional Blindness for Dynamic Events, Perception, 28, p. 1059-1074.

SmartMoney, 2010. http://www.smartmoney.com/map-of-the-market/, retrieved November 2010.

Soo Yi, J., Kang, Y. A., Stasko, J. and Jacko, J. A., 2007. Toward a Deeper Understanding of the Role of Interaction in Information Visualization, IEEE Transactions on Visualization and Computer Graphics (TVCG), 13(6). Presented in InfoVis 2007, Sacramento, California, October 28 - November 1, pp. 1224-1231.

Soo Yi, J., Melton, R., Stasko, J. and Jacko, J., 2005. Dust & Magnet: Multivariate Information Visualization using a Magnet Metaphor, Information Visualization, 4(4), p. 239-256.

Sopan, A., Freire, M., Plaisant, C., Golbeck, J. and Shneiderman, B., 2010. Distribution Column Overviews in Tabular Visualizations, April 2010, HCIL-2010-01.

Stone, M., 2006. Choosing Colors for Data Visualization, January 17, 2006, http://www.perceptualedge.com/articles/b-eye/choosing_colors.pdf, retrieved March 2010.

Thomas, J.J., 2009. Visual Analytics Techniques that Enable Knowledge Discovery: Detect the Expected and Discover the Unexpected, June 28, 2009, ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery (VAKD '09), Paris, France.

Thomas, J.J. and Cook, K.A., eds., 2005. Illuminating the Path: The Research and Development Agenda for Visual Analytics, IEEE Computer Society Press, http://nvac.pnl.gov/agenda.stm#book, retrieved November 2010.

Tufte, E., 1990. Envisioning Information, Graphics Press, 1990.

VAC Views, 2009. The VAST 2008 Challenge, May 2009, p.14-15, http://nvac.pnl.gov/vacviews/, retrieved May 2011.

VADL, 2011. http://vadl.cc.gatech.edu/, retrieved January 2011.

Visual Analytics Inc, 2011. http://www.visualanalytics.com/, retrieved May 2011.

Wang, T.D., 2010. Interactive Visualization Techniques for Searching Temporal Categorical Data, Ph.D. Dissertation from the Department of Computer Science, University of Maryland, May, 2010.

Ware, C., 2000. Information Visualization – Perception for Design, Academic Press, 2000.

Wertheimer, M., 1938. Laws of Organization in Perceptual Forms, in A Source Book of Gestalt Psychology, W. D. Ellis (ed), p. 71-88, Harcourt Brace.

Wolf, J. 2009. How to Improve IT Cyber-Security with Visual Analytics, IT Security & Network Security News, eWeek.com, http://www.eweek.com/c/a/Security/How-to-Improve-IT-Cyber-Security-with-Visual-Analytics/, retrieved June 2011.

Wright, W., Schroh, D., Proulx, P., Skaburskis, A. and Cort, B., 2006. The Sandbox for Analysis - Concepts and Methods, ACM CHI, 2006.

AUTHORS BIOGRAPHY

Valérie Lavigne

Ms Valérie Lavigne obtained a Master degree at Université Laval in electrical engineering in 2005. She worked as a consultant in various scientific and defence application domains (optronic surveillance, electro-optical warfare, image processing, hyperspectral imaging, modeling and simulation) before joining Defence R&D Canada – Valcartier in 2008. She is now a member of the Innovative Interfaces and Interactions Group within the Intelligence and Information Section. She is now leading an applied research project on interactive visualization and collaboration for maritime domain awareness. She also contributes to R&D activities concerning social network analysis for counter-insurgency operations, intelligence preparation of the battlefield and virtual intelligence analysis capability. Her research interests include visual analytics, multimodal human-computer interactions, adaptive interfaces and collaborative technologies.

Denis Gouin

Mr Denis Gouin holds a bachelor degree in Computer Sciences. He has been working as a defence scientist with Defence R&D Canada – Valcartier since 1978. Throughout his career, he has been actively engaged in developing novel concepts and prototypes of command and control and intelligence information systems for defence and security applications. His fields of interest include geomatics, human-machine interfaces, information visualization, collaboration environments, enterprise portal technology and open-system architectures. He has led the design and development of the Open Geospatial Datastore Interface (OGDI) which is an open infrastructure that provides an open access to geospatial information. He was the project manager for the COP 21 Technology Demonstration which has produced and demonstrated a prototype of a Situation Awareness Knowledge Portal. Mr Gouin was the lead of the Situation Awareness sub-project, as part of the Joint Command Decision Support (JCDS) 21 technology Demonstration. Finally, he participates as the Canadian national lead to the TTCP C3I Technical Panel 2 on Command Information Interfaces. Mr Gouin is now the lead of the Innovative Interfaces and Interactions Group. He is involved in a number of R&D projects that exploit visual analytics and information visualization, applied to maritime domain awareness, intelligence preparation of the battlefield and social-network analysis in counter-insurgency operations.

IS LVC A PANDORA FOR ALL TRAINING NEEDS ...?

Ian McVeigh[(a)], Jon Sackett[(b)]

[(a),(b)]AgustaWestland

[(a)]ian.mcveigh@agustawestland.com, [(b)]jon.sackett@agustawestland.com

ABSTRACT

The military have been encouraging the development of training assets, particularly platform simulators that can be linked to create large scale virtual training exercises. Interoperability standards such as DIS and HLA have been created and a number of exercises conducted, realising considerable savings against the large scale deployment of live assets. Appended training systems have also advanced over the years enable live assets to train together realistically and safely in monitored environments improving training effectiveness. However, there has until recently been little demand for the integration of the Live, Virtual and Constructive domains, despite the advantages of force multiplication, tempo and realism (especially in communications). With even more emphasis on costs saving, coupled with the continued advance in technology, the military are now turning to industry to improve interoperability still further to make maximum use of all available assets.

Keywords: workstation design, work measurement, ergonomics, decision support system

1. ROTARY WING INTEGRATION

Rotary wing offers an interesting example in the LVC domain. Helicopters operate across the land, sea and air environments, are operated by all services and in many case by one service on behalf of another (e.g., the Air Force providing tactical lift in a battlefield environment).

However, they are rarely truly integrated into any combined force training other than as occasional role players – largely down to cost. Role playing without training benefit is not cost effective and deployment costs are also high.

Therefore, LVC offers a considerable training advantage that the military may now be prepared to adopt on a larger scale.

AgustaWestland has conducted a number of studies and self funded demonstrations into the technology necessary to delivery truly integrated rotary wing training environments :

- Apache Live-Virtual Integration: A successful demonstration has been conducted by Agust aWestland with Aviation Training International Limited (ATIL) and The Boeing Company, integrating the live training and virtual training environments, using Apache AH Mk1 Collective Training System (CTS) & the ATIL Attack Helicopter simulator network. This highlighted benefits in force multiplication and tempo, along with reducing environmental demands and the pressure on the training fleet by lessening the impact of live aircraft. Other applications include the integration of UAV's into the live training domain, via their ground stations.
- Ship borne Helicopter: AgustaWestland has also conducted technical feasibility studies into the stimulation of helicopter sonar systems, on the ship and airborne, to enable training with constructive targets or with submarine simulations, classroom trainers or embedded trainers. The data is passed to the aircraft via off board stimulators when on the deck or the ships Data Links when airborne.
- Air/Land Integration: Study activity has taken place with UK MoD and Industry to integrate t he Apache AH Mk1 CTS with training ranges and the UK's fxed wing Rangeless Airborne In strumented Debriefing System (RAIDS). This would enable AH and the strike aircraft to train together, including the helicopter designating t argets for fixed wing attack.

2. LVC - A ROTARY WING CASE STUDY

In addition to rotary specific applications, AgustaWestland has also assisted the UK MoD in demonstrating linked virtual training exercises (the Combined Arms Tactics Trainer) with the live domain (Salisbury Plain Training Area), using the UK's tactical communication Bowman network.

This paper takes the Apache Live-Virtual integration study and looks at some of the lessons learned.

2.1. The Exercise

In 2004 AW, Boeing and ATIL developed a training scenario to demonstrate the practicalities of LVC, based on a 4 ship exercise. The demo had 3 simulators on a wide area network, linked to an instrumented live aircraft, which had a training mod to allow for simulated weapons effects.

The demo was a Sal missile engagement with both live and virtual entities targeting 3 live vehicles at Middle Wallop airfield. The networked simulators (at Dishforth, Wattisham and Middle Wallop) were stationed on HMS Ocean in Southampton water and the live aircraft based at Middle Wallop.

All the aircraft departed their bases, approach march and flight into a holding area just south of the target area, before moving to the battle position, at which point a series of engagements started.

The first engagement (Sal missile engagement - handoff) had a virtual aircraft (FMS) designating for a live aircraft, targeting the furthest of the instrumented targets (now a virtual entity within the simulated environment). The FMS selected the target, confirmed and passed the designation data to the Live aircraft so the live aircraft could run through a remote Sal engagement, once the live aircraft had confirmed the designation code, a virtual simulated missile was fired against the Live target. During the engagement the FMS had to ensure it achieved the critical illumination period to ensure the missile found its target and on impacts the Live target registered a hit and flashed its instrumentation to indicate to the live aircraft the targeting was successful destroyed. At the same time all the simulators could see a constructed vehicle on fire following a successful engagement.

Second engagement, the roles were now reversed with the Live aircraft designating for the FMS and the next target selected. This created a new set of issues, with the Live aircraft now having responsibility for achieving the critical illumination period so the FMS could conduct a successful engagement of the target.

Following the successful completion of both engagements the live aircraft returned to base, prior to a full after action review for the 4 aircraft (3 sim's and 1 live). During debrief the following issues were identified: tempo, data bases / entities, simulation within the cockpit and aircraft qualification.

2.2. The Results

Exploring these issues in detail:

- Tempo the exercise had been run using the simulators the day prior to the demo to establish the schedule for the main event and to esure that critical events took place when advertise. However during the live demonstration that schedule was significantly impacted by the effects of the LIVE aircraft and the diferences between simulated flying and live aircraft manoeuvring, which created a significant delay to the overall event. So if time critical events are a key part of the training requirement, then care must be taken to ensure a realistic duration is achieved .. Adding a live entity into the exercise may enhance the training fidelity and ultimately the effect
- Data bases / Entities – The resolution of the terrain data base and the location of imported entities coursed a few issues .. we had vehicles appearing semi submerged or hovering above the ground, but this was easily resolved by tagging the entity to ground. More significant were the impacts of visibility, with the live aircraft having its view of the targets obscured and a revised fire position required prior to the engagement, even though the simulators had clear line of site at all times.
- Simulation within the cockpit – The crew of the live aircraft committed that the exercise would have been more realistic had they been able to see the simulatd aircraft alongside them during the engagement, even though the 4 aircraft could communicate clearly they still felt the training lacked a level of realism from not having tht capability.
- Aircraft qualification – Simulated training systems on board Live helicopters is still not widely available, its more common in fixed wing and for land forces, but to gain the full potential out of systems available today, a base line configuration should be established which provides the building block for training systems development. How do we partition onboard software, yet still display realistic systems / effects. How do we manage growth, without the massive expense of aircraft requalification?

3. SIMULATION INTEGRATION

We have learned much from the work completed to date, and the technology to achieve integration has been available for some time, but very little live-virtual integration has actually been achieved. There have been many reasons for this temporary stagnation, but this is starting to change. Some of the key issues listed below, focused on the UK, but with similar observations across geographies:

- A focus on the more immediate training need for operations, with some limited success in integrating virtual assets for specific tactical training needs such as the Military Training thrugh Distributed Simulation (MTDS) programme in the UK which has matured from an experimental into and operational programme (Distributed Synthetic Air Land Training – DSALT) with plans to develop further through the new Defence Operational Training Capability (DOTC). This has a stated objective to lay foundations for full LVC post 2020, paving the way for new and innovative solutions to be realised.
- A lack of budget to integrate training assets. The budgets for training have traditionally been associated with and individual platform or groups of platforms. The integration of these assets has required the individual programmes to effectively give up part of their training budgets which they have been reluctant to do. Interestingly it is the current lack of budget for costly all live training that is causing the military to look again at simulation, instrumentation of live assets and integration of training assets to deliver a collective training benefit. It is

recognised now that to achieve this, budgets need to be allocated for integrated collective training as well as individual. By illustration, the UK MoD has recently conducted an Industry consultation on this issues and international military presenters at the recent TEC event sited networked training as something we have talked about for a long time and now need to do.

- Maturity of Technology, especially in the live domain where the integration of laser and radio based training systems with the aircraft has often beenfinancially prohibitive, especially where such instrumentation would primarily benefit the land forces training with the airborne platform rather than he aircrew specifically. However, this too is changing, with new systems available, such as cards to incorporate into on board processors (as opposed to bespoke Line Replaceable Units-LRUs) and light weight sensors using radio wave networks to replace hard wiring.
- Industrial Inertia. The incumbent defence simulation providers have stagnated in terms of the level of technology and innovation they can provide o the Military. The commoditisation strategy must be pursued even harder to deliver true value for money. Part of this solution must include the wholesale adoption of gaming technologies which can provide a level of fidelity unmatched by the traditional options – gaming technologies that are a result of tens of billions of revenues each year globally and themselves drive the adoption of common standards and tools. In order to achieve the savings, investments must be made in the correct areas to allow adoption of the technology. In parallel, military procurement organisations should pursue a "Small & Medium Enterprise" policy that ensures a certain proportion of overall funding is spent with agile, innovative organisations, unhindered by bureaucracy, who can raise the bar to the larger organisations.

4. OPEN STANDARDS AND COMMON DATA BASES

LVC integration will depend upon the cost-effective application of common standards and common databases. These are maturing and being used more widely as more implementation tools are becoming available moving away from the traditional domain of the large primes.

For LVC applications the most significant development has been in the production of simulator databases from real-world data, using satellite imagery. This has a number of advantages to Live – Virtual integration:

- Ability to align the real and simulated world so that all players have the same view of the world and other players position init.
- Real world Maps can be used by all players, as opposed to maps derived from databases that all take a slightly different view of the same world.
- Rapid Database development of detailed terrain areas for specific training purposes which should be the same for all players.

Figure 1

The continued improvement in graphics processing power will also help by enabling the processing of detailed databases for all players. Such detail may not be required at the individual training level, just for collective training end especially live-virtual integration emphasising the need to budget at the Macro level to ensure that each platform system is properly equipped.

5. AIRCRAFT INTEGRATION

The integration of the necessary embedded and appended simulation equipment is the most costly aspect of LVC and therefore the most difficult to justify on a cost basis. However, without it, collective training exercises lose significant value with aircraft being invulnerable to ground fire and unable to engage targets in a realistic and meaningful way. As helicopters in particular are increasingly becoming an essential integrated component for most operations it is time to review this situation as we call more and more to 'train as you fight and fight as you train'.

Fortunately technology has moved on, enabling a lower cost approach to be taken to integration with more effective solutions. Some examples are given below:

- Lightweight laser sensors: It is now becoming possible to move away from large pods hardwired into the aircraft systems and to use lighter laser sensorsthat communicate using radio based systems.
- DAS stimulation: Card based solutions are available to stimulate the aircraft DAS from simulated threats either produced from within the systemor from externally simulated entities
- Helmet projector systems: The quality and capability of these systems has also improved to the point now where simulated targets can be practically preented to the aircrew flying in the live assets, overcoming many of the limitations experienced during our previous demonstration.
- Radar and Sonar stimulation: Technical feasibility studies completed by AgustaWestland and Lockheed Martin have shown that on-board systems may be stimulated to provide training in the live environment whilst on the ground or airborne. These systems can in turn be networked to virtual and constructive entities. This could be of particular benefits in the Naval environment and operators have been interested for some time in helicopter integration, but requirements have yet to emerge.

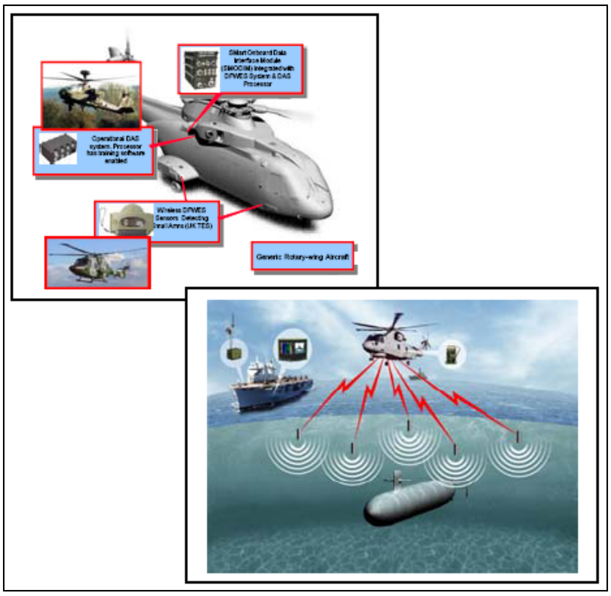


Figure 2

These systems can all communicate to ground stations that can link into networks using DIS and HLA.

6. CONCLUSION

The technology exists and has been proven to work, creating the opportunity to link in Live, Virtual and Constructive training assets to meet the needs of all types and levels of collective training as required by individual nations or at the multi-national level. However there are some barriers that must be overcome:

- Military stove pipes: Inevitably individual services and units will have to give up some live training budget to fund better synthetic and integratd training which has proved difficult if not impossible. Unless forced (increasingly more likely) they make take some convincing, but if we can demonstrate real training value rather than inert role playing then this may help.
- Procurement stove pipes: Individual programmes are still cutting costs by droving out interoperability with bespoke solutions, especially in visual databases with several versions of the same area
- Industry stove pipes: Preservation of IPR has lead to incompatible systems and databases with proprietary interfaces. Real common standards must be the way ahead.

Clearly we all need to work together if we are to get the full benefit of LVC.

REFERENCES

Lenoir, T, 2003 . PROGRAMMING THEATRES OF WAR: GAMEMAKERS AS SOLDIERS, Stanford University. Available from: http://jhfc.duke.edu/jenkins/publications/Lenoir_TheatresOfWar.pdf, Accessed 1/6/2011

Saltmarsh, J, MacKenzie, S, 2008 . The Future of Collective Training: Mission Training through Distributed Simulation, RUSI. Available from: http://www.rusi.org/downloads/assets/29saltmarsh.pdf, Accessed 1/6/2011

Mace, Dr C, 2011. The Strategic Roadmap for Defence, May 2011 (Cologne, Germany)

Snow, C, 2011. Future Threats, Challenges and Opportunities, May 2011 (Cologne, Germany)

Vierick, K, 2011. Key Note Address: Training Transformation in NATO, May 2011 (Cologne, Germany)

Warrick, G, 2004 . Apaches display virtual versatility, Stanford University. Available from: Flightglobal, http://www.flightglobal.com/articles/2004/12/21/191672/apaches-display-virtual-versatility.html, Accessed 1/6/2011

AUTHORS BIOGRAPHY

Ian McVeigh : Ian joined AgustaWestland in 2000 after working in production engineering and programme management for 22 years. At AgustaWestland spent 2 year as resident programme manager for Apache and returned to the UK to develop the embedded Collective Training system for rotary. Currently Ian is the Business Training Manager for the UK.

Jon Sackett : Jon joined AgustaWestland in 1981 after completing a Mechanical Engineering Degree at Imperial College. He has worked on a variety of engineering simulation projects including R&D simulators for Helicopter and Fighting Vehicle Man

Machine Interface (MMI) development. This activity extended to training applications including collective training. Currently Jon is responsible for establishing civil regional training centres for AgustaWestland.

DISTRIBUTED SIMULATION OF ELECTRONIC WARFARE COMMAND AND CONTROL SCENARIOS

Sergio Tortora[a], Davide Cannone[b], Massimo Giovacchini[a], Giuseppe F. Italiano[b], Maurizio Naldi[b], Andrea Sindico[a]

[a]Elettronica S.p.A.
[b]Dipartimento di Informatica, Sistemi e Produzione, University of Rome "Tor Vergata", Italy

sergio.tortora@elt.it, cannone@disp.uniroma2.it, massimo.giovacchini@elt.it, italiano@disp.uniroma2.it, naldi@disp.uniroma2.it, andrea.sindico@elt.it

ABSTRACT

In this paper we describe the main features of distributed simulation environment we have designed and developed for testing a Command and Control system for distributed Electronic Warfare (EW) Network Centric Operations,

Keywords: Network Centric Warfare, Network Centric Operations, Electronic Warfare, Distributed Simulation

1. INTRODUCTION

The exploitation of networked distributed sensors and actuators is becoming a key factor in military operations. This trend, called network centric warfare or network centric operations (Alberts, Garstka and Stein 2000), allows to have several views of the surrounding environment, which eventually leads to an improved and more precise situation assessment. That's the key to quick and effective decisions.

Different kind of sensors will indeed perceive different characteristics of the same entity. For instance, passive radar detectors, such as Electronic Support Measures (ESM) or Radar Warning Receivers (RWR), are able to detect, measure, and identify radar emissions. Passive communication detectors, such as Communications Electronic Support Measures (CESM), perform the same task on radio signals. Traditional primary radars are capable of detecting the presence of moving targets relying on their radar cross sections. An integrated management of those sensors allows to exploit sensing diversity, spanning several domains, such as space, frequency, and waveform. The inclusion of passive sensors makes the detection task even more effective because of their zero probability of being intercepted by possible opponents.

The exploitation of this wide variety of distributed sensing information is however possible if a proper C4ISR infrastructure is in place. C4ISR is a military acronym standing for *Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance*. The complexity of the functions incorporated in a C4ISR infrastructure makes its definition a challenging task and require a complex chain of definition, design and optimization, and test phases. The very nature of the operations associated to C4ISR makes it impossible to rely on an experimental validation of the strategies envisaged. The test of C4ISR strategies has therefore to be accomplished through simulation.

We have developed a simulator employing the principles of distributed simulation to realize realistic battlefield scenarios and test Command & Control strategies in a C4ISR framework. In this paper we describe the main features of our simulator and report the results of a first test case devoted to analyzing the robustness of routing protocols in a mobile ad-hoc network under cyber attacks.

The paper is organized as follows. In Sections 2 and 3 we describe respectively the simulator as a whole and its module dedicated to the simulation of cyber threats. The test case concerning the robustness of routing protocols is described in Section 4.

2. THE ELT-950 SIMULATOR

In order to test the functionalities of C4ISR systems in a realistic battlefield environment, we have developed a simulator based on the principles of distributed simulation. In this section we review those principles and then describe the structure and characteristics of that simulator, named ELT-950.

Our simulation environment must allow the definition and deployment of network-centric warfare scenarios, with the possibility of automatic generation of inputs. An important requirement for such simulation environment was the capability of scaling well to scenarios having different complexities. Moreover, the possibility to add real systems into the simulation loop was strongly desired.

In order to achieve these requirements we designed a distributed simulation environment consisting of a scenario generator, which continuously updates the status of a scenario in a shared data space implemented through HLA (High Level Architecture), an IEEE standard for distributed simulation (IEEE 2010). The use of HLA allows computer simulations to interact with other computer simulations regardless of the computing platforms. The interaction takes place through the Run-Time Infrastructure (RTI), and consists in communicating data and synchronizing actions. The set of computing platforms involved in the simulation is named a federation, and the computing platform themselves (each hosting a simulation entity) are named federates. The federated entities connect to the shared data space through a dedicated bus. The resulting simulation architecture is shown in Figure 1.

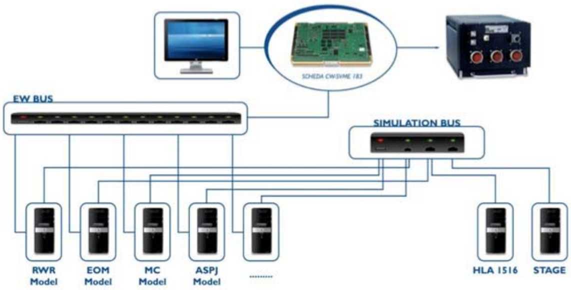


Figure 1: Distributed Simulation Architecture

Through the use of this distributed approach our simulator can boast the following features:

- Wide variety of hosting platforms
- Wide variety of sensors
- Wide variety of actuators
- Wide variety of communications systems
- Location optimization capabilities for sensors
- Integrated situation awareness and assessment
- Full OODA cycle (Observe, Orient, Decide, and Act)

The simulator can include in the battlefield scenario both fixed and moving platforms. The platform may be aircraft, ground vehicles (e.g., tanks or faster tactical vehicles), or ships. Their movement may be simulated by a number of models. In particular the model of our choice is the Reference Point Group Mobility Model (RPGM), which accounts both for movements of groups of vehicles as a whole and for the movements of individual vehicles within the group (Hong, Gerla, Pei and Chiang 1999). Each platform (or each group of platform) is simulated by a different computer in the distributed simulation framework.

Each platform may host a variety of sensors, ranging from radars to ESM systems, Laser Warning Systems, and passive sensors. The platform communicate between them through an ad-hoc network, by employing a number of communications devices (for both data and voice). At the same, we assume platform to host specific devices to accomplish offensive actions against the sensing and the communications capabilities of other platforms. In particular, we may simulate ECM systems, the use of chaff and flares, and attacks on the computing platforms to disrupt the networking capabilities of the platforms (cyber attacks). The simulator includes a meta-model specifically designed to take into account electromagnetic aspects when stating distribution strategies (Sindico, Tortora, Petrelli and Fasano 2010).

The simulator also includes the capability to choose the best location and the best frequency range for each sensor of the suite to maximize the overall surveillance capability. Having different moving passive sensors enables their distribution in space and frequency with the aim of obtaining a wide coverage of detection in both geographic and frequency dimensions. Moving Radars can therefore be distributed, oriented, and tuned, in order to minimize the interferences with ESM. Distribution is important not just to increase the surveilled area but also to allow fine target localization by means of triangulation (Benvenuti and Sindico 2010) or other techniques.

The knowledge gained through the sensors is exploited in the OODA loop, which embodies the Command \& Control capabilities of the C4ISR system. The concept of viewing the combat operations process, even at the strategic level, as a continual loop of four basic activities (Observe, Orient, Decide, and Act) was developed by military strategist and USAF Colonel John Boyd (Osinga 2006). A pictorial representation of that loop is shown in Figure 2.

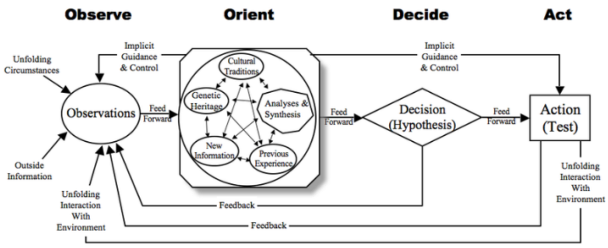


Figure 2: The OODA Loop

In our case the four phases are mapped as follows:

1. Observe → Identify possible threats, their location and evolution over time AND the effect of actions taken at previous rounds of the loop.
2. Orient → Project observations against attack signature databases and learning systems.
3. Decide → Choose the best countermeasure, spanning both the kinetic domain and the electromagnetic/cyber domain.
4. Act → Apply the countermeasure.

3. SIMULATION OF CYBER THREATS

A significant feature of the ELT-950 simulator described in Section 2 is its capability to simulate the occurrence of cyber threats in a mobile environment. The case of mobile platforms moving in a hostile environment and subject to attacks at all levels of the communications protocol stack is a very hard testbed for any simulator. We have therefore considered the simulation of cyber threats as a very first test case for our simulator. In this section we describe the set of cyber threats that the ELT-950 is able to simulate.

We wish to study the performance of a MANET in a hostile environment, i.e., containing an adversary which aims at downgrading the performance of the network, disturbing the correct functions of the network, or acquiring information about the nodes in the network.

Security issues in MANET have already been discussed in other works (Yang, Luo, Ye, Lu and Zhang 2004; Djenouri, Khelladi and Badache 2005); by studying system vulnerabilities and modeling adversary, a list of cyber attacks have been already consolidated. All of the approaches are related to a discipline called *threat modeling*: it is the process of enumerating and risk-rating malicious agents, their attacks, and those attacks' possible impacts on a system's assets. The benefits of threat modeling are: aiding in prioritizing types of attack to address, helping more effectively in mitigating risk, augmenting assessments with new potential attack vectors, identifying business-logic flaws and other critical vulnerabilities that expose core business assets.

In this section we describe our threat model, that focuses on a generalized view of known cyber attacks to untie the simulation from unnecessary details.

Threats are strictly related to the model of the adversary, so, the first step in threat modeling is to characterize the adversary. In this context our assumptions about the adversary are:

- an adversary can take control of a friendly node, replacing it with a malicious node; this event, in general, happens with a probability different from 100%, and it depends on many factors (e.g., type of intrusion attack, human behavior) but we considered the worst case, where this event happens with a probability of 100%; in this way there is no need to distinguish between internal and external adversaries, and we can assume that an adversary can always get access to the network;
- an adversary can take control of more than one node;
- malicious nodes are mobile;
- malicious nodes can cooperate to attack the system and can communicate on a reserved wireless channel. This means that friendly nodes cannot obtain information about an attack (before the attack itself) and use it to organize a defense;
- friendly nodes cannot detect malicious nodes and organize a defense;
- an adversary cannot deploy malicious nodes on the network before the deployment of friendly nodes. This means that some specific attacks, like Rushing attacks (an attack that acts like a Denial of Service against on-demand ad hoc network routing protocols; it is based on the possibility to forward routing packets before friendly nodes do, to create routes that include malicious nodes; see Hu, Perrig and Johnson 2003), are not achievable;
- malicious nodes are at least as computationally strong as the friendly ones. This means that the former can access as many resources as the latter.

When approaching a simulation, it is necessary to make some assumptions and decide what and how to simulate: indeed, in a simulation, some differences in attacks may not be well perceptible (e.g., Eavesdropping, an attack where the opponent listen passively the wireless channel to sniff packets, versus Traffic Analysis, an attack, in ciphered network, where the opponent try to infer information on the network and its participants by watching characteristics of the traffic (Raymond 2001), and some attacks may not be easily reproducible without adding unnecessary complexity (e.g., Sleep Torture Deprivation, an attack, executed in a network of battery-powered nodes, where the adversary try to exhaust the battery of the node target (Stajano and Anderson 2000) and the necessity to have a power consumption model).

We decided to abstract most from specific attacks, to create a threat model able to represent well the majority of known cyber attacks against a MANET and not to focus on details that cannot be reproduced in the simulation; for example, our threat model does not reckon with security issues in the application layer.

For the purpose of this paper, we define as *fake* a packet sent by a malicious node and as *intercepted* a packet received by a malicious node.

We grouped cyber attacks in four main categories:

- *Denial of Service*: The adversary causes an overloading, an interruption or a disturb in the network such that it begins to misbehave; often this misbehavior represents a downgrade of the performances. The attack can be reproduced by sending to the system more requests that it can handle or by using flaws in protocols used on the system. In the simulator this attack is executed by replacing a friendly node with a malicious one, and forcing the malicious node to send a constant flow of messages towards a target friendly node, at a rate higher than normal communications. All the outgoing packets of the malicious node are tagged as *fake* and all the incoming packets are tagged as *intercepted*. This attack can reproduce a *Distributed Denial of Service* when there is a single target node and multiple malicious nodes.
- *Fabrication*: The attacker fabricates spurious messages, whose nature depends on the attacker's access level to the system, and inserts them in the network; for example, these can be replication of packets that have already traveled on the network, fake signaling messages or packets to promote fake links or fake nodes. In the simulator this attack is

executed by inserting a malicious node near a friendly one and tagging as *fake* all the packets sent by this node.

- *Interception*: The attacker does not interfere with the network operations and limits itself to eavesdrop packets. The information gained depend on the level of encryption of the system layers. Whenever the adversary cannot obtain any information due to encryption, he can use traffic analysis techniques. By analyzing the messages flows, the attacker can infer information not directly accessible: for example, he can understand that two nodes have different functions by noticing the frequencies of output messages. In the simulator this attack is executed by tagging a friendly node and all the incoming packets as *intercepted*.
- *Impersonation*: The attacker mimics a target node, intercepting its messages and sending packets signed by it. This attack has both the scope to gain sensible information and have an active role in the network; in this way, the malicious node can participate to the distributed operations of the network (e.g., routing) and have the possibility to give fake information. This type of attack is often known as Man in the Middle. In the simulator this attack is executed by replacing a friendly node with a malicious one, and tagging as *fake* all the packets sent and tagging as *intercepted* all the incoming packets.

At last, there is the need to evaluate the impact of a cyber attack towards the analyzed system; other than using metrics related to the performances of the network and of the single nodes, we decided to use as metrics the percentage of the overall *fake* packets received by any friendly node and the percentage of the overall packets *intercepted* by the adversary; these measures are taken at the routing layer.

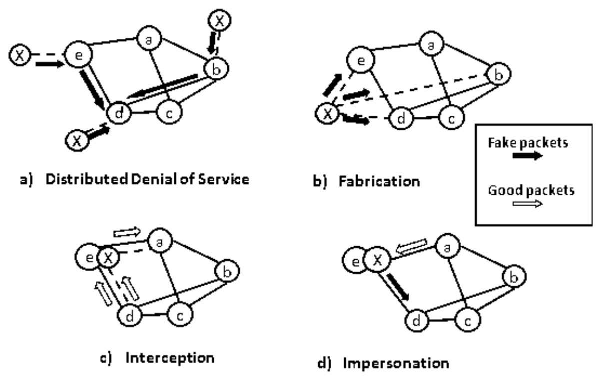


Figure 3: Threat Models

Figure 3 is a graphical representation of the threat model that we have chosen: nodes marked with an X represent malicious nodes, inserted in the network or by replacing a friendly ones; dotted lines represent communication channels with a malicious node.

By modifying properties of these four attacks and combining them, other more specific attacks can be represented in the simulator: for example, in the *Impersonation* attack, when the malicious node does not fabricate any fake packet, this can be a representation of a sinkhole attack (in a sinkhole attack, the adversary's goal is to lure nearly all the traffic from a particular area through a compromised node, creating a metaphorical sinkhole with the adversary at the center; see Karlof and Wagner 2003).

4. A TEST CASE: ROBUSTNESS ANALYSIS OF MOBILE ROUTING PROTOCOLS

We have run a first test of the capabilities of our simulator. We have compared the performances of a set of MANET routing protocols in the face of cyber attacks (Cannone, Naldi, Italiano and Brancaleoni 2011). In this section we report a brief overview of the simulation results.

In order to test our simulator we have defined a realistic simulation scenario, considering mobile ad-hoc network operating in a nearly flat geographical area (i.e., there are no relevant obstacles either for movements or signal propagation) extending over 100 square kilometers. Inside the region there are 15 nodes, either fixed (representing base stations) or mobile (representing slow ground vehicles). Every node communicates through bidirectional wireless channels and mounts an omnidirectional antenna. The receiving sensitivity threshold has been set so that any two mobile stations are connected if their distance is lower than 2 kilometers. Any node can generate traffic network towards any other node: the network traffic matrix has random entries, with every flow having a probability of 50% to exist. Every packet source spawns traffic according to an On/Off process with exponential distributions for both On and Off times, and an average rate of 1Mbit/s. The average packet size is 1000 Byte.

As to the routing protocols, in the test case we have considered the following selection of routing protocols (see Akkaya and Younis 2005) for a survey of routing protocols in MANETs):

- Destination-Sequenced Distance-Vector (DSDV);
- Ad-Hoc On-Demand Distance Vector (AODV);
- Dynamic Source Routing (DSR);
- Zone Routing Protocol (ZRP);
- Fisheye State Routing (FSR).

Among them, DSDV and FSR are proactive protocols, where the nodes maintain up-to-date routing information. Instead, AODV and DSR are reactive protocols, setting up routes on demand. Finally, the ZRP is a hybrid protocol, employing the proactive approach for the nodes inside a local area and the reactive one to reach nodes outside of it.

As to the threat models, we report here the results obtained with a scenario representing an aggressive attack towards the network, composed mostly of malicious nodes performing Denial of Service and Fabrication attacks.

We have considered the following performance parameters:

- Connectivity
- Goodput
- Packet delivery ratio
- Delay
- Percentage of intercepted packets
- Percentage of fake packets

The goodput (expressed in Kbps) is the amount of Kbits of useful data received in time unit, excluding routing information and duplicates. The packet delivery ratio expresses the same quantity, but expressed in packets rather bits. The delay (expressed in milliseconds) is defined as the time between the sending of a message until its complete reception by its recipient. We wish it to be as low as possible, though it will grow with the traffic. The percentage of intercepted packets is the ratio of all packets, tagged as *intercepted*, received by any malicious node, and the number of packets not tagged as *fake*. This metric represents the probability that the attacker gets routing information, breaking the anonymity of the network. A system designer aims at minimizing this metric, especially in scenarios where eavesdropped information can open the door to a more lethal attack; in an encrypted communication layer the system designer must pay attention to internal attackers, who can access network services and intercept packets more easily. The percentage of fake packets is the ratio of all packets, tagged as *fake* and received by any friendly node, and the number of packets received by any friendly node; we exclude packets received by any malicious node. This metric represent the probability that a friendly node receives spoofed or corrupted packets, e.g., with bogus routes or making some weak link more attractive.

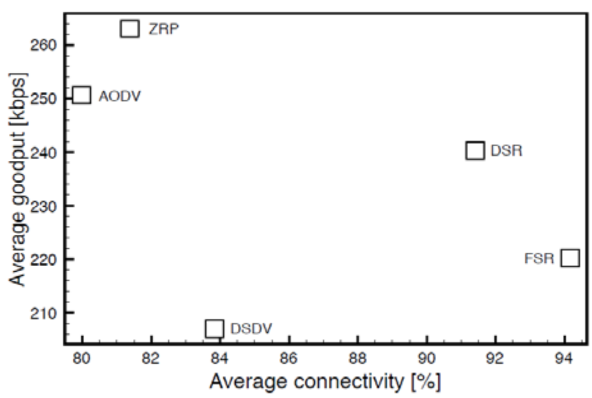


Figure 4: Comparison of Protocols in Sending Data Through

In the first graph, in Figure 4, we show the performance parameters concerning the capability to getting data through. Namely we plot the goodput versus the connectivity. We expect a robust routing protocol to guarantee connectivity and to have as high a goodput as possible. Hence, the best protocols are those positioned on the upper right corner of the graph. Though there is not a clear winner, we see that the reactive protocols perform generally better than the proactive ones.

A second set of results describes again the capability of protocols to get data through (represented by the packet delivery ratio), but compared with the delay experienced by packets. We wish to have a low delay as well as a high delivery ratio. In Figure 5 we show the delay vs. the delivery ratio; in that graph the best performing protocols should appear on the lower right corner. In this case the two proactive routing protocols (FSR and DSDV) are undoubtedly the best.

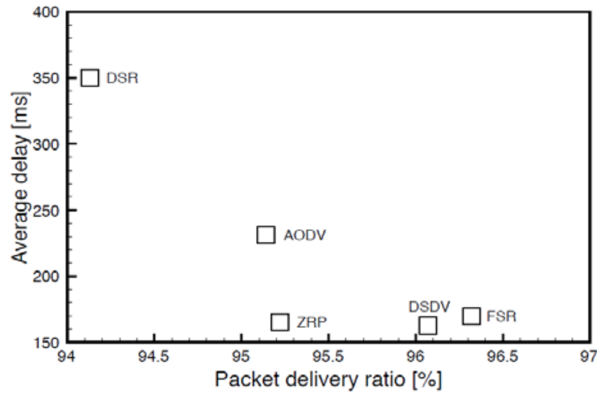


Figure 5: Delay Performances of Protocols

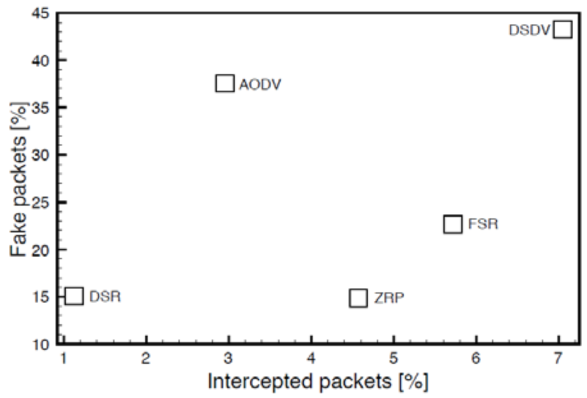


Figure 6: Routing Corruption Performances

Finally, we consider the capability of malicious node to affect the routing mechanisms. In Figure 6 we show the percentage of fake packets versus the percentage of intercepted packets. We wish to have low values for both figures of merit. The best performing protocols are those appearing on the lower left corner. Proactive protocols exhibit a problem, since they send their own routing tables at regular intervals, continuously providing the attacker with up-to-date infos on the network status. On the overall, the reactive DSR is clearly the winner here.

5. CONCLUSION AND FUTURE WORKS

We have developed a simulator to test Command & Control strategies in distributed electronic warfare command and control scenarios.

The simulator employs the principles of distributed simulation through a HLA architecture, and allows for a wide variety of platforms, sensing and communications devices, as well for attacks over the whole protocol stack.

We have run a first test case for the simulator to evaluate the robustness of routing protocols to cyber attacks. For a scenario where the majority of attacks are of the Denial of Service and Fabrication type, we have shown that reactive protocols appear as the most robust, though they are affected by larger delays.

What presented is the first result of an ongoing work. The next steps encompass a classification of communication packets with respect to the electronic warfare function to which they relate (i.e. passive search, emitter tracking, jamming, etc.). This is important to also evaluate the consequences in the EW domain that could derive from an attack to the network. Another important feature we are about to introduce is the capability of evaluating the best assignment of platforms position and functionalities (i.e. Radar search, passive search, countermeasures, etc.) in order to maximize the surveilled area within a range of frequencies chosen by the operator.

REFERENCES

Alberts, D.S., Garstka, J.J., Stein, F., 2000. *Network Centric Warfare: Developing and Leveraging Information Superiority (2nd Edition)*. CCRP Press.

Akkaya, K., Younis, M., 2005. A survey on routing protocols for wireless sensor networks. *Ad hoc networks*, 3 (3), 325-349.

Benvenuti, D., Sindico, A., 2010. Design and simulation of a distributed sensor network for passive emitter tracking and coast surveillance. *Proceedings of the International Waterside Security Conference*, pp. 1-5. November 3-5. Carrara (Italy).

Cannone, D, Naldi, M., Italiano, G.F., Brancaleoni, A., 2011. A simulator of a mobile ad-hoc network in a hostile environment. *Proceedings of the 1st International Conference on Simulation and Modeling Methodologies, Technologies, and Applications, Simultech*. July 29-31, Noordwijkerhout (Netherlands).

Djenouri, D., Khelladi, L., Badache, A.N., 2005. A survey of security issues in mobile ad hoc and sensor networks. *Communications Surveys & Tutorials, IEEE*, 7 (4), 2-28.

Hong, X., Gerla, M., Pei, G., Chiang, C.C., 1999. A group mobility model for ad hoc wireless networks. *Proceedings of the 2nd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, pp. 53-60. August 20-20. Seattle (Washington, USA).

Hu, Y.C., Perrig, A., Johnson, D.B., 2003. Rushing attacks and defense in wireless ad hoc network routing protocols. *Proceedings of the 2nd ACM workshop on Wireless security*, pp. 30-40. September 19-19, San Diego (California, USA).

IEEE, 2010. IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-- Framework and Rules. *IEEE Std 1516-2010 (Revision of IEEE Std 1516-2000).* August 18, 1-38.

Karlof, C., Wagner, D., 2003. Secure routing in wireless sensor networks: Attacks and countermeasures. *Ad hoc networks*, 1 (2-3), 293-315.

Osinga, F., 2006. *Science, strategy and war: the strategic theory of John Boyd*. USA: Routledge.

Raymond, J.F., 2001. Traffic analysis: Protocols, attacks, design issues, and open problems. *Proceedings of International workshop on Designing privacy enhancing technologies: design issues in anonymity and unobservability*, pp. 10-29. Berkeley (California, USA).

Sindico, A., Tortora, S., Petrelli, A.C., Fasano, M.V., 2010. *Proceedings of the 2nd International Workshop on Cognitive Information Processing*, pp. 23-28. June 14-16. Elba (Italy).

Stajano, F., Anderson, R., 2000. The resurrecting duckling: Security issues for ad-hoc wireless networks. *Proceedings of Security Protocols Workshop*, pp. 172-182. April 3-5. Cambridge (UK).

Yang, H., Luo, H., Ye, F., Lu, S., Zhang, L., 2004. Security in mobile ad hoc networks: challenges and solutions. *Wireless Communications, IEEE,* 11 (1), 38-47.

SIMULATED REALTIME MULTI-CHANNEL IQ DATA GENERATOR FOR AN AESA RADAR

R. Gordini[a], M. Giorgiutti[b]

[a] SELEX Galileo S.p.A., Ronchi dei Legionari, Italy
[b] SELEX Galileo S.p.A., Ronchi dei Legionari, Italy

[a] romolo.gordini@selexgalileo.com, [b] michele.giorgiutti@selexgalileo.com

ABSTRACT
Modern airborne radar systems require a complex digital processing for target detection and tracking. An increasing challenge on testing is required to validate these algorithms before flight trials. Hence the need for a development rig, capable of simulating the sensor in a realistic and dynamic environment.
The research highlights potentialities of "HW in the loop" for the radar simulation, describing a simulated real-time multi-channel IQ data generator to be used by AESA radar manufacturer to test critical modes in the radar RIG.
The system provides at runtime the IQ data for the 4 complex channels, with different PRFs scheduling, in A/A modes.
The sensitive point of the work is the matching of realtime performance requirements w.r.t. band requirements. In fact, the IQ generator should guarantee an output bit-rate of 400 Mbytes/sec, scheduling its processing, triggered by an external sync signal, at 400 Hz, with reaction times of fractions of ms.

Keywords: Radar Simulator, IQ channels, Active Electronically Scanned Array (AESA) radar, realtime performances

1. INTRODUCTION

This industrial research is aimed to provide a real-time multi-channel IQ Data Generator as produced by the SELEX Galileo radar simulation toolkit for a modern Active Electronically Scanned Array (AESA) radar prototype test RIG.

The simulated IQ Data Generator allows the Signal Processing and Radar Tracking functions of the AESA Radar to be tested against simulated real time targets in a Lab Environment.

The radar simulator provides, for each radar burst, the IQ Data (number of coherent integrated pulses × number of non-ambiguous range samples) to the Radar Signal processor for the Sum Channel Digital Data, for the Difference Azimuth Channel Digital Data, for the Difference Elevation Channel Digital Data and for the Guard Channel Digital Data for the radar modes MPRF air-to-air, HPRF air-to-air, LPRF air-to-air and Calibration modes. The System consists of 4 independent commercial servers. Each server produces the IQ data for one radar channel. The IQ Data Generator receives real-time input data via an high speed serial link at 400 Hz for navigation data, and via the 1Gbit Ethernet link at 400 Hz for radar burst data. It sends out the generated IQ data via four 10 Gbit Ethernet optical links.

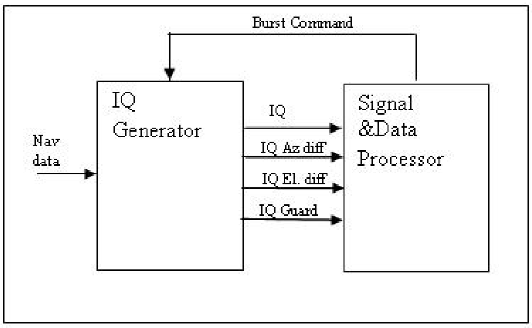


Figure 1: Architectural schema

The configuration of the test-rig is depicted in Figure 1. All elements perform in real-time. Input data is supplied to the IQ data generator from the Scenario Generator and the Navigation Simulator. Radar operational mode selection is provided by the Radar Control Panel Sim. According to the required operational mode, the Radar Data Processor (RDP) selects the radar waveform and creates a series of commands to generate the waveform and control the AESA radar. Data from these command is selected to stimulate the IQ data generator. The IQ data generator creates data representative of the target configuration at the time the command is received and sends the simulation results to the IQ output buffer. The external hardware combines the IQ data with the other parameters required from the RDP command message to mimic inputs to the processor. This complete IQ simulated data, plus header data, is sent to the processor over optical links (one link per channel).

For each channel the following radar effects are simulated:

1. Thermal noise.
2. Jet Engine Modulation (JEM).
3. Antenna Effects of electronic steering in azimuth and elevation.
4. Antenna Transmit pattern for Sum.
5. Antenna Receive pattern for Sum, Guard and difference channels.
6. Presence of multiple different targets into the scenario. A target signal accounts for the effects of:
   - Range
   - Range rate
   - Angular position within antenna pattern (including monopulse in difference channel(s))
   - Radar Cross Section (RCS)
   - Fluctuation models (Swerling 0,1 & 2)

The Radar Data Processor (RDP) generates new burst demands at rates up to 400Hz. A burst demand contains waveform and antenna command information. Each new burst demand is generated and sent whilst the current burst is in the progress of being generated. Whenever a new burst identifier is received, the real-time data generator provides the simulated IQ data for the new burst after the current burst has been concluded. The gap time between bursts has to be minimised (about 100 microseconds). Most demanding burst durations are for Air Combat MPRF and HPRF.

Considering the great amount of data to be processed and the communication rate required to the IQ Data Generator, the implemented solution consists of multiple independent workstations within a rack, basically one per channel to carry out the simulation. Each work-station shall be equipped with at least 8 cores (physical cores). The HW platform shall be identical for all the 4 radar channels. This solution not only distributes the computational load but the required band as well. Indeed a point to point communication topology is established to prevent any data collision, given the huge amount of data flowing. The chosen operating system is QNX 6.5 Neutrino (QNX).

Quantitative results of the implemented solution are presented and analyzed in details.

2. DESIGN CONSIDERATIONS

The main goal of this project is to reduce the design costs for an advanced airborne coherent radar, minimizing the need of flight trials to collect performance data. As customary in modern design processes, the development must be as standard as possible in order to make possible the reuse of the design for other radar installations. Considering these facts, the natural choice is to orient the implementation of a flexible radar test rig towards a software solution, which can be easily customized. The radar simulation toolkit is a multiplatform software that can run on the most widespread operating systems/processors, so that the choice of machine/OS is a degree of freedom for the designer (Windows, Linux, QNX, Unix and others).

A constraint of the system is that it must be hard realtime, so it has to respond within predictable time frames. This leads to the choice of QNX 6.5 Neutrino, a consolidated OS for embedded applications. The architecture chosen is the 64 bit Intel platform, given its high performance/cost ratio.

The other main constraint is the huge amount of data that must flow from the IQ generator to the signal processor. The theoretical figure is 400 Mbytes per second, in the worst case. Rather than centralizing all the simulation into a high performance machine, it is better to distribute the computational load, so that the required bandwidth for each single machine is a fraction of the centralized solution, and there is guaranteed parallelism between the machines. Even a mainframe with four powerful network adapters may not guarantee the necessary bandwidth all the times, depending on the implementation of that particular machine. This does not meet the objective of flexibility of such a system, since it binds the performance of the system to a peculiar machine, rather than to a machine class.

It can be argued that this implementation does not strictly guarantee realtime. In a sense it is true, because the rig relies on network technologies, but given the architecture, the probability of an overrun it isn’t likely to occur. And if on overrun occurs it is detected, an exception is thrown. An overrun is not hazardous, since this is not a safety critical application, so just the reporting of the overrun it is enough. The exception will invalidate the exercise results, so that they won’t be used in the radar processor development.

3. DATA SYNCHRONIZATION

Since a distributed computation scheme has been chosen for the simulation, a synchronization mechanism is necessary to have a consistent simulation. In particular NAV data and target data must be the same for each of the workstations performing the processing. To achieve input data alignment, an exercise based implementation of the targets has been devised. An exercise is a trajectory in time and space which is assigned to each of the simulated targets. Trajectories are defined as a sequence of segments, and each segment must have the following parameters specified:
Segment start and stop times (t_0, t_1)
Segment velocity as

$$\begin{cases} V_x(t) = V_{x,0} + V_{x,C}\,\cos(\omega_a(t-t_0))\,\cos(\omega_b(t-t_0)) \\ V_y(t) = V_{y,0} + V_{y,C}\,\sin(\omega_a(t-t_0))\,\cos(\omega_b(t-t_0)) \\ V_z(t) = V_{z,0} + V_{z,C}\,\sin(\omega_b(t-t_0)) \end{cases}$$

Segment linear acceleration (constant vector).
All these vectors are referred to a North oriented fixed Cartesian reference system (scene centre origin). Angular velocities Ω_a, Ω_b represent respectively the heading rate and the pitch rate of target. Before starting the realtime, hardware in the loop simulation, exercises are sent to the Synthetic Environment. The SE

computes target trajectories as define above, and sends the time tagged information to all processing machines. In this way all processing machines can implement the same identical interpolation of the data between two consecutives simulation frames.

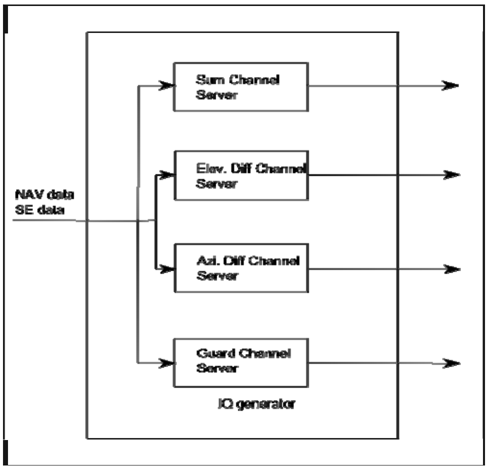


Figure 2: IQ generator block diagram

If any problem occurs, or the processing requires more time than prescribed, the output buffer is marked as not valid before is sent to the signal processor.

4. IQ GENERATOR PROCESSING OVERVIEW

Burst operation is often implemented in coherent radar modes. During burst operation the radar transmits packets of pulses, called bursts, and each pulse is separated from the previous one by dozens of µs. The radar return from each pulse of the burst is elaborated, and amplitude and phase of the signal are extrapolated for each range bin, representing the radar response in the range-time domain. This information is more suitable for digital processing in Cartesian representation, so amplitude and phase are converted into in phase and in quadrature components. Of course this gives the name to the IQ generator. Data to be processed for each burst can be stored in a matrix of complex numbers: the rows are ordered by pulse, hence time, the columns by range-bin, hence distance. Indeed the time range matrix represents the output of the IQ generator.

Table 1: IQ matrices

$I_{11} + j\, Q_{11}$	$I_{12} + j\, Q_{12}$	…	…	$I_{1N} + j\, Q_{1N}$
…	…	…	…	…
$I_{M1} + j\, Q_{M1}$	…	…	…	$I_{MN} + j\, Q_{MN}$

For non coherent radar operation the usual radar equation is employed (Picardi), hence only the received power is known, there is no way of estimating the relative phase of consecutive pulses. The relative phase of the signal can be computed considering the estimated wave vector $\vec{k}$ as follows: $\alpha = \vec{k} \cdot \vec{r}$, where α is the phase and $\vec{r}$ is the ray vector that connects the antenna bore-sight to the simulated target. From this it is clear that it is mandatory to perform accurate target slant range measures. Indeed the meter to be used is the wavelength at working the frequency, so the slant range resolution Δ must be less than $\lambda / 30$.

This implies that the simulation of targets, along with the navigation data, must be as accurate and as synchronized as possible. Standard Synthetic Environments do not meet the latter constraint so that a custom SE based exercise has been designed, as previously discussed. Between two consecutive frames an interpolation is applied, to limit side effects due to space quantization.

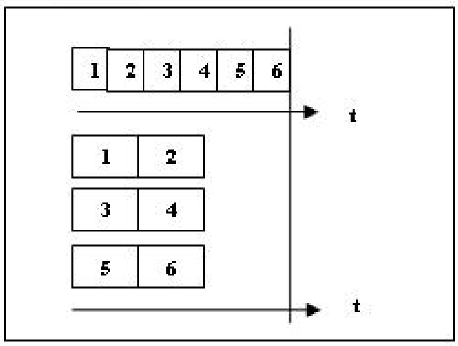
Additive white noise is inserted to model the receiver noise figure NF, the mean power of the random noise been computed as: $Pn = K_b T B NF$, where K_b is Boltzmann constant, T is the absolute temperature and B is the receiver's bandwidth.

5. PARALLELISM SCHEME

Given the stringent time constraint, it is essential to perform the processing as fast as possible. In order to achieve this, a two level parallelism is employed, at channel level and at pulse level. Data to be filled in each channel is of course not dependant from data of other channels. Indeed there is one IQ matrix per channel. Each machine, there is a server per channel as shown in Figure 2, implements a parallel computing scheme based on pulses. In fact, given the limited computational power, the time simulation of a single pulse is not feasible, because the pulse lasts few µs. For this reason the burst β to be simulated is partitioned into sets of pulses:

$$\beta = \{\{\rho_1, \rho_2, ..., \rho_{N1}\}, \{\rho_{N1+1}, ..., \rho_{N2}\}, ..., \{\rho_{Nk-1,...,} \rho_{Nk}\}\}.$$

The cardinality of β is equal to the number of processors available, and each element of $\{N_1, N_2, ..., N_k\}$ represents the number of pulses in the burst. The simulation for each of the pulses subsets starts simultaneously on different CPUs, and this the key factor that allows the burst simulation on time.



Example 1: Pulse Parallelization

It is evident from Example 1 that even though the single pulse time is grater than the actual radar pulse time, exploiting more CPUs makes possible to simulate the burst in realtime.

6. ANTENNA SIMULATION

The implemented antenna is a four lobe electronic scan monopulse antenna (Sletten). All monopulse channels are considered: sum channel, elevation channel and azimuth difference channel. AESA radars can form multiple beams to scan the volume without mechanical steering. The gain patterns of the antenna have been sampled from the actual radar system, but a parametric gain pattern customization is possible as well. Given the azimuth and elevation pointing angles the antenna gain patters are extracted. In the former case, sampled gain patterns, it is just matter to retrieve table indexes and perform an interpolation if required. In the latter case, parametric gain pattern, there is the need to call a designed mathematical function that implements the gain pattern.

Of course sampled gain patters yield a more reliable antenna simulation, but they require a large amount of system memory to store runtime lookup tables. Parametric gain patterns are useful solely when a standard "data-package" of the radar is available to the simulation, and hence only global figures are known.

7. PERFORMANCE ANALYSIS (1 CHANNEL MOCK-UP)

7.1. SYSTEM OVERVIEW

A prototype of the test rig has been developed to investigate the feasibility of the project. The test setup is depicted in Figure 3. Workstation 1 implements the sum channel, while an emulator of the radar signal (SIP) and data processors (RDP) mimics the functionality of the devices under test. Having the control of the emulated radar processors enables to effectively measure the overall time performance of the system, including data transfer time, which is not negligible.

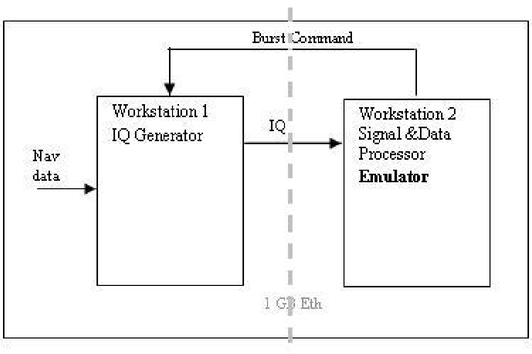


Figure 3: Prototype schema

The communication is carried out through an 1 GB Ethernet link (both directions of Figure 3). Final radar rig has got, as previously detailed, 10 Gbit optical links, i.e. the measured output throughput of the mock-up has got more band limitations than the target system.

At application level, a basic communication protocol has been devised:

- The IQ generator starts and waits for a burst command from the emulator.
- The emulator starts and sends a burst command to the IQ generator. The emulator waits for an acknowledge.
- The IQ generator receives the command and begins the processing, while sending the acknowledge
- If the emulator doesn't receive the acknowledge before the time out (half the simulation period), it throws an exception, otherwise it puts itself in an idle state, waiting for the IQ generator response.
- While the processing is over, the IQ generator sends the time-range matrix to the emulator that collects the data. If no exceptions occurred, a new cycle can start.

The most demanding modes of operation are Medium PRF and High PRF, the latter for the amount of data to be processed, the former for the data rates required.

Table 2: Performance estimation

1 CHANNEL	HPRF	MPRF
Data	175 kB	50 kB
Burst Time	10 ms	2.5 ms
Predicted proc. time	5 ms	2 ms
Comm. time	5 ms	0.5 ms
Data rate	35 MB/s	100 MB/s

To obtain a quantitative benchmark of the system, it is necessary to measure the global time frame, which includes data IO for the stimuli, processing time and data IO for the result. This global time must be less than the prescribed frame, with a reasonable safety margin. Time measures of the pure processing time are also recommended, since this gives a metric for the computing efficiency and the input output time performance. This information can be used to assess the level of parallelism reached for each machine, changing the number of concurrent threads. Increasing the number of parallel threads reduces the overall computing time.

7.2. HARDWARE

The mock-up of the test rig is composed of two servers, one running the IQ sum channel simulation software, the other the radar processors Emulation software. The characteristics of these machines are briefed in Table 3.

Table 3: Server characteristics

Processor	X 2	Intel Xeon processor E55303, 2.40 GHz, 8 MB cache, 1066 MHz memory, Quad-Core
RAM	X 2	8GByte of DDR3 RAM
Network adapter	X 2	NetXtreme BCM5764M Gigabit Ethernet PCIe1 GBit
Operating system	X 1 X 1	QNX 6.5 Neutrino SUSE Linux Enterprise Server 11, RT

Standard serial ATA hard drives are installed into the machines for a total storage capacity of 250GB for each channel.

7.3. HPRF

HPRF modes provide long range air-to-air detection of closing targets w.r.t. the radar. Target signals are well separated from surface clutter returns so that target detection is noise-limited. In this mode the radar has got a pulse repetition frequency so high that the non ambiguous range is very short. This means that the number of range bins is limited in number, while there are many pulses in every burst. It follows that the time-range matrix has got more columns than rows. Using figures of Table 2, it is clear that this mode has the greatest number of elements of the matrix, and consequently the longest processing time, but it has also a large time frame to achieve its task.

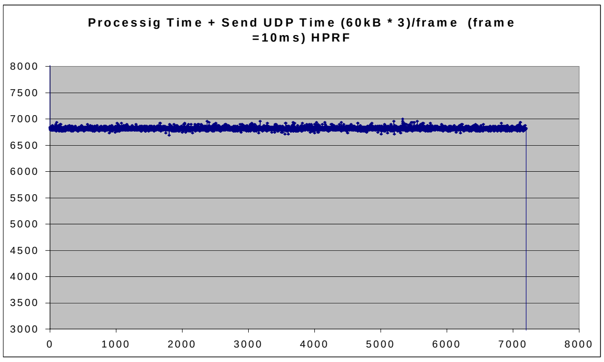


Figure 4: Global time frame measurements vs. burst number, HPRF

Figure 4 shows that, on the average, the global time frame is less than 7 ms, and hence yields a 30% margin. An interesting remark can be made comparing the above figure with the results figure 5. Indeed it shows the processing time. It takes 4ms, hence the input output operations occupy 3ms, which is equal to the 75% of the active modeling. The IO operation are expected to be faster on a 10 Gbit optical link, so there is further room for improvement of the bearable time frame.

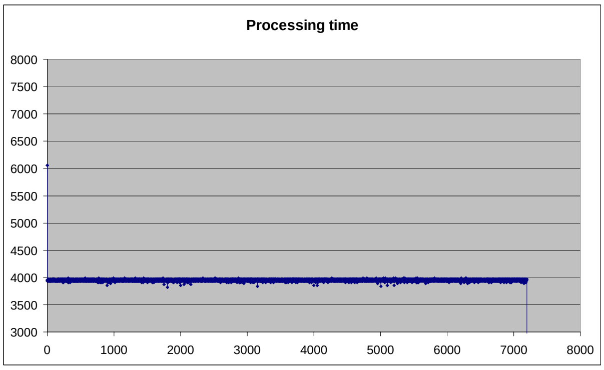


Figure 5: Processing measurements versus burst number, HPRF

7.4. MPRF COMBAT

This is an air-to-air detection mode designed to rapidly acquire targets at close range before transition from TWS to STT operation. The measurements for MPRF COMBAT shows that the global I/O time is about 1 ms, which is more than one third of the prescribed time frame of 2.5 ms. The processing itself for the IQ data generation is about 1.6 ms.

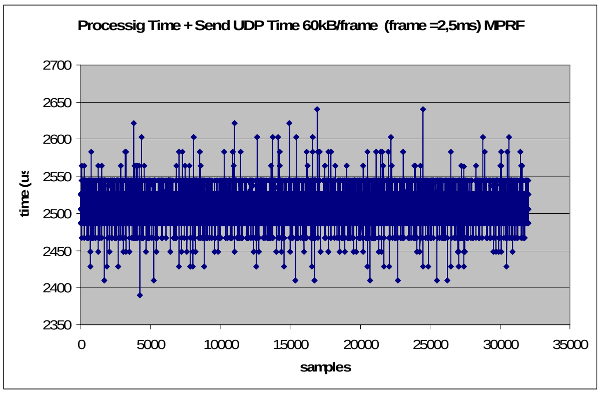


Figure 6: Global time frame measurements vs. burst number, MPRF

From the empirical evidence, there is an additional delay (over the 2.5 ms). It must be considered though that these tests were performed with a 1GBit Ethernet (125 Mbyte maximum theoretical rate), and that the time required for the IO amounts to 40% of the whole time frame. With a 10 GB (1.25 Gbyte maximum theoretical rate) this aspect will improve more than significantly.

For example if the band doubles, the weight of the transmission time will be 20 % (500 us) possibly yielding a cycle time of 2 ms.

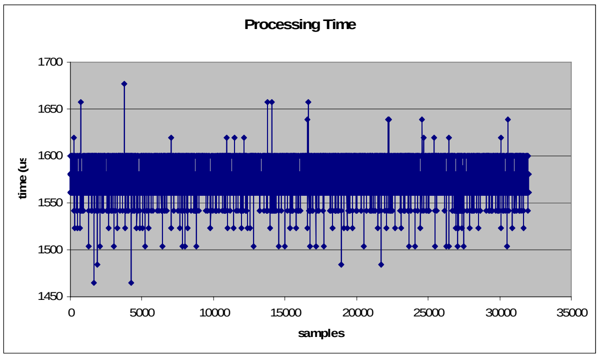


Figure 7: Processing measurements versus burst number, MPRF

8. CONCLUSIONS

Performed tests lead to the conclusion that HPRF bursts (the worst case for data I/O rate) can be successfully handled by the realtime IQ Data Generator with the identified solution. Having theoretical dwell periods of 10 ms, 4 ms shall be used for IQ data processing, 2,9 shall be used for data I/O. The idle time is 3.1 ms (31 %). The usage of 10 Gbit Ethernet boards will improve more and more these numbers.

The AESA RDP Emulator did not registered a single overrun, proving that the comprehensive I/O mechanism (data exchange protocol and synchronization) is finely working, as per hypothesis.

Tests run so far lead to the conclusion that MPRF COMBAT bursts (the worst case for computation rate) can be successfully computed by the realtime IQ Data Generator with the identified solution. It remains an uncertainty on the time needed for data I/O. Having theoretical dwell periods of 2.5 ms, 1.6 ms shall be used for IQ data processing, 1 shall be used for data I/O. Spurious overruns of 0.5 - 1.0 ms have been measured both on the IQ Data Generator and on the AESA RDP Emulator. It must be considered though that these tests were performed with a 1 Gbit Ethernet, and that the time required for the IO amounts to 40% of the whole time frame. With a 10 Gbit this aspect will improve more than significantly.

Using a machine with more that 8 cores and 10 Gbit boards, it is quite likely to be able to handle also this case as per requirements.

The usage of the simulated IQ Data Generator allows the Signal Processing and Radar Tracking functions of the real AESA Radar to be tested against simulated real time targets in a Lab Environment in a consistent and repeatable manner. Very often such evaluation with flight trials are very complicated, expensive and difficult to be repeated.

REFERENCES

Picardi, G., 1988. *Elaborazione del segnale radar, metodologie ed applicazioni,* 31-41

Asoke K. Bhattacharyya D.L. Sengupta: Radar Cross Section Analysis & Control

Carlyle J. Sletten, 2010: Reflector and Lens Antennas Analysis and design using personal computers

QNX Software System Ltd, 2004 : QNX Neutrino RTOS V6.3 system architecture

AUTHORS BIOGRAPHY

Romolo Gordini graduated in Business Engineering at the University of Udine in 1997. He joined SELEX Galileo in 2001 where he has been working in the Radar Simulation, that he led as a Project Leader since 2004. In 2005 he has been assigned as Head of Sensors Simulation Department in Ronchi dei Legionari, with design and integration responsibility of sensors simulation for all the simulation programmes of the Line of Business, such as the CAPTOR radar on the Eurofighter Typhoon, the Tornado Nose radar and the APS-784 radar on the AW EH101 helicopter.

He is one of the 2 authors of the realtime radar simulation toolkit MARS (Multimode Airborne Radar Simulator).

Michele Giorgiutti graduated in Electronic Engineering at the University of Udine in 2009, with the thesis “Discrete geometrical formulations for the solution of 2D and 3D electromagnetic problems”. He also worked as collaborator for the DIEGM department of the University of Udine. in the same year. During that period he developed C/C++ and Fortran software for the numerical solution of three dimensional eddy currents problems employing a geometrical method called “Cell method”. Since June 2009 he has been working for SELEX Galileo, in Ronchi dei Legionari, Italy, where he is a System Engineer for the Sensor simulation & Database Department. He contributed to the analysis and design of realtime interfaces between avionic equipments and sensor simulators, running on commercial servers.

DISTRIBUTED SIMULATION SCIENCE

Margaret L. Loper
Georgia Tech Research Institute
margaret.loper@gtri.gatech.edu

ABSTRACT
Computing technology has advanced dramatically over the last twenty years, enabling new applications for networked simulation. Along with these applications are architectures and standards that support the interoperability of heterogeneous simulations. This paper begins by looking at the historical roots of distributed simulation, from an architecture and policy perspective. The paper then examines distributed simulation as a technology, which is based on the science of distributed systems. The paper identifies two key characteristics of distributed systems and then describes the modern distributed simulation architectures based on these characteristics.

Keywords: distributed simulation, interoperability standards, distributed systems

1. INTRODUCTION

Early computer simulations were limited from several perspectives. Most simulators required identical computer hardware and software in order to operate and create the environment. In this sense they were closed systems, and not expandable. Also computer networks were synchronous; the fixed nature of synchronous systems using shared memory and time slicing of data arrivals made adding simulators to these networks impractical. Technically it was a challenge to separate simulator capabilities so that functions could be performed autonomously.

This changed with the evolution of computer and communications technologies. The late 1960's and early 1970's brought the development of several key technologies that enabled distributed simulation. The UNIX operating system brought the ability to handle asynchronous events, such as non-blocking input/output and inter-process communication. That was followed by expanded computer-to-computer communications with the creation of the Advanced Research Projects Agency Network (ARPANET). The ARPANET enabled a computer to communicate with more than one machine by transmitting packets of data (datagrams) on a network link. The invention of Ethernet technology enabled the connectivity of a variety of computer systems within a local geographical area. Thus simulators no longer needed to be located in the same room, but could be distributed. Then, in 1974, the Transmission Control Protocol and the Internet Protocol (TCP/IP) emerged as a means to enabled intercomputer communications. This development supported heterogeneous computer communications, which would change simulators from being bounded by their computational resources to being bounded by their ability to send and receive information.

The remainder of the paper examines distributed simulation as a technology. It begins by looking at the historical roots of distributed simulation, from an architecture and policy perspective. It then reviews the science of distributed systems and identifies two key characteristics that apply to distributed simulation. Using these characteristics, the paper describes the modern distributed simulation architectures in use today.

2. DISTRIBUTED SIMULATION BEGINS

As computing technology advanced, applications for networked simulation emerged. These applications included inter-crew training, live tests and training, and analysis. For inter-crew training, improving tactical proficiency and training large groups was a growing need. In live tests and training, reducing the number of events was desired due to safety, environmental and cost concerns. In the discipline of analysis, there was a need for individual subsystem evaluation and fine-tuning of tactics and strategies. The emerging application areas and the advances in computing and communication technology led to the first generation of distributed simulation architectures.

Initiated in 1983 by the Defense Advanced Research Projects Agency (DARPA), the SIMulator NETworking (SIMNET) project was the first attempt to exploit the developments in communications technology for simulation (Miller and Thorpe 1995). SIMNET emphasized tactical team performance in a battlefield context, including armor, mechanized infantry, helicopters, artillery, communications, and logistics. Combatants could visually see each other "out the window" and communicate with each other over radio channels. This distributed battlefield was based on selective fidelity (only provide features for inter-crew skills training), asynchronous message passing, commercial computer networks, and replicated state information. SIMNET was based on six design principles:

- *Object/Event Architecture*; the world is modeled as a collection of objects which interact using events.

- *Common Environment*; the world shares a common understanding of terrain and other cultural features.
- *Autonomous Simulation Nodes*; simulations send events to other simulations and receivers determine if that information is relevant.
- *Transmission of Ground Truth Information*; each simulation is responsible for local perception and modeling the effects of events on its objects.
- *Transmission of State Change Information*; simulations transmit only changes in the behavior of the object(s) they represent.
- *Dead Reckoning Algorithms*; simulations extrapolate the current position of moving objects based on its last reported position.

The first platoon-level system, as shown in Figure 1, was installed in April 1986 and over 250 networked simulators at eleven sites were transitioned to the U.S. Army in 1990. Two mobile platoon sets (in semi-trailers) were delivered to the Army National Guard in 1991.

Figure 1: SIMNET Simulators at Fort Knox, KY

Among the many contributions of the SIMNET program was the invention of semi-automated forces (SAF). The purpose of a SAF simulation was to mimic the behavior of different objects on the battlefield, whether vehicles or soldiers. Realizing it was impractical to have large numbers of operators to control both friendly and opposing forces, COL James Shiflett created the idea of SAF as a way to put opposing forces on the battlefield. His inspiration for SAF was "Night of the Living Dead", a movie in which teenagers are attacked by zombies. COL Shiflett wanted a simulation that could produce a large number of "dumb" targets (i.e., zombies) to roam the battlefield and provide targets for SIMNET operators (Loper 2008). Eventually SAF simulations turned into something much more intelligent; they could plan routes, avoid obstacles, stay in proper formations, and detect and engage targets.

Soon after the SIMNET project, DARPA recognized the need to connect aggregate-level combat simulations. The Aggregate Level Simulation Protocol (ALSP) extended the benefits of distributed simulation to the force-level training community so that different aggregate-level simulations could cooperate to provide theater-level experiences for battle-staff training. In contrast to the SIMNET simulators, these simulations were event-stepped and maintaining causality was a primary concern. The ALSP program recognized that various time management schemes and more complex simulated object attribute management requirements were needed. The requirements for ALSP were derived from the SIMNET philosophy (Weatherly, Seidel, and Weissman 1991) and included:

- Simulations need to be able to *cooperate over a common network* to form confederations.
- Within a confederation, *temporal causality* must be maintained.
- Simulations should be able to *join and exit a confederation* without major impact on the balance of the participating simulations.
- The system should be network-based with *no central controllers* or arbitrators.
- *Interactions do not require knowledge of confederation participants* and should support an object-oriented view of interactions.

3. LIFE AFTER SIMNET – THE NEED FOR STANDARDS

Several efforts to evaluate simulation technology during this timeframe supported and encouraged the need to develop and invest in distributed simulation. The Defense Science Board (DSB) task force on *Computer Applications to Training & Wargaming* stated "Computer-based, simulated scenarios offer the only practical and affordable means to improve the training of joint operational commanders, their staffs, and the commanders and staffs who report to them." (DSB 1988) This was followed by the report on *Improving Test and Evaluation Effectiveness*, which found that Modeling & Simulation (M&S) could be an effective tool in the acquisition process throughout the systems life cycle, especially if employed at the inception of the system's existence. (DSB 1989)

Then in 1991, the potential for distributed simulation for the military was realized in an operational context. The Battle of 73 Easting was a tank battle fought during the Gulf War between the U.S. Army and the Iraqi Republican Guard (Krause 1991). Despite being alone, outnumbered and out-gunned, the 2nd Armored Cavalry (ACR) struck a decisive blow destroying Iraqi tanks, personnel carriers and wheeled vehicles during the battle. The 2nd ACR had trained intensely before the battle both in the field and on SIMNET. Immediately, SIMNET's potential for network training was confirmed.

The following year, the DSB looked at the impact of advanced distributed simulation on readiness, training and prototyping (DSB 1993). They concluded that distributed simulation technology could provide the means to substantially improve training and readiness; create an environment for operational and technical innovation for revolutionary improvements; and transform the acquisition process.

Recognizing the importance of the SIMNET program and concerned that activity related to networked simulation was occurring in isolation, a small conference was held in April 1989 called "Interactive Networked Simulation for Training". The group believed that if there were a means to exchange information between companies, distributed simulation technology would advance more rapidly. The group also believed that technology had stabilized enough to begin standardization. The conference soon developed into the Distributed Interactive Simulation (DIS) Workshops.

Through these workshops, networked simulation technology and the consensus of the community were captured in proceedings and standards. The standards initially focused on SIMNET, but quickly evolved to include a broader range of technology areas. In 1996 the DIS Workshops transformed itself into a more functional organization called the Simulation Interoperability Standards Organization (SISO). An international organization, SISO is dedicated to the promotion of M&S interoperability and reuse for the benefit of a broad range of M&S communities.

One of the lasting contributions introduced during the time of the DIS Workshops was the definition of Live, Virtual, and Constructive (LVC) simulations (the term LVC was originally coined by GEN Paul Gorman). Live simulation refers to M&S involving real people operating real systems (e.g., a pilot flying a jet). A virtual simulation is one that involves real people operating simulated systems (e.g., a pilot flying a simulated jet). Constructive simulations are those that involve simulated people operating simulated systems (e.g., a simulated pilot flying a simulated jet). The LVC taxonomy is a commonly used way of classifying models and simulation.

4. DISTRIBUTED SIMULATION SCIENCE

Distributed simulation technology is based on the science of distributed systems. A distributed system is a collection of independent computers that appear to the users of the system as a single computer (Tanenbaum 1995). This definition has two aspects. The first one deals with hardware: the machines are autonomous; the second deals with software: the users think of the system as a single computer. This characterization provides a good foundation for distributed simulation technology. The goal of a distributed simulation is to create the illusion in the minds of the users that the entire network of simulations is a single system rather than a collection of distinct machines. Therefore, understanding how to separate the hardware and software design issues is key to developing the technology.

There are numerous challenges associated with building software to support distributed simulation. These include transparency, openness, scalability, performance, fault tolerance and security. Transparency is specifically important as it refers to hiding the distribution of components, so the system is perceived as "whole" and not a collection of "independent" simulations. Tools are needed to support the construction of distributed simulation software, specifically protocols that support the patterns of communication as well as naming and locating simulation processes.

There are two types of characteristics that distinguish the basic patterns of communication in distributed simulations: communication mechanisms and event synchronization. Communication mechanisms refer to the approach for exchanging data among two or more simulations. This includes message passing, shared memory, remote procedure call and remote method invocation. With message passing, there are several variations of delivery depending on the number of receivers. Data can be sent unicast to individual simulations, broadcast to every simulation, or multicast to a selected subset of simulations. Mechanisms such as publish/subscribe can also be used to define subsets of potential receivers.

Event synchronization refers to the approach for synchronizing the sending and receiving of data among the participants of a distributed simulation. Important properties include time, event ordering and time synchronization. Each simulation in a distributed simulation is assumed to maintain an understanding of time. That can include an informal relationship or a very strict adherence to a simulation or wall clock. In either case, simulations assign a timestamp to each message it generates. Event ordering refers to the way in which events are delivered to each simulation. There are several choices. Receive order delivers events regardless of the message time stamp and its relationship to the global distributed system. Timestamp order delivers events in an order directly related to a global interpretation of time. Time synchronization is related to both time and event ordering in that it's concerned with the global understanding of time in the distributed system. If global time is needed, there are a number of conservative and optimistic synchronization algorithms that can be used to achieve this state.

Communication mechanisms and event synchronization can be implemented in one of two ways: by individual simulations or by an operating system. There are three types of operating system commonly used in distributed systems. A network operating system is focused on providing local services to remote clients, and a distributed operating system focuses on providing transparency to users. Middleware combines the scalability and openness of a network operating system and the transparency and ease of use of a distributed operating system to provide general-purpose services. There are a number of trade-offs with the different approaches, including performance, scalability and openness. Modern distributed simulation has implemented a range of these approaches.

5. LIVE VIRTUAL CONSTRUCTIVE SIMULATION ARCHITECTURES

The most widely used LVC simulation architectures in the DoD are Distributed Interactive Simulation (DIS), High Level Architecture (HLA) and Test and Training Enabling Architecture (TENA). A fourth architecture exists but will not be covered in this paper. The Common Training and Instrumentation Architecture (CTIA) was developed to link a large number of live assets requiring a relatively narrowly bounded set of data for purposes of providing After Action Reviews on Army training ranges in the support of large-scale exercises. A description of CTIA can be found in (Henninger, et al 2008).

This second generation of distributed simulation architectures has evolved over the last 20 years using different technologies, standards and funding strategies. The following sections give a brief description of the architectures, characterizing its approach for communication and event synchronization.

5.1. Distributed Interactive Simulation

"The primary mission of DIS is to define an infrastructure for linking simulations of various types at multiple locations to create realistic, complex, virtual "worlds" for the simulation of highly interactive activities." (DIS 1994) Distributed Interactive Simulatio is based on the fundamental design principles of SIMNET. The goal of DIS is to create a common, consistent simulated world where different types of simulators can interact. Central to achieving this goal are protocol data units (PDUs); standard messages exchanged to convey state about entities and events. The PDUs comprise object data related to a common function, for example entity state, fire, detonation, and emissions were all frequently used PDUs. The Institute of Electrical and Electronics Engineers (IEEE) approved the first DIS standard in 1993 with 10 PDUs; the most recently published standard has 67 PDUs (IEEE 1278.1a 1998).

From an implementation perspective, simulation owners either custom-develop DIS interfaces or buy commercial products. There is also an open-source initiative, Open-DIS, to provide a full implementation of the DIS protocols in C++ and Java (McGregor and Brutzman 2008). The first DIS demonstration was held at the 1992 Interservice/Industry Training, Simulation and Education Conference (I/ITSEC) in San Antonio, TX. The demo included 20 companies, 25 simulators, and one long haul connection. The network layout for the demonstration is shown in Figure 2. A minimal set of PDUs (Entity State, Fire and Detonation) were used, and the interaction among participants was focused mainly on unscripted free-play (Loper, Goldiez, and Smith 1993).

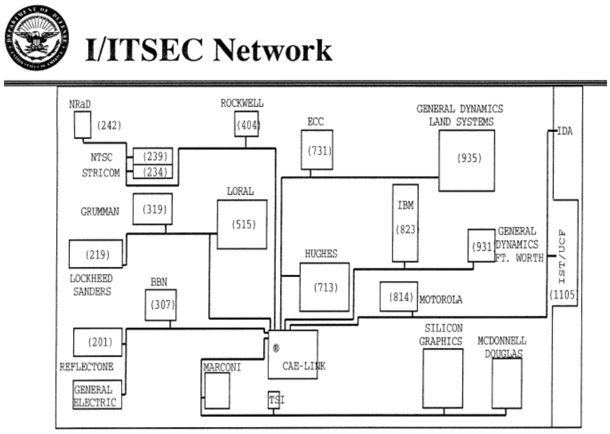


Figure 2: 1992 I/ITSEC DIS Demo Network

From a distributed system viewpoint, DIS is based on the idea that the network and simulators are integrated, i.e. there is minimal transparency. All communication about simulation entities and their interactions occurs via the PDUs. Reasonably reliable delivery is sufficient; dead reckoning algorithms are robust, so 1-2% missing datagram (randomly distributed) does not have an adverse impact on performance. As a result, most PDUs are sent using the best-effort user datagram protocol (UDP). The network is assumed to provide a certain level of assured services including, 300 msec end-to-end latency for "loosely coupled" interactions and 100 msec total latency for "tightly coupled" interactions (IEEE 1278.2 1995). Due to the potential for high latency in wide area networks, DIS is best for exercises on local area networks.

Interaction among DIS simulations is peer-to-peer and occurs using a message-passing paradigm. Since PDUs are broadcast to everyone on the network, bandwidth and computing resources can be consumed processing data that is not relevant to a specific simulation. A study of multicast communications occurred in the early 90's, with the idea of developing a new protocol for highly interactive applications. Developing a new protocol proved problematic and was abandoned. However, progress was made in understanding how to create multicast groups. One of the most commonly understood approaches to grouping information was called *Area of Interest* (Macedonia, et al 1995). Multicast was difficult to implement in DIS due to the lack of middleware or a distributed operating system, which could provide transparency to the simulations.

Time in DIS simulations is managed locally. Each simulation advances time at its own pace and clocks are managed locally using a local understanding of time. There is no attempt to manage time globally. Each PDU has a timestamp assigned by the sending simulation and PDUs are delivered to simulations in the order received. Simulations provide ordering locally, based on their understanding of time.

5.2. High Level Architecture

The High Level Architecture (HLA) program emerged in the mid-90s based on several assumptions. The first premise is no one simulation can solve all the DoD functional needs for modeling and simulation. The needs of the users are too diverse. Changing user needs define the second premise; it is not possible to anticipate how simulations will be used in the future or in which combinations. It is important, therefore, to think in terms of multiple simulations that can be reused in a variety of ways. This means as simulations are developed, they must be constructed so that they can be easily brought together with other simulations, to support new and different applications.

These assumptions have affected the HLA design in several ways. Clearly, the architecture itself must have modular components with well-defined functionality and interfaces. Further, the HLA separated the functionality needed for individual simulations (or federates) from the hardware infrastructure required to support interoperability. The HLA architecture is defined by three components:

- *Rules* that simulations must obey to be compliant to the standard
- *Object Model Template (OMT)* specifies what information is communicated between simulations and how it is documented
- *Interface Specification* document defines a set of services that simulators use to communicate information

The HLA standards began in 1995 under a government standards process managed by the Architecture Management Group. The DoD adopted the baseline HLA architecture in 1996 and the standards were moved to an open standards process managed by SISO (IEEE 1516 2010; IEEE 1516.1 2010; IEEE 1516.2 2010).

From a distributed system viewpoint, HLA is based on idea of separating the functionality of simulations from the infrastructure required for communication among simulations. This separation is accomplished by a distributed operating system called the Run-Time Infrastructure (RTI). The RTI provides common services to simulation systems and provides efficient communications to logical groups of federates. Data can be sent using both best effort (UDP) and reliable (TCP) internetwork protocols. An important distinction is that the HLA is not the same as the RTI. The RTI is an implementation of the HLA Interface standard, and thus there can be many different RTIs that meets HLA Interface standard. From an implementation perspective, HLA follows a commercial business model. There have been a variety of open-source initiatives, but none have produced an HLA compliant RTI.

In contrast to the static DIS PDUs, HLA uses the concept of OMTs to specify the information communicated between simulations. This enables simulation users to customize the types of information communicated among simulations based on the needs of the federation (what DIS called an exercise). When the OMT is used to define the data for a federation, the Federation Object Model (FOM) describes shared information (e.g., objects, interactions) and inter-federate issues (e.g., data encoding schemes). It didn't take long, however, for the community to understand the difficulty in developing FOMs. This led to the emergence of reference FOMs (SISO 2001), a mechanism for representing commonly used information, and Base Object Models (BOMs), a mechanism for representing a single set of object model data (SISO 1998).

From a communications perspective, HLA learned that broadcasting information to all simulations has serious implications on performance. The HLA defined a publication/subscription paradigm, whereby producers of information describe data it can produce and receivers describe data it is interested in receiving. The RTI then matches what is published to what has been subscribed. This approach maximizes network performance by allowing individual simulations to filter data it wants to receive at many different levels.

The HLA does include time management services to support event ordering. Both time stamp order, where messages are delivered to simulations in order of time stamp, and receive order, where messages are delivered to simulations in order received, are supported. Global time advance and event ordering is implemented by means of synchronization algorithms. The HLA interface specification supports the two commonly defined approaches: conservative and optimistic. While HLA provides global time management, use of these services is not required. Simulations can chose to advance time at its own pace, not synchronized with other simulations.

5.3. Test & Training Enabling Architecture

The Test and Training Enabling Architecture (TENA) emerged in the late 90's, after the HLA initiative was underway. The purpose of TENA is to provide the architecture and the software implementation necessary to do three things. First, TENA enables interoperability among Range systems, Facilities, Simulations, and C4ISR (Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance) systems in a quick, cost-efficient manner. It also fosters reuse for Range asset utilization and for future developments. Lastly, TENA provides composability to rapidly assemble, initialize, test, and execute a system from a pool of reusable, interoperable elements.

The principles of the TENA architecture include constrained composition, dynamic run-time characterization, subscription service, controlled information access, and negotiated quality of service. Constrained composition refers to the ability to compose the system for specific intended purposes that may be either transitory or permanent in nature.

Constraints apply to use of assets including physical proximity and location, coverage regions, performance capabilities, and subsystem compatibility. Dynamic, run-time characterization is focused on responding to many allowable compositions and permitting rapid reconfigurations. This is accomplished by establishing methods for self-description of data representations prior to or concurrent with data transfer or negotiating representation issues before operation starts. Similar to HLA, the subscription service is an object-based approach to data access, which matches producers and consumers of information. Due to the nature of many range assets, controlled information access is particularly important. Levels of access allow users to limit information access to a desired subset of all users. Since some services have significant performance and cost implications (e.g., data streams with large capacity requirements or strict latency tolerance), users can request specialized assets be allocated when needed. The negotiated quality of service protocols relies on the principal of separation of control information from data.

The TENA project uses a government standards process and is managed by Architecture Management Team (AMT). The AMT controls implementation content and Government members of the AMT recommend implementation changes. As such, no open standards have been published for TENA, however they do follow a formal process for standardizing object data.

From a distributed systems view, TENA separates the functionality of range assets from the infrastructure required to communicate among assets using middleware. The TENA Middleware is a common communication mechanism across all applications, providing a single, universal data exchange solution. Data exchanged among range assets is defined in object models, which can be sent using both best-effort (UDP) and reliable (TCP) internetwork protocols. A logical range object model is defined for a given execution, and can include both standard (time, position, orientation, etc.) and user-defined objects.

The TENA Middleware combines several communication paradigms, including distributed shared memory, anonymous publish-subscribe, remote method invocations, and native support for data streams (audio, video, telemetry, and tactical data links). Central to TENA is the concept of a Stateful Distributed Object (SDO) (Noseworthy 2008). This is a combination of a CORBA (Common Object Request Broker Architecture) distributed object with data or state. It is disseminated using a publish-subscribe paradigm, and subscribers can read the SDO as if it were a local object. An SDO may have remotely invocable methods.

Given the nature of real-time range assets, there is no requirement for time management to support event ordering. Messages are delivered to assets in the order they are received. The clock services defined in TENA are to manage time issues for the test facility. This includes synchronization and time setting services, as well as maintaining a global clock for exercises.

6. SIMULATION INTEROPERABILITY

Modeling and simulation interoperability is defined as "the ability of a model or simulation to provide services to and accept services from other models and simulations, and to use the services so exchanged to enable them to operate effectively together". Interoperability exists when different systems exhibit the "same" behavior when stimulated by a set of standard procedures. (DoD 2010). One commonly accepted approach for describing interoperability is the Levels of Conceptual Interoperability Model (LCIM). As shown in Figure 3, the LCIM identifies seven levels of interoperability among participating systems and the complexity of interoperations (Tolk 2003). The LCIM associates the lower layers with the problems of simulation interoperation while the upper layers relate to the problems of reuse and composition of models.

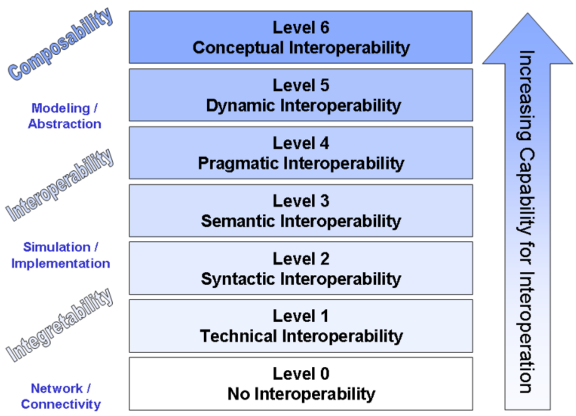


Figure 3: Levels of Conceptual Interoperability Model

DIS, HLA, and TENA are solutions focused on the lower-layers of the LCIM. Since DIS, HLA, and TENA-based federations are not inherently interoperable with each other, additional steps are needed to enable effective communication among those simulations. These steps typically involve using gateways or bridges between the various architectures. While effective, these approaches can introduce increased risk, complexity, cost, level of effort, and preparation time into the simulation event.

Gateways and bridges, however, do not address the issues of reuse and composition associated with the upper layers of the LCIM. As stated in (Tolk 2003), "simulation systems are based on models and their assumptions and constraints. If two simulation systems are combined, these assumptions and constraints must be aligned accordingly to ensure meaningful results". Thus ability to reuse supporting models, personnel (expertise), and applications across the different architectures is limited.

The lack of interoperability between the different architectures introduces a significant and largely unnecessary barrier to the integration of live, virtual, and constructive simulations. This barrier needs to be greatly reduced or eliminated.

7. CONCLUSIONS

Distributed simulation architectures in use within the DoD today have all been designed to meet the needs of one or more user communities. These architectures continue to evolve and mature based on changing requirements. The existence of multiple architectures allows users to select the methodology that best meets their individual needs. It also provides an incentive for architecture developers and maintainers to competitively keep pace with technology and stay closely engaged with emerging user requirements (Henninger, et al 2008).

One of the challenges in achieving the transparency desired in distributed simulation however is that multiple architectures exist. Incompatibilities between DIS, HLA and TENA require the development of point solutions to effectively integrate the various architectures into a single, unified set of simulation services. Integration is typically achieved through gateway solutions, which can often restrict users to a limited set of capabilities that are common across the architectures. The successful integration of distributed simulations will continue to rely upon the development of simulation standards.

Despite the advances in distributed simulation technology and standards, challenges remain. In a 2008 survey on future trends in distributed simulation, the most promising areas of research for the simulation community were identified as distributed simulation middleware, human-computer-interfaces, and the semantic web/interoperability (Strassburger, Schulze, and Fujimoto 2008). Within simulation middleware, the greatest needs identified were plug-and-play capability, standardization and interoperability between different standards, semantic connectivity and ubiquity (accessible anywhere with any device).

The results of this survey combined with the findings of the Live Virtual Constructive Architecture Roadmap panel (Henninger, et al 2008) define the needs for the next generation of distributed simulation. The DoD has been a driving force in shaping the technology and standards for nearly 30 years, and they will continue to have a major role defining the way forward.

REFERENCES

Defense Science Board, 1988. Computer Applications to Training and Wargaming. Final Report, ADA199456, May.

Defense Science Board, 1989. Improving Test and Evaluation Effectiveness. Final Report, ADA274809, December.

Defense Science Board, 1993. Simulation, Readiness and Prototyping. Final Report, ADA266125, January.

DIS Steering Committee, 1994. The DIS Vision: a map to the vision of distributed simulation. IST Technical Report, May.

Henninger, A, Cutts, D, Loper, M, Lutz, R, Richbourg, R, Saunders, R, Swenson, S. Live Virtual Constructive Architecture Roadmap (LVCAR) Final Report. M&S CO Project No. 06OC-TR-001, September 2008.

IEEE 1278.1a – 1998. Standard for Distributed Interactive Simulation - Application Protocols, Version 6 (amendment to IEEE 1278.1-1995).

IEEE 1278.2 – 1995. Standard for Distributed Interactive Simulation - Communication Services and Profiles.

IEEE 1516 – 2010. Standard for Modeling and Simulation High Level Architecture - Framework and Rules.

IEEE 1516.1 – 2010. Standard for Modeling and Simulation High Level Architecture - Federate Interface Specification.

IEEE 1516.2 – 2010. Standard for Modeling and Simulation High Level Architecture - Object Model Template (OMT) Specification.

Krause, M., 1991. The Battle of 73 Easting, 26 February 1991, a historical introduction to a simulation. Draft report, US Army Center of Military History, May 2 May.

Loper, M., 2008. *Interview with COL James Shiflett on History of Distributed Simulation*. Live Virtual Constructive Architecture Roadmap (LVCAR) project. Report, M&S CO Project No. 06OC-TR-001, September.

Loper, M., Goldiez, B., and Smith, S., 1993. The 1992 I/ITSEC Distributed Interactive Simulation Interoperability Demonstration. *Proceedings of the 15th Interservice/Industry Training Systems and Education Conference*. November 29 - December 2, Orlando (Florida USA).

Macedonia, M, Zyda, M, Pratt, D, Brutzman, D, Barnham, P., 1995. Exploiting Reality with Multicast Groups. *IEEE Computer Graphics and Applications*, 15 (5), 38-45.

McGregor, D, Brutzman, D., 2008. Open-DIS: An Open Source Implementation of the DIS Protocol for C++ and Java. *Proceedings of the Fall Simulation Interoperability Workshop*. 15-19 September, Orlando (Florida USA).

Miller, D. and Thorpe, J., 1995. SIMNET: the advent of simulator networking. *Proceedings of the IEEE*, 83 (8), 1114 – 1123.

DoD, 2010. *DoD Modeling and Simulation (M&S) Glossary*. Available from: http://www.msco.mil/ [accessed 15 July 2011].

Noseworthy, J. R., 2008. The Test and Training Enabling Architecture (TENA) Supporting the Decentralized Development of Distributed

Applications and LVC Simulations. *Proceeding DS-RT '08 Proceedings of the 2008 12th IEEE/ACM International Symposium on Distributed Simulation and Real-Time Applications*, pp. 259-268. October 27-29, Vancouver, (British Columbia, Canada).

Simulation Interoperability Standards Organization, 1998. Reference FOM Final Report. SISO-REF-001-1998, March 9.

Simulation Interoperability Standards Organization, 2001. BOM Study Group Final Report. SISO-REF-005-2001, May 15.

Strassburger, S, Schulze, T, Fujimoto, R., 2008. Future Trends In Distributed Simulation and Distributed Virtual Environments: Results Of A Peer Study. *Proceedings of the Winter Simulation Conference*. December 7-10, Miami (Florida, USA).

Tanenbaum, AS., 1995. *Distributed Operating Systems*. 1st Edition New Jersey: Prentice Hall.

AUTHORS BIOGRAPHY

Margaret Loper is the Chief Scientist for the Information & Communications Laboratory at the Georgia Tech Research Institute. She holds a Ph.D. in Computer Science from the Georgia Institute of Technology, a M.S. in Computer Engineering from the University of Central Florida, and a B.S. in Electrical Engineering from Clemson University. Margaret's technical focus is parallel and distributed simulation, and she has published more than 50 papers as book chapters, journal contributions, or in conference proceedings. She is a senior member of the IEEE and ACM, and member of the Society for Modeling and Simulation. She is a founding member of the Simulation Interoperability Standards Organization (SISO) and received service awards for her work with the Distributed Interactive Simulation (DIS) and High Level Architecture (HLA) standards and the DIS/SISO transition. Her research contributions are in the areas of temporal synchronization, simulation testing, and simulation communication protocols.

RESEARCH GAPS FOR ADAPTIVE AND PREDICTIVE COMPUTER-BASED TUTORING SYSTEMS

Robert A. Sottilare [(a)], Stephen Goldberg [(b)] and Paula Durlach [(c)]

[(a)]U.S. Army Research Laboratory – Human Research and Engineering Directorate
Learning in Intelligent Tutoring Environments (LITE) Laboratory

[(b,c)] U.S. Army Research Institute

[(a)]robert.sottilare@us.army.mil [(b)]stephen.goldberg@us.army.mil [(c)]paula.durlach@us.army.mil

ABSTRACT

Researchers continue to enhance individual computer-based training capabilities to support self-directed learning and account for individual differences (e.g., personality or domain competence). Student-centric tutoring approaches recognize that each student's unique affect, motivation, skills, knowledge, preferences and experiences should influence the content, flow and challenge level of computer-based instruction. In other words, these individual differences should be the basis for adapting instruction to promote learning and predicting the future learning states of the student. This article explores current trends in adaptive and predictive computer-based tutoring methods, identifies gaps and discusses opportunities for future research. The intent of this paper is to introduce concepts discussed in the "adaptive and predictive computer-based tutoring" track of the the Defense and Homeland Security Simulation (DHSS) Workshop 2011.

Keywords: adaptive training, predictive modeling, intelligent tutoring systems, computer-based tutoring

1. INTRODUCTION

In March of 2011, a group of U.S. military scientists met to discuss the research and development of adaptive methodologies for computer-based military training. The Adaptive Training Workshop brought together representatives from each of the military services, the Defense Research Projects Agency (DARPA), the Advanced Distributed Learning (ADL) Co-Lab, and the Department of Education. In part, this workshop influenced the scope and content of this article which has been focused to specifically identify research gaps for adaptive/predictive computer-based tutoring systems that were identified during this March 2011 workshop.

For purposes of the workshop, adaptive training was defined as any training that was adjusted to meet the specific learning needs of individuals or teams. Particular attention was paid to research programs involving intelligent tutor technology which uses artificial intelligence techniques to adapt instructional information to match the student's cognitive and affective needs. Intelligent tutors produce instruction that is the product of an interaction between a student model, representing the state of knowledge, motivation, personality, and other student variables, an expert model representing the material to be trained and a pedagogical model representing the training methods to be employed in presenting the knowledge and skills.

1.1. Adaptive Tutoring Model

By way of example, the **Figure 1** illustrates a functional model of an adaptive tutoring system (Sottilare, 2010) which was adapted from Beck, Stern and Haugsjaa (1996) tutoring model. The major components include a student model (also known as a user, trainee or learner model) which generally contains information about the student's performance, and their overall competency in the domain being trained. It may contain information about their physiological state and their behaviors where this information is used to ascertain their cognitive (e.g., level of engagement) and affective state (e.g., emotions).

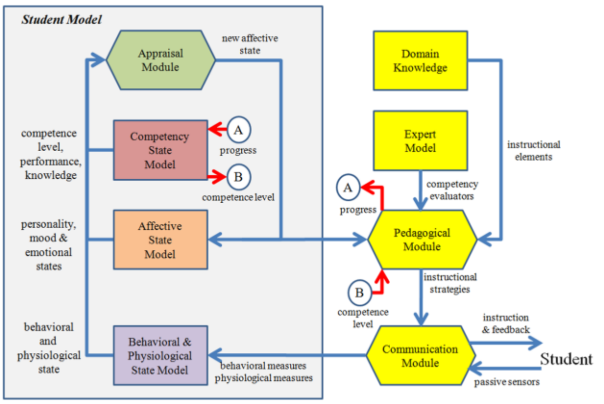


Figure 1: Adaptive Tutoring Model (Sottilare, 2010)

The pedagogical module assesses student progress based on the student's interactions and data in the student model. It uses this information to determine which instructional strategies (e.g., direction, support or questioning) to employ during the training session.

The expert model is used to measure the progress of the student in the learning domain defined by the domain knowledge. The domain knowledge also defines challenge levels, options for feedback and content presentation.

The communication module is the student interface and includes mechanisms (e.g., visual displays, speakers or haptic devices for touch) to present instruction and feedback to the student. It may also include sensor input for physiological and behavioral measures to assess the cognitive and affective state of the student.

1.2. Tutor Adaptability

Similar to adaptive training, we defined adaptive tutoring "the ability of any computer-based tutor to adjust to meet the specific learning needs of individuals or teams." Student model data allows for the assessment of the student's cognitive (e.g., motivation, comprehension, level of engagement) and affective states (e.g., mood, emotions) by the tutor and is a basis for the tutor to adapt to the student by choosing appropriate instructional strategies. The effectiveness of the tutor's instructional decisions (or strategies) is limited by the tutoring system's ability to accurately classify the student's state. While human tutors generally perform this function with some difficulty, computer-based tutors can use machine learning classifiers and other techniques to evaluate real-time and historical data to interpret the student's current state and adapt the training content, flow and feedback to match. The student data might include, but is not limited to performance, behavioral and physiological sensor data, demographic data, personality profiles, mood surveys and student-system interaction (e.g., graphical user interface selections like check-boxes).

1.3. Tutor Predictive Accuracy

Adaptive tutoring indicates that the tutor assesses and reacts as needed. It would be useful for the tutor to be able to assess and predict future states so the tutor could be proactive and head off any negative aspects of the training. For example, the flow of training could be adapted to include interruptions to refocus engagement and reduce training time.

The inclusion of real-time data over the course of a training scenario allows for the prediction of future states and thereby makes the tutor predictive rather than reactive. Ideally, computer-based tutors would be able to fully perceive student behaviors and interpret physiological measures through unobtrusive sensing methods to predict the student's cognitive and affective states. Predicting the student's state is a necessary first step in selecting optimal instructional strategies (e.g., scaffolding for developing students).

Figure 2 illustrates a generic state transition model that might be utilized to examine localized trends and "dead reckon" future states. In the figure, the slope of the predictive vector (shown as an arrow) represents the strength of the trend from State A to State B. A state transition zone can vary in width from a single line (immediate transition shown by blue arrow) to an established position in the new state as shown by the red arrow.

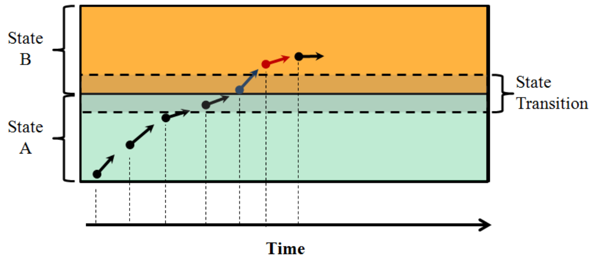


Figure 2: State Transition Model

Research thrusts needed to realize a fully adaptive and predictive tutoring system are reviewed below with particular focus on student modeling, authoring and expert modeling, and instructional strategy selection.

2. CHALLENGES IN ADAPTIVE/PREDICTIVE TUTORING

2.1. Student Modeling

Student models are often referred to interchangeably as student models, user models or trainee models. In order to make appropriate decisions about instructional content, flow, challenge level and feedback to the student, the tutor must first construct a sufficient student model. Ideally, this model would include everything the tutor needs to know about the student to guide the student through the learning experience, but some "tutoring systems" do not explicitly contain a student model. Those tutors that do have student models generally focus on student knowledge and performance to support instructional decisions.

It seems intuitive that having a student model would result in better instructional decisions than not having one, but having a student model isn't sufficient to ensure superior learning since you still must have an effective instructional strategy, intervention or pedagogical technique. In a truly "intelligent" tutoring system (ITS), the student model has to accurately reflect the state of the student and the instructional strategy (e.g., feedback, reflection, pumping, questioning, supporting) must be appropriate to the situation and the student's state to optimize learning.

There are two major challenges (and many questions) in deciding what needs to be in the student model: identifying what information is relevant to instructional decisions; and collecting that information unobtrusively so as not to interfere with the learning process (Sottilare and Proctor, 2011). Outstanding questions include: What inputs, processes and outputs are critical for student models to support adaptive/predictive training applications? Do student models adequately address individual differences?

So what types of data are included in student models today? Currently, data models include both unprocessed and processed (or derived) data sets. Unprocessed data could include self-reported data like demographics, opinions and survey information, but could also include raw physiological or behavioral data from sensors (e.g., cameras, recording devices). Derived data sets include student states (e.g., competence-level, cognitive state, affective state).

Student models may not be necessary for training simple, drill and practice type tasks, but as the type of task becomes more complex or ill-defined then the value of the student model should take on more importance.

Another consideration in student modeling is the notion of static and dynamic data types. For our purposes, we define "static" data as student data that remains unchanged for the life of the training event and includes, but is not limited to personality data (e.g., openness). "Dynamic" data changes during training and includes, but is not limited to performance measures.

Other factors are referred to as macro/micro or global/local parameters. For example, local adaptation includes actions that are based on recent student events (e.g., selected response "B" and the correct answer is "C") and an intervention might be chosen based on this small sample of performance. A more global approach focused on static personality data leaves little room for prediction or adaptation during training. Hybrid models take global variables into account when initializing strategies and adapt/change through local variables. We need both.

Social interaction data (e.g., trust, communication) may be more important for training on tasks that require more than the knowledge and skills of the individual student (e.g., team training). Individual attributes like as cooperation, adaptability, openness/friendliness and situational awareness might have significant value in obtaining objectives and in assessing team performance as part of a collective student model.

Finally, the literature seems to agree on the importance of accounting for knowledge and competence in instructional design and delivery, and so they are important to student modeling. However, the influence of other factors (e.g., ethics/values, technology acceptance) on the learning effectiveness of tutors remains an open question.

2.2. Authoring Tools and Expert Modeling

A grand challenge in the development, usability and efficacy of computer-based tutors is the ease with which tutors can be configured configuring tutors to support different training domains and populations. Today, computer-based tutoring systems are generally handcrafted products with little standardization, interoperability or reusability. Authoring tools are needed to support the development of training content and expert models by domain experts with little programming skills. Interoperability standards would go a long way toward making the modularity and reuse of tutor components (model structures, communication protocols and scenario content) easy to produce, modify and maintain.

The development of expert models remains a key cost in developing tutoring systems. Expert models represent ideal student performance to which actual student performance is compared. Expert models are typically painstakingly developed through observation of subject matter experts performing specific tasks.

Some of the questions driving research in the area of expert modeling include: How does the state of knowledge of expertise in some domains differ from others? How do we describe and formalize expertise in general? How do we articulate learning objectives in such a way that they support development of expert models?

Finally, "expert model" may be a misleading term. It may really be a journeyman model with standards defined so that students complete the training when they obtain sufficient competency to get the job done, but not at an expert level. An expert model is really a model of what is correct. The student's performance is compared to correct performance.

There was a discussion over whether or not an expert model should be computational or not. A position was put forth that for domains that are stable the effort to develop a computational model should be expended. For less stable tasks were there maybe multiple ways to accomplish the task or adaptive behaviors are appropriate it is hard to articulate what expertise is and therefore very difficult to model quantitatively.

2.3. Instructional Strategy Selection

A key research task in developing computer-based tutors that can select appropriate instructional strategies is too observe, assess and model the behaviors of experts. In other words, build an expert model for instructing that is adapted not only for the student, but also for the learning context. This task analysis is complicated by the fact that many teachers broadcast information to many students and it is difficult to sort out what behaviors make a difference in individual learning.

Major functions for the instructor are diagnosis, remediation, prescription, demonstration, feedback, motivational support, attention orienting, and questioning. One of the most difficult tasks for a computer-based tutor is to understand what the student knows, assess options to correct deficiencies and then pick the optimal strategy. For example, feedback has to be at the right level of understanding to be useful to the student.

The instructor model is considered by many to be the long pole in the computer-based tutoring tent. Another challenge is that diagnosis doesn't work well without a good student model. Providing all of the pedagogical approaches that are needed is very difficult using the current artificial intelligence techniques. A question is often raised about whether the computer-based tutors should replace teachers or whether we should be developing methods for providing decision support aids to teachers in the near term. The authors view the intended use of tutoring technologies as supportive of training tasks in environments where human tutors are either unavailable or impractical.

3. DISCUSSION

3.1. Assessing Computer-based Team Tutor Performance

What should be the basis for assessing the maturity and effectiveness of computer-based team tutoring technologies? Sottilare and Gilbert (2011) identified several factors that should be considered in this assessment: adaptability, perception, accuracy, instructional strategy selection, interoperability and most importantly, learning effect.

Adaptability of the tutor is the capability to understand the student's learning needs and change the content, flow and interaction (e.g., feedback, questioning) prior to and during instruction to meet those learning needs. Adaptability is the result of perception, accurate assessment of student state and optimized instructional strategies.

Perception, in this context, is the ability of the computer-based tutor to sense and understand the student's physiological and behavioral data to populate the student model. Today, the gold-standard for perception is the human tutor who uses behavioral cues to interpret the student's state. This is more art and less science since cues can be misinterpreted. Good human tutors use multiple cues and the student's performance to assess the "readiness to learn".

"Readiness to learn" is a multidimensional state defined here to be the student's level of engagement, their motivation, their understanding of prerequisite skills and their affective state (e.g., personality factors, mood and emotions). Computer-based tutors have the potential to integrate additional sensor information (e.g., physiological data including heart rate, neurological data and respiration rate) beyond the capabilities of human tutors. The limiting factors in maximizing the perceptive powers of computer-based tutors lies in their abilities to sense student behaviors and physiological data unobtrusively, and then use that data to accurately model the student's state.

Computer-based tutoring systems use a variety of methods to evaluate student data and accurately determine student state (e.g., cognitive and affective). Machine learning classifiers are extensively used to assess state and include, but are not limited to rule-based classifiers, Bayesian networks, decision trees and regression algorithms. It is generally accepted that improvements to state classification will result in higher probability of selecting an appropriate and more effective instructional strategy.

Examples of instructional strategies include scaffolding, modeling, cooperative learning and prior knowledge activation (Cooper, 1993). Scaffolding places heavy emphasis on support early in learning and gradually less support as the student's competence grows (Cooper, 1993). In modeling, the tutor demonstrates specific concepts and skills for the learner (Bandura, 1986). Finally, cooperative learning leverages the experience of peers to engage with learners to improve their knowledge and understanding of instructional content (Wells, 1990).

The accessibility of instructional content is enhanced by the interoperability of the tutor. The easier it is to link computer-based tutors to instructional media, the more useful and accessible that tutor will be. Recently, instructional developers (Thomas and Young, 2009) have adapted tutor interfaces to accept what has traditionally been entertainment content (e.g., computer games) to enhance the engagement level of students resulting in "serious games" or games for training.

The bottom line for any tutor (human or instructional technology) is its positive influence on learning or effect size. Bloom (1984) described a two-sigma (2σ) difference between a learner's achievements in a classroom environment vs. a learner's achievement with a one-on-one tutor. Kulik (1994) analyzed 97 research studies on tutors and found that most tutoring systems have an average difference (or "effect size") of 0.32σ. This low compared to expected larger effect sizes ($> 2.0\sigma$).

3.2. Conclusions

While computer-based tutoring had been around for some time, there is little evidence of their use in military training. This is likely in part due to the absence of authoring tools to make tutors easy to develop and maintain without the need for technical experts.

Tutors for the most part have been developed for structured (well-defined) knowledge domains such as algebra, physics and trouble shooting. Tutors for ill-defined domains that require decision making in the face of complex, and confusing situations are a significant challenge given the maturity of tutoring technology today.

Competence and state of knowledge are the key elements in student models. The student model can also take into account more local time sensitive information like attention. Other potential components of student models need to have their contributions investigated further. The richer the student model the finer grained the tutoring objects can be and the more accurate the assignment of instructional strategies.

3.3. Recommendations for Future Research

Pedagogy was seen as a major hurdle that needed further research, since there were very few firm guidelines on the relationship between student performance diagnosis and the ensuing instructional method.

It makes sense that the more the tutor knows about the student, the more effective its pedagogical decisions will be. The student model in today's tutors is insufficient to account for individual differences (e.g., personality), which include states (e.g., motivation, engagement) and traits (e.g., preferences). The influence of individual differences is in many cases unknown and requires additional research. An expanded student model needs to be developed based on empirical evidence to define the influence of individual differences.

Another recommended area for research is accelerated learning and retention. While the effect of computer-based tutoring technologies on learning is still not well understood, their ability to accelerate learning is even less clear.

Retention continues to lag behind learning research due to issues with retaining participants over a series of experiments, but "it does little good to attain a higher level of competence quickly if it leads to poorer knowledge and skill retention" (Andrews & Fitzgerald, 2010).

Finally, five instructional strategy research topics that would benefit from additional investment are analysis, diagnosis, prescription, mental model mismatch (misconceptions) and demonstration.

REFERENCES

Andrews, D. and Fitzgerald, P. (2010). Accelerating learning of competence and increasing long-term learning retention. In Proceedings of the *International Training and Education Conference* (ITEC), London, UK, 18-20 May 2010.

Bandura, A. (1986). Psychological modeling: Conflicting theories. Chicago, IL: Aldine-Atherton.

Beck, J., Stern, M., and Haugsjaa, E. (1996) Applications of AI in Education, *ACM Crossroads* 3(1), 11-15.

Bloom, Benjamin S. (1984) The 2-sigma problem: The search for methods of group instruction as effective as one-to-one tutoring, Educational Researcher 13: 4-16.

Clark, R. (2002). Learning outcomes: The bottom line. *Communication Education, 51(4), 396-404.*

Cooper, J.D. (1993). Literacy: Helping children construct meaning (2nd ed.). Boston, MA: Houghton Mifflin.

Kulik, J. A. (1994). Meta-Analytic studies of findings on computer-based instruction. In E. L. Baker and H. F. O'Neil (Eds.), Technology Assessment in Education and Training (pp. 9-33). Hillsdale, NJ: Lawrence Erlbaum Associates.

Sottilare, R. (2010). Toward the Development of an Intelligent Tutoring System for Distributed Team Training through Passive Sensing. In V. Aleven, J. Kay, & J. Mostow (Eds.), Proceedings of the 10th International Conference on Intelligent Tutoring Systems (ITS '10), LNCS (Vol. 6095, pp. 411-413). Berlin: Springer.

Sottilare, R. and Gilbert, S. (2011). Considerations for tutoring, cognitive modeling, authoring and interaction design in serious games. *Authoring Simulation and Game-based Intelligent Tutoring workshop at the Artificial Intelligence in Education Conference (AIED) 2011*, Auckland, New Zealand, June 2011.

Sottilare, R. & Proctor, M. (*in press*). Passively Classifying Student Mood and Performance within Intelligent Tutors. *Journal of Educational Technology.*

Thomas, J. and Young, R. M. (2009). Dynamic Guidance in Digital Games: Using an Extensible Plan-Based Representation of Exploratory Games to Model Student Knowledge and Guide Discovery Learning. In Working Notes of the Intelligent Educational Games Workshop at the International Conference on Artificial Intelligence and Education (AIED 09), Brighton, UK, July, 2009.

Wells, G. (1990). Creating the conditions to encourage literate thinking. Educational Leadership, 47(6), 13-17.

AUTHORS BIOGRAPHIES

Dr. Robert A. Sottilare is the Associate Director for Science & Technology within the U.S. Army Research Laboratory - Human Research and Engineering Directorate (ARL-HRED). Dr. Sottilare has over 27 years of experience as both a U.S. Army and Navy training and simulation researcher, engineer and program manager. He leads the international program for ARL's Simulation and Training Technology Center (STTC) and participates in training technology panels within both The Technical Cooperation Program (TTCP) and the North Atlantic Treaty Organization (NATO). He has a patent for a high resolution, head mounted projection display (U.S. Patent 7,525,735) and his recent publications have appeared in the Educational Technology Journal, the Journal for Defense Modeling and Simulation, the NATO Human Factors and Medicine Panel's workshop on Human Dimensions in Embedded Virtual Simulation and the Intelligent Tutoring Systems Conference 2010. Dr. Sottilare is a graduate of the Advanced Program Managers Course at the Defense Systems Management College at Ft. Belvoir, Virginia and his doctorate in modeling & simulation is from the University of Central Florida. The focus of his current research program is in machine learning, student modeling and the application of artificial intelligence tools and methods to adaptive training environments. Dr. Sottilare oversees the research conducted at the Learning in Intelligent Tutoring Environments (LITE) Laboratory at STTC in Orlando, Florida.

Dr. Stephen Goldberg has over 35 years experience as a U.S. Army Research Institute for the Behavioral and Social Sciences (ARI) Research Psychologist. Dr. Goldberg received a doctorate in Cognitive Psychology from the State University of New York at Buffalo in 1974. He has served as an ARI researcher at locations in Alexandria, VA, and Ft. Knox, KY. In 1984 he was selected to be ARI's liaison to the U.S. Army's Training and Doctrine Command (TRADOC), Ft. Monroe, VA. Dr. Goldberg moved to Orlando in 1989 and became the Chief of ARI's Technology Based Training Research Unit. He supervises a research program focused on on feedback processes and After Action Review, training and performance in virtual simulations and games, and adaptive training.

Dr. Paula J. Durlach is a research psychologist and team leader at the U.S. Army Research Institute for the Behavioral and Social Sciences. She received her Ph.D. in Psychology from Yale University, and conducted postdoctoral research at both the University of Pennsylvania and Cambridge University. Her research career has focused on learning processes in several guises from simple associative learning to complex training.

CULTURAL CONSIDERATIONS IN A SIMULATED EMERGENCY

Elizabeth Whitaker[(a)], Rita Gonzalez[(b)], Ethan Trewhitt[(c)], Reid MacTavish[(d)], Margaret Loper[(e)]

[(a,b,c,d,e)]Georgia Tech Research Institute

[(a)]betty.whitaker@gtri.gatech.edu [(b)]rita.gonzalez@gtri.gatech.edu [(c)]ethan.trewhitt@gtri.gatech.edu
[(d)]reid.mactavish@gtri.gatech.edu [(e)]margaret.loper@gtri.gatech.edu

ABSTRACT
This paper describes the exploration cultural considerations in a simulated emergency. We explored the use of cultural variables as components of human behavior modeled by an agent-based system. The goal of the project was to provide a framework to enable training for first responders in the reactions of populations in crisis. As part of this exploration we have built a prototype that models a small population of individuals going about their daily routines and responding to the societal stress introduced by a flu outbreak and by an earthquake. The model allows a user to explore the ways in which habits and behaviors affect the spread of the disease. We have constructed a framework that allows for experimentation with the influences of a number of cultural and behavioral.

Keywords: agent-based simulations, cultural variables, simulation-based training

1. INTRODUCTION

The Georgia Tech Research Institute team researched the representation in an agent-based model of cultural considerations in a simulated emergency. The goal of this project was to research new ways for first responders to experience training that would allow them to become familiar with possible population behaviors and reactions to crisis and to the presence and activities of the first responders. This would allow them to explore appropriate interactions to anticipate the population's reactions. We have designed and prototyped a representation and framework to be populated with cultural and psychological variables that can be used to drive behaviors of the agents in the simulation. Agents will be assigned variable values representative of the cultures in the population being studied

2. CULTURAL MODELING

This project explores the use of cultural variables as components of human behavior modeled by an agent-based system. As part of this exploration we have built a prototype that models a small population of individuals going about their daily routines and responding to the societal stress introduced by a crisis such as a flu outbreak or an earthquake. The phase I model allows a user to explore the ways in which habits and behaviors affect the spread of the disease. We have constructed a framework that allows for experimentation with the influences of a number of cultural and behavioral variables on the spread of the disease. A user can change the value of the behavioral variables for individual agents while performing what-if experiments to gain insight into the effects on the outcome. Different cultural and behavioral variables can be inserted into the model to experiment with their effects on the disease spread. The current model instantiation is based on a hypothetical culture. As the model matures, it would be useful to model multicultural populations, such as might be represented in a US city.

Culture, for the purposes of this project, refers to the arts, customs, habits, beliefs, values, behaviors, or objects that constitute a people's way of life and the human interactions, rules, and processes that bind and separate people as individuals or groups. The cultural variables considered as candidates for inclusion in our behavioral model come from literature (e.g., Hofstede, D'Andrade and Strauss).

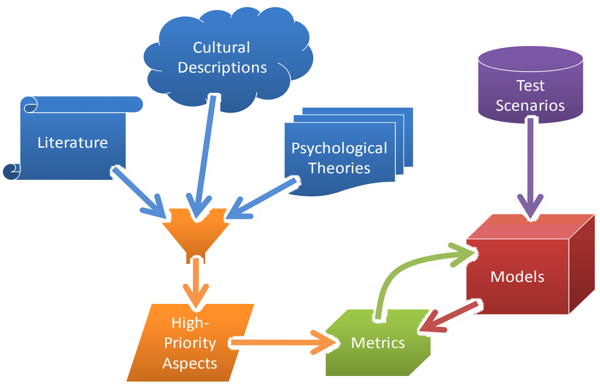


Figure 1. Integrating cultural variables into behavioral models

Every model is a simplified representation of the real world. Models are typically constructed for the use of analysts wishing to gain a greater understanding of a situation and its influences. Tools and methodologies to support this will be most effective when they include representations of behavioral drivers that support the analyst's view of the population's cultural and psychological makeup. To build a representation of the external world, a modeler or analyst must choose which entities are most important to include in the model, based on information from subject matter experts, the intended use of the model, and the relative impact that each entity has on the system's behavior. This research focuses on the use of cultural variables and explores the choices of those variables and the implementation of their effects on the agent behaviors.

During phase I we chose some sample variables from the following three psychological theories:

- **Social Identity Theory** – members of all social groups come together around at least one common denominator
- **Social Influence Theory** – people are driven to agree with others in order to be liked and accepted by them
- **Attribution Theory** – issues of causation and explanation are crucial to understanding social phenomena and all human interaction

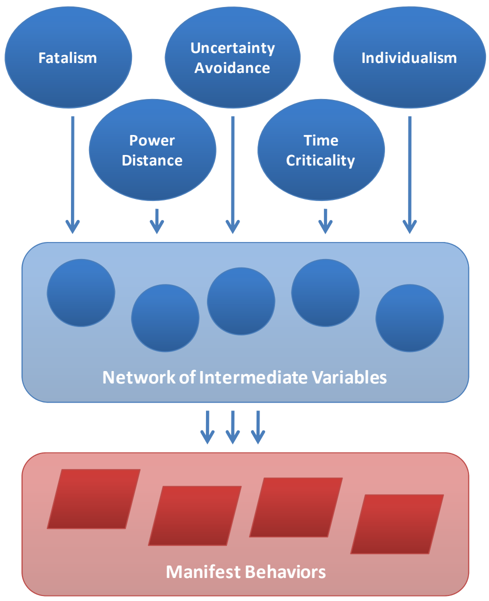


Figure 2. Relationship between high-level independent variables and low-level manifest behaviors (Phase I)

3. PHASE I SCENARIO

For the initial experiment in modeling the flu outbreak, we have chosen the following set of variables, which are loosely based on Hofstede's work. Figure 2**Errore. L'origine riferimento non è stata trovata.** shows how these variables generally relate to the manifest behaviors.

- Fatalism
- Uncertainty avoidance
- Individualism
- Power distance
- Time-criticality

We include a number of related behavioral variables such as busyness, age, and fastidiousness that will help guide the behaviors of the simulated individuals (agents). These variables lead to activities that in the model impact the spread of the disease: getting inoculated, going to work, shopping, and visiting a doctor.

With cultural variables and their expected effects as identified by psychologists and other behavioral experts, models of this type could be used to plan public health motivational campaigns and to identify the locations, schedules, and types of services that would most likely be effective in inhibiting the spread of the disease. This framework would also be useful for modeling population reactions to other types of crises and associated educational or public health approaches, emergency response activities, and government policies, in order to support human decision-making.

4. PHASE I MODELING APPROACH

We created a modeling environment in which the agents representing individuals in the society would go about their daily lives in the context of the unfolding crisis, the flu epidemic. The prototype was meant to demonstrate the use of this approach to explore the spread of disease in a community:

- Explore the ways that human behaviors and cultural variables might be represented
- Explore how human behaviors affect disease spread
- Explore ways to support decision-makers to understand behaviors and motivations

As we began to develop the prototype model we went through the following questions to guide our model development and to describe the agent-based modeling activity.

Table 1. Modeling Questions

Question	Answer
What questions is this model trying to answer?	How does the culture of a society affect the spread of the flu?
What are the agents?	Individual people
How will we specify the agent behaviors?	Agent attributes are encoded as a series of causally-related cultural variables, and variables influence agent decision resulting in specific behaviors.
Variables influence agent decision resulting in specific behaviors.	Agents (people) meet in geographic space and come into contact based on some probability; the flu will spread (with a given changeable probability) if a sick person contacts a well person.
What is the	The agents interact in a

context?	geographical space representing homes, work, schools, shopping and hospital. At the start, a “seed” group has the flu already.
How will they behave?	Each person has a specific instantiation of cultural variables and reacts according to his/her preferences.

In addition to the variables given to each agent, our agent-modeling environment allows us to create a set of behavioral rules that determine how the agents will make decisions as the model plays out. The rules that we developed for this model are described in Table 2 below:

Table 2. Selected Individual Agent/Person Variables

Individual *State* Variables
age
sick? (is the person visibly sick?)
inoc? (is the person immune due to inoculation?)
tried-inoc? (did the person try to get inoculated?)
infectious? (is the person infectious?)
Individual *Intention* Variables
will-get-inoc?
will-go-to-workschool?
will-go-to-store?
will-go-to-doctor?

Example: members of a fatalistic society are less likely to believe in the efficacy of their own actions, and as a result are less likely to get inoculated. Figure 5 (next page) shows how these rules affect the decision of an individual agent in this system as to whether or not to get an inoculation.

The sequence of screenshots in Figure 6 (next page) shows the spread of the disease unfolding over time. The graph to the right of each screenshot shows the number of individuals in each category. Table 3 shows what each color represents.

Table 3. Screenshot Color Legend

Agent Colors		
White		Not yet exposed
Red		Visibly sick
Beige		Infected but not yet showing symptoms
Blue		Inoculated but not yet immune
Purple		Immune due to inoculation
Green		Cured after being sick (immune)
Background Colors		
Red		Store/public place
Blue		Hospital/doctor’s office
Yellow		School
Orange		Work
Grey		Homes

Figure 3 and Figure 7 below shows the differences in outcome when the community has a high fatalism variable value vs. a low fatalism value.

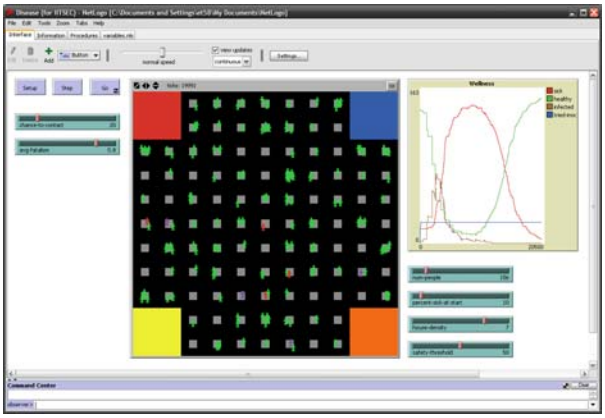


Figure 3. Screenshots showing high fatalism result

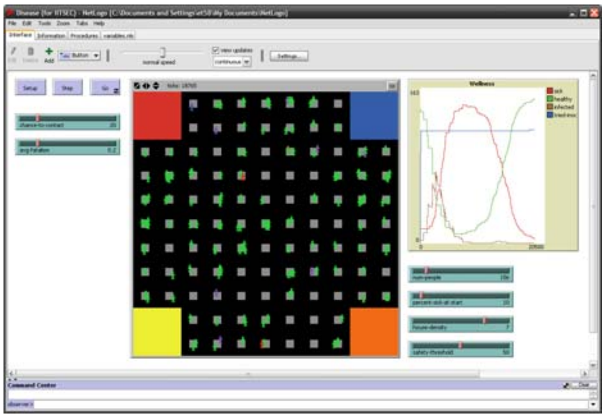


Figure 4. Low fatalism result

5. PHASE II SCENARIO

As part of the Phase II exploration we have built a prototype that models a small population of individuals going about their daily routines and responding to the societal stress introduced by an earthquake emergency. The model allows a user to explore the ways in which habits and behaviors affect each person’s injury level during and immediately following the earthquake. We have constructed a framework that allows for experimentation with the influences of a number of cultural and behavioral variables on each person’s preparedness and resilience. A user can change the value of the behavioral variables for individual agents while performing “what if” experiments to gain insight into the effects on the outcome. Different cultural and behavioral variables can be inserted into the model to experiment with their effects on each person’s wellbeing. This model instantiation includes a hypothetical multicultural population consisting of three different cultures.

Culture, for the purposes of this project, refers to the arts, customs, habits, beliefs, values, behaviors, or objects that constitute a people's way of life and the human interactions, rules, and processes that bind and

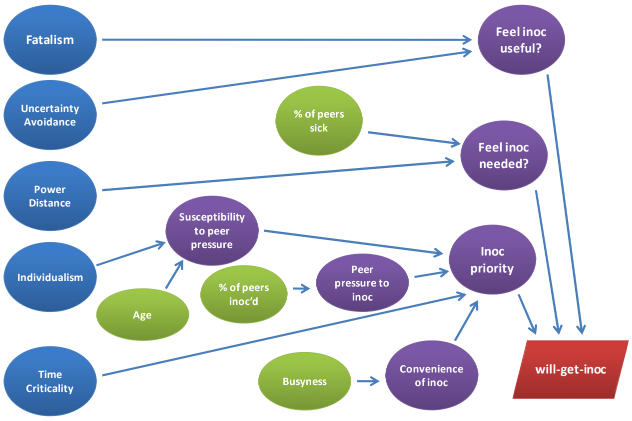
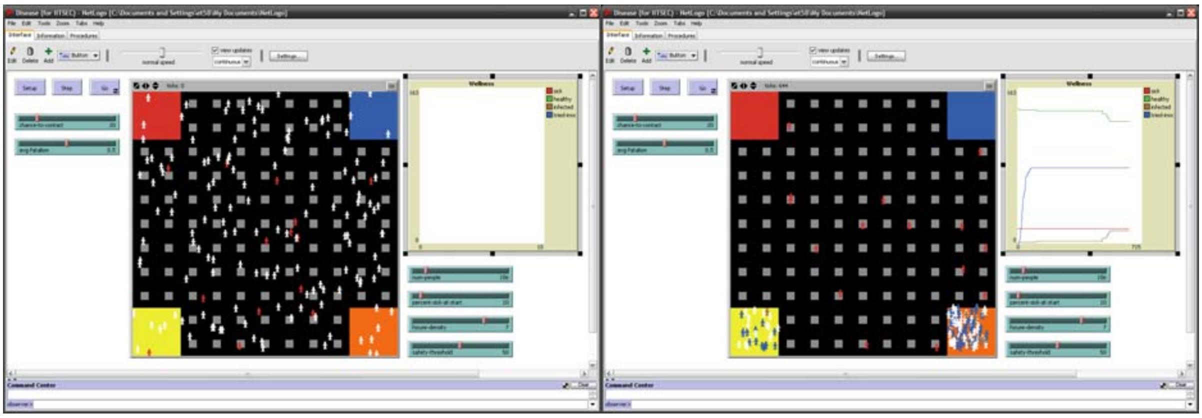
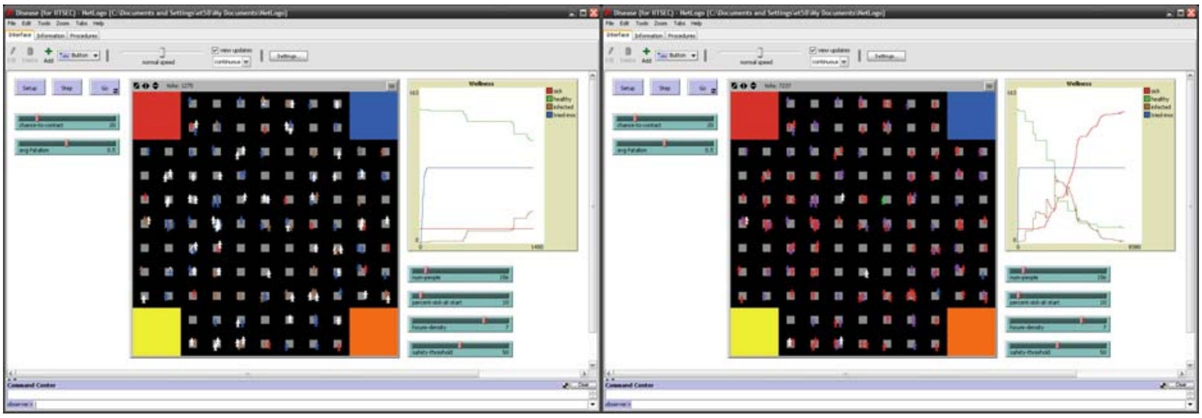


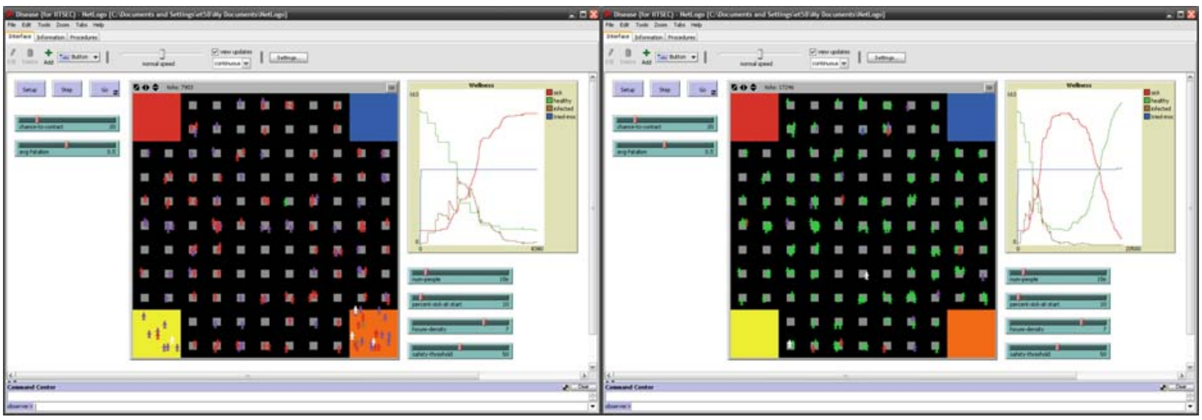
Figure 5. Agent Inoculation Decision



Simulation Start Day 1



Night 1 Night 5 (note sick people)



Day 6 (note presentees – sick people at work) Day 12 (mostly cured)

Figure 6. Screenshots of flu simulation execution

separate people as individuals or groups. The cultural variables considered as candidates for inclusion in our behavioral model come from literature (e.g., Hofstede, D'Andrade and Strauss).

Every model is a simplified representation of the real world. Models are typically constructed for the use of analysts wishing to gain a greater understanding of a situation and its influences. Tools and methodologies to support this will be most effective when they include representations of behavioral drivers that support the analyst's view of the population's cultural and psychological makeup. To build a representation of the external world, a modeler or analyst must choose which entities are most important to include in the model, based on information from subject matter experts, the intended use of the model, and the relative impact that each entity has on the system's behavior. This research focuses on the use of cultural variables and explores the choices of those variables and the implementation of their effects on agent behaviors.

5.1. Overview of Geert Hofstede's Cultural Dimensions

Geert Hofstede is a Dutch academic renowned for his research on national and organizational cultures. Hofstede's work has provided a framework for understanding cultural differences. He defines culture as the "collective mental programming" of a group, organization, or nation. It is this collective mind that distinguishes on group from another. In Hofstede's view, there are two aspects to culture. First, the internal aspects of a culture consist of the shared values and beliefs of a group; not easily discernible but crucial in understanding how human behavior is influenced by it. Second, the external aspects of a culture consist of a group's practices, language, and mythology.

Hofstede conducted a Values Survey Module and with its results he generated major themes or "dimensions" that described the characteristics of national cultures. Below are Hofstede's five cultural dimensions:

1. **Power Distance:** This characteristic of culture refers to the extent to which the less powerful person in a society readily accepts inequality in power and expects power to be distributed unequally. "All societies are unequal, but some are more unequal than others" (Hofstede 1980).
2. **Individualism:** This characteristic of culture refers to an individual's behavior towards the group. In an individualist society, "the social ties of individuals are loose and everyone is expected to look primarily after their own interests and their immediate families." In a collectivist society, "from birth individuals belong to close in-groups from which they cannot detach themselves. An individual is expected to give his/her in-group unquestioned loyalty but in exchange the group will protect the interests of each member.
3. **Masculinity:** This characteristic of culture refers to a society's proclivity to "use the biological existence of two sexes to define very different social roles for men and women." Masculine cultures socialize men to be assertive, ambitious, and competitive, and to strive for material success. Women in masculine cultures are expected "to care for the nonmaterial quality of life, for children, and for the weak." In contrast, men and women in feminine cultures have overlapping social roles.
4. **Uncertainty Avoidance:** This characteristic of culture refers to the extent to which people in a society "are uncomfortable with unstructured, unclear, or unpredictable situations and try to avoid these situations by adopting strict codes of behavior a belief in absolute truths." Societies characterized by high degree of uncertainty avoidance are aggressive; seek to be secure, and intolerant.
5. **Long-Term Orientation:** This characteristic of culture relates to the values of thrift and perseverance. In contrast, values associated with a short-term orientation, are a "respect for tradition, fulfillment of social obligations, and protecting one's face."

We include a number of related behavioral variables like gender, age, and busyness, which are unrelated to culture but still guide the behaviors of the simulated individuals (agents). These variables lead to activities that in the model impact each person's state of earthquake preparedness, e.g. becoming educated in earthquake safety or taking a proactive stance toward home preparedness.

With cultural variables and their expected effects as identified by psychologists and other behavioral experts, models of this type could be used to plan public health motivational campaigns and to identify the locations, schedules, and types of services that would most likely be effective in inhibiting the spread of the disease. This framework would also be useful for modeling population reactions to other types of crises and associated educational or public health approaches, emergency response activities, and government policies, in order to support human decision-making.

6. PHASE II APPROACH

This project uses a scenario-based approach, which provides a framework for analyzing the representation and reasoning functionality needed for the user to explore a given set of scenarios. The scenarios are textual descriptions of situations and include the context and environment in which they might exist. They are chosen to represent a portion of the space, which is typical of many of the problems that an analyst may want to explore. This includes, in this project, identification of a category of questions to be answered by models, cultural descriptions and psychological theories of interest to apply to these models, and environmental contexts, which will supply a cross-section of the issues that modelers may encounter. The scenarios will cover a variety of population behaviors and types, and a variety of environmental variables, so that the requirements extracted will support a selected space of situations.

This effort extracts metadata from cultural descriptions and psychological theories, where the extraction process is the first step in representing qualitative data in a computational or quantitative form. The modeling process begins with reasoning about the characteristics of the description or theory that needs to be included in models to drive the behavior of individuals or organizations and that will impact societal trends. This process includes a review of relevant psychological theories and cultural descriptions and a process for identifying components of this qualitative information that could be represented in a model. Literature studies were used to identify those components that typically have the most effect on the behavior of individuals, organizations, and societies. The extracted metadata has been used to represent multiple cultures and psychological theories.

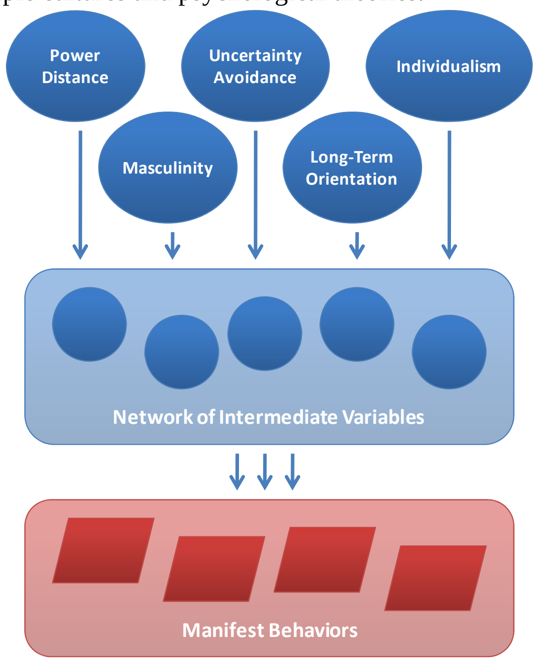


Figure 7. Relationship between high-level independent variables and low-level manifest behaviors (Phase II)

Once the appropriate independent and dependent cultural variables were extracted, a causal influence model was built to capture the graph of relationships between the high-level cultural variables (e.g. the five core Hofstede dimensions) and the low-level behaviors that are manifest within each person to affect his/her experience before, during, and after the earthquake. Figure 7 shows the basic relationship present in each simulated individual, while Figure 8 shows an example of the causal network that results in a person's decision to seek earthquake preparedness education.

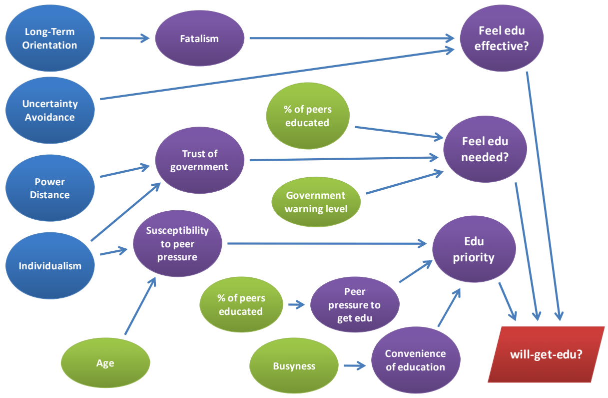


Figure 8. Casual influence network of variables affecting a decision about seeking earthquake education

7. EARTHQUAKE SCENARIO

For this scenario we will focus upon the inner East Bay area, specifically the city of Fremont. The inner East Bay area is generally urban, densely populated and ethnically diverse, with older building construction. As of 2009, Fremont's total population was 213,000 people grouped into 68,000 households. The demographic groups of the City of Fremont are:

- Non-Hispanic Whites 32.0%
- Asian-American 48% (mostly Afghan community)
- Hispanic/Latino 14%
- Other 6% (U.S Census Bureau)

In this scenario, a 7.5 magnitude earthquake occurs in the San Francisco Bay Area, with the epicenter located along the Hayward Fault zone. The Hayward Fault is a dangerously unstable fault that runs along the Eastern hillside of the San Francisco Bay Area, one of the most densely populated areas of the state. The Hayward Fault runs beneath homes, schools, hospitals, and the UC Berkeley campus.

The earthquake will disrupt normal community life and services. The vast majority of the city will lose electricity for the first 12 hours after the earthquake. Hospitals, senior centers, and civic places will maintain a minimal source of electricity via generators. Gas pipes are damaged as are telephone lines. In a few high-traffic intersections, natural gas lines rupture causing major structural fires, and damaged water lines complicate extinguishing these fires. Furthermore, a greater threat to the population's water supply is possible damage to the Bay Division Pipelines (BDPL). The City of Fremont is the point where the Hetch Hetchy Aqueduct, which provides water to San Francisco from the Tuolumne River (Yosemite National Park), splits into four separate Bay Division pipelines. All four of these pipelines run across the Hayward Fault.

The seismic shock would induce soil liquefaction which damages the foundation of some old buildings.

Additionally, "soft-story" buildings not retrofitted before the earthquake are particularly vulnerable. These are typically classic apartment buildings with a store or restaurant on the first floor or "tuck under" parking. It is not uncommon for soft-story buildings to house a substantial number of residents living in affordable or low-income units. Residents of these buildings will be displaced and most will be dependent on emergency housing.

Subsequent aftershocks: there are 70 with magnitudes higher than 3.0 in the immediate 24 hours following the earthquake; 30 more the second day after the earthquake.

The theoretical framework used in this earthquake simulation was built upon Geert Hofstede's Five Cultural Dimensions. In this report, an overview of the Five Cultural Dimensions and the cultural profiles used in this simulation are provided.

8. MODEL IMPLEMENTATION

The simulation created as part of this project is an agent-based model built within the freeware Java-based framework NetLogo. NetLogo supports rapid prototyping of multi-agent simulation systems with built-in GUI user controls, plus graphing, and visualization tools.

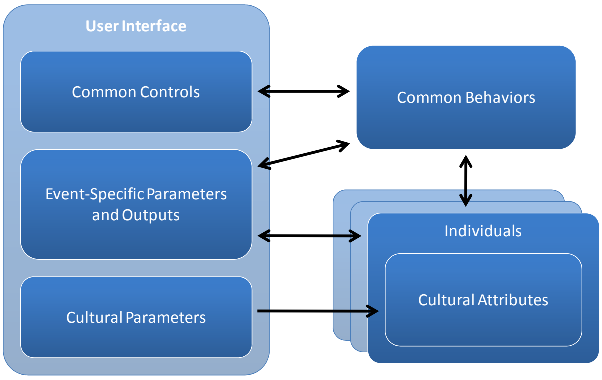


Figure 9. Basic modular NetLogo model architecture

The earthquake model itself is created using a modular architecture that supports the exchange of cultural and event elements (see Figure 9). It is an extension of the earlier flu-simulation work of this IRAD in which the non-flu-specific components have been generalized, with the earthquake elements substituted as an exchangeable module. Additionally, cultural function elements are separated into an exchangeable module, and the user-configurable frontend offers sliders that control the value of the various independent variables for up to three distinct simulated cultures. Figure 10 shows a screenshot of the user interface, which supports numerous experimental adjustments before and during execution.


Figure 10. Netlogo Model User Interface (Phase II)

9. CONCLUSION

In this project, we explored the representation of cultural and psychological attributes as agent variables that will guide the agent behavior and decision-making. In this project we focused on agent-based models of a community responding to a crisis. An important focus of this work was the representation of a crisis as a modular set of events with a context. We developed a design approach that will allow different crises to be inserted or overlaid upon the agent-based community to explore the community response to the crisis. These approaches can be used to develop agent-based models that could be used by crisis response and preparation planners to forecast the behaviors of a community under different sets of conditions. The planners could explore approaches to communication, crisis education, preparation instructions and instructions during a crisis and could then tailor interactions with the community in a way that is most effective for the multiple cultures involved.

Potential next steps for continuing this research include developing approaches to modeling various forms of communication so that planners could explore the impacts of e.g.,

- authoritative versus non-authoritative information sources,
- agent-to-agent versus broadcast distribution of information
- government sources versus cultural leaders.

In addition the model should be extended in the simulation time horizon to include post-disaster reconstruction. This would allow a modeling approach to support policy makers in evaluating the impact of cultural influences in reconstruction programs in the long-run.

Another important next step would be characterization of potential emergencies and crises in terms of representation by a set of features and the development of an overlay template that could more easily be used to represent and insert a new crisis into the agent community.

o

10. APPENDIX: SAMPLE CULTURAL PROFILES

These profiles were used to instantiate the core cultural variables before model execution.

10.1. Afghan Cultural Attributes

- Likely to hold a fatalistic orientation which may cause an individual to accept circumstances as "God's will (COP 2002).
- Highly distrusting of anyone who is not a kinsmen even of other Afghans. (CHANGE 2008: 50, Braakman 2005:30)
- Use of religion to spiritually reinterpret life events (Omeri, A., C. Lennings and L. Raymond. 2004)
- Familiar with Adversity (Omeri, Lennings, and Raymond 2004; SAFE 2003)
- High degree of resilience among 1st generation immigrants and elderly segment of Afghan community. This is due to the great amount of adversity they have experienced during lifetime. (CHANGE 2008: 51, Monsutti 2008: 1))
- Potential for some degree of Abseenteism (on Friday)) to attend Mosque and visit family. (Lipson and Omidian 1992: 272)
- Higher rate of "homeboundess" of women and elderly due to social isolation and lack of English speaking skills. (Lipson and Omidian 1992:272)
- Obsessed with cleanliness. "Concepts of purity and impurity are integrated into ideas of health and disease." (Lipson and Omidian 1992: 273)
- High level of suspicion of public authorities due to their negative experiences of the state and highly stratified society (CHANGE, 24)
- Hierarchical and patriarchal culture. Due to the fact that Afghan diplomas are rarely recognized outside of Afghanistan, many Afghan men have had to take "lower-class employment" and have to depend on their children who learn English quicker. These two factors have contributed to Afghan men's "loss of authority." Likewise, older generation find themselves more dependent on the younger generation (Braakman 30)
- High degree of collectivism. An individual is a representative of his/her family not an independent individual (Braakman 85)
- High incidence of social and religious conservativism. While not all Afghans were conservative during their time in Afghanistan, some liberal individuals in the Diaspora become "conservative" because this is closely associated with being "Afghan." (Braakman 86; Nawa 2001). However, some conservative Afghans shed this aspect of their identity in order to better integrate into their new environment.
- High degree of Social Conformity. Afghans gossip a lot. They do not want to appear "un-Afghan" or *azad* "free." The notion of social conformity among the Afghan Diaspora is closely linked to the preservation of an individual's reputation but ultimately the family name and honour (Braakman 2005:86). This reflects the "honour-shame" orientation of the Afghan culture.

10.2. Latino Cultural Attributes

- In general, Latinos place a high value on interpersonal relationships. *Respeto* refers to a quality of self that must be presented in all interpersonal relationships.6 It signifies attention to proper and moral behavior and indicates an expression of deference to the person one confronts (NC Latino Health 2003).
- Trust is an important cultural value tied closely with respect. Trust is built on mutual respect over time (NC Latino Health 2003)
- Latinos place a great deal of importance on the family as the primary social unit and source of support for individuals. Help and advice are usually sought from within the family system first, and important decisions are made as a group (NC Latino Health 2003). Some consider "familismo " is an extreme form of collectivism (Perilla, Norris, Lavizzo 2002).
- *Machismo*: Traditionally the Latino male has been acknowledged as the authority figure in the family (NC Latino Health 2003)
- View of life and death; the individual perceives little personal ability or responsibility for success or failure in life (external locus of control). The person feels that control over what has happened and what will happen has an external locus, and hence is wholly out of his or her hands (NC Latino Health 2003).
- Latinos are generally more concerned with the present than with the future. Priority is given to current activities rather than planning ahead. Thus, being late for an appointment is not due to lack of respect or reluctance, but to priority and concern over current activity or personal Interaction (NC Latino Health 2003).
- *Simpatia* calls for positive interpersonal relationships (Padilla 2002).

10.3. Anglo American Cultural Attributes

- Individualistic: Social ties between an individual and in-groups are much looser. An individual seeks his/her well-being and their nuclear family. Competition is regarded higher than cooperation. (Weaver 2001, Hofstede 1980)

- Egalitarian: Americans believe that all persons are equal in status and should be given an equal opportunity to better themselves. Furthermore, a person is not born with a high social status but rather an emphasis is placed on individual achievement (Weaver 2001). An individual's status is based on what s/he does not his/her family or heritage.
- Independent relationships. Due to the individualistic cultural attribute, Americans seek to be self-reliant. Extended family does not play role in major decision-making. (Hong, Ip, Chiu, Morris and Menon 2001).
- Time Orientation: Anglos value timeliness and planning ahead. Time is linear, sequential, absolute, and prompt (Redding 1980 cited in Kirkbride, Yang, Westwood 1991).
- Open and direct communication
- Personal Responsibility

11. BIBLIOGRAPHY

Afghanistan Culture. 2009. Available from: http://www.about-afghanistan.com/afghanistan-culture.html

Braakman, M., 2005. *Roots and Routes: Questions of Home, Belonging, and Return in an Afghan Diaspora*. M.A. Thesis. Leiden University. Available at: http://www.ag-afghanistan.de/files/braakman_roots_a_routes.pdf

Brannon, C., 2007. "Cultural Competency: Values, Traditions, and Effective Practice." *Nutrition Dimension*.

Hanifi, M. J., 2000. "Anthropology and Representation of Recent Migrations from Afghanistan," in *Rethinking Refuge and Displacement: Selected Papers on Refugees and Immigrants*. Volume VIII. American Anthropological Association: Arlington. Available from: http://www.farda.org/english/articles/j_hanifi/Refugee.pdf

Hofstede, G., 1980. "Motivation, Leadership, and Organization: Do American Theories Apply Abroad?" *Organizational Dynamics*.

Hofstede, G., 1984. "The Cultural Relativity of the Quality of Life Concept." *The Academy of Management Review*.

Hofstede, G., 1980. *Culture's Consequences*. London: Sage Publications.

Lipson, J.G., and Omidian, P. A., 1992. "Health Issues of Afghan Refugees in California," in *Cross-Cultural Medicine- A Decade Later (Special Issue) West J Med*, Sep 157:271-275. Available from: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1011275/pdf/westjmed00085-0061.pdf

Monsutti, A., 2008. "Afghan Migratory Strategies and the Three Solutions to the Refugee Problem," *Refugee Survey Quarterly*. Available from: http://graduateinstitute.ch/webdav/site/executive_education/shared/executive_education/summer_international-affairs_faculty-IA_professors/A%5B1%5D.MONSUTTI-RSQ-AM.pdf

Morioka-Douglas, N., Sacks, T. and Yeo, G., 2004. "Issues in caring for Afghan American elders: Insights from literature and a focus group," *Journal of Cross-Cultural Gerontology* 19: 27-40.

Nawa, F., 2001. "Out of Bounds: Afghan Couples in the United States: A Study of Shifting Gender and Identity." Aftaabzad Publications: San Francisco. Available from: http://www.faribanawa.com/outofbounds.pdf

NC Latino Health. 2003. "Cultural Factors and the Health of North Carolina Latinos."

Office of Minority Health of Rhode Island Department of Health. "Latino Culture and Health." Available at: http://www.health.ri.gov/chic/minority/lat_cul.php

Omeri, A., Lennings, C. and Raymond, L., 2004. "Hardiness and Transformational Coping in Asylum Seekers: The Afghan Experience" *Diversity in Health and Social Care* 1:21-30.

Perilla, J., Norris, F., and Lavizzo, E., 2002. "Ethnicity, Culture, and Disaster Response: Identifying and Explaining Ethnic Differences in PTSD Six Months after Hurricane Andrew." *Journal of Social and Clinical Psychology*, Vol. 21, No.1, pp20-45.

Rubel, A. J., 1960. "Concepts of Disease in Mexican-American Culture." *American Anthropologist* Vol. 62, No.5, pp. 795-814. Available from: http://www.jstor.org/stable/667103

Sadat, M. H., 2008. "Hyphenating Afghaniyat (Afghan-ness) in the Afghan Diaspora," *Journal of Muslim Minority Affairs* Vol. 28, No.3.

SAFE (The Sabawoon Afghan Family Education and Counseling Centre). 2003. "Exploring the Mental Health Needs of Afghans in Toronto." Available from: http://ceris.metropolis.net/Virtual%20Library/EResources/SAFE2003.pdf

The Cultural Orientation Project (COP). 2002. "Afghans-Their History and Culture," Available from: http://www.cal.org/co/afghan/aval.html

U.S. Census Bureau. 2006. "Fremont Ethnicity." Available from: http://www.fremont.gov/DocumentView.aspx?DID=843

United Kingdom (Department for Communities and Local Government). 2009. "The Afghan Muslim Community in England: Understanding Muslim Ethnic Communities," Published by The CHANGE Institute. Available from: http://www.communities.gov.uk/documents/communities/pdf/1203127.pdf

Weaver, G., 2001. "American Cultural Values." Kokusai Bunka Kenshu (Intercultural Training), Special Edition, 1999, pp.9-15.

A CANADIAN APPROACH TOWARDS NETWORK ENABLED CAPABILITIES – I: SIMULATION VALIDATION & ILLUSTRATIVE EXAMPLES

Richard McCourt[a], Kevin Ng[a][b], Roy Mitchell[a]

[a]Centre for Operational Research and Development, Defence Research and Development Canada
[b]Telfer School of Business, University of Ottawa

richard.mccourt@drdc-rddc.gc.ca, kevin.ng@drdc-rddc.gc.ca, roy.mitchell@drdc-rddc.gc.ca

ABSTRACT
This paper presents an approach to understanding networked enabled operations using agent-based simulations. We describe the newly created agent-based software ABSNEC, highlighting some of its salient features: the ability to represent human factors towards the analysis of battle outcomes in network operations; and the ability to represent realistic force structures with tiered C2 architectures. We provide affirmative results of three validation techniques to date on the model. Finally, we demonstrate the utilization of ABSNEC to acquire meaningful insights for analysis through two examples: a study on the interrelationship between fratricide, human factors and situation awareness; and generation of alternative combat strategies for a military engagement.

Keywords: agent-based model, network enabled capabilities/operations, human factors, military operations research, model validation

1. INTRODUCTION

Network Enabled Operations (NEO) and Network Enabled Capabilities (NEC) are increasingly being recognized as critical enablers of military capability in the 21st Century and essential for military transformation. However, we are still unable to define their full potential. Although NEO/NEC concepts have enormous potential to transform and improve defence capabilities, there are risks associated with these issues. NEO/NEC depends heavily on technology but may be vulnerable to asymmetric attack. There are also concerns about the ability of the Canadian Forces to interoperate and integrate with allies who have greater or lesser sophistication with respect to NEO/NEC. To turn potential into reality, we have to assess, explore and understand the impact of NEO/NEC.

In a network centric system, humans are often required to extract, interpret and validate (machine) information from raw data at one or more points along the path from source to users. However, information is not the same as raw data. Often, raw data must be understood and interpreted by humans to produce information. It is well known that networks are the most efficient means devised to date for distributing en-masse large volumes of data. However, any networked system can, and also will, distribute erroneous data just as efficiently as valid data. Regardless of whether it is incorrect human interpretation, simple typographical errors, or faults and limitations in sensors, defective data can be quickly distributed across any war-fighting system. In summary, humans can produce large-scale damage effects quickly in a networked environment. Numerous case studies of this problem exist. These include for example the USS Vincennes incident, the Kosovo tractor bombing incident, and the more recent bombing of Canadian troops in Afghanistan. It is essential to stress that in each of these networked operations, multiple human errors, compounded by sensor limitations contributed to tragedy.

Time has always been of critical importance in military networked operations and combat. In a typical discussion of Command and Control, it is taken as axiomatic that the information presented to the commander must be timely as well as accurate and complete. Little or nothing is said about how timely is timely enough; nor is any yardstick given by which to measure timeliness. We propose to develop a measure of performance based on the timeliness and quality of information under the influence of human error (the human-in-the-loop effect). This NEO/NEC performance metric has the potential to establish a baseline for comparing future networked operations. Such a metric could redefine the rules of engagement in networked operations or combat and may provide a decision tool enabling CF and NATO allies to recognize and exploit opportunities to integrate sensors, weapons, and platforms in optimal NEO/NEC architectures to achieve greater value from future capital investments. Finally, it may be able to improve force effectiveness, decrease combat casualties due to enemy actions, and to decrease confusion-related friendly fires.

Our research involves using analytical modeling and agent-based simulation to quantify and assess the impact of timeliness and quality of information towards NEO/NEC. Currently, there exist a number of battlefield-specific ABM platforms, such as Map Aware Non-uniform Automata (MANA) (McIntosh et al 2007), ISAAC/EINSTein (Ilachinski 2000), WISDOM II (Yang et al 2006) and BactoWars (Millikan et al

1996). A set of commonly used ABMs were recently reviewed by Railsback et al (2006). Our recent contribution, ABSNEC, models after the well-known MANA, developed by the New Zealand Defence Technology Agency. MANA has been widely used in the international defence science community and is acknowledged to be one of the leading agent-based distillation combat models. ABSNEC has a high degree of similarity to MANA, but its design is much better suited to the implementation of the new capabilities required by the Canadian research projects.

Among ABM models, ABSNEC is designed to effectively model networking and human factors. According to a recent survey for the NATO MSG-088 report on Data Farming in Support of NATO (to be published), ABSNEC remains to be the only system that is designed for networked operations studies that can efficiently track multiple intangible human factors parameters. It balances powerful features against the need for transparency, simplicity and execution speed. It adheres to the design philosophy pioneered in the Project Albert series of workshops in that it uses relatively simple representations of physical systems (distillation modeling), with an emphasis on the behavior and interaction of the entities within the model (agents). A list of some particularly important features of ABSNEC is as follows:

1. Detailed network characteristic modeling capability, such as latency and bandwidth, built into the model;
2. Ability to create custom algorithms that define network agents that control routing and capacity assignment;
3. Ability to represent human factors such as stress, fear, and other human factors towards the analysis of battle outcomes in network operations;
4. Ability to define custom agent state triggers with a simple graphical user interface; and
5. Ability to represent realistic force structures with tiered C2 architectures.

The above features are explained in detail in the ABSNEC users' manual. This paper will address the validation to date conducted on the model and will explore two illustrative case studies highlighting the unique features of ABSNEC.

2. MODEL VALIDATION

A simulation model is an abstract representation of a physical system and intended to enhance our ability to understand, predict, or control the behaviour of the system. As such, the simplification and assumptions will introduce inaccuracies to the simulation model. An important task is to determine how accurate a simulation model is with respect to the real system. The main difficulty remains to be there is no a universal approach for the validation. Balci (1998) presents 75 validation, verification, and testing techniques that are largely used in validating the models of engineering and business processes. The ABSNEC validation approach to date involves using 3 different techniques – face validation, model-to-model comparison and simple statistical analysis/test.

2.1. Face Validity

Face validity is asking the subject matter experts (SME) whether the model behaves reasonably and makes subjective judgments on whether a model is sufficiently accurate (Balci, 1998).

We consider the Ben Hasty scenario (Horne 2011) originated by the U.S. Naval Postgraduate School, for which 50 Red agents oppose 125 Blue agents. An additional 25 Blue agents are on their way to give Blue a 3:1 size ratio with Red prior to attacking. Blue could attack now (with only 125 agents) and take Red by surprise. However, the question addressed in the scenario was:

What would happen if Blue delayed the attack and waited for reinforcements?

Subject matter experts (SMEs) were consulted to describe tactics that Red could use to improve their outcome in a battle with Blue. The SMEs identified three tactics for Red: fortified defence, obstacle placement, and use of a spoiling force. In the fortified defence, Red clusters tightly together and waits for Blue to attack. In the obstacle placement tactic, Red uses obstacles to force Blue to attack through narrow choke points. And, for the final tactic, Red uses a spoiling force to immediately engage Blue. The spoiling force continues to attack Blue as they retreat towards the remainder of the Red agents that are waiting in a fortified defensive position. ABSNEC is used here as the agent-based platform to sample a large possibility of Red tactics and to data farm possible outlier results where Red is able to defeat Blue. What is of key importance in the face-validation process is whether any outlier results can reproduce the SME's recommendations.

In the scenario considered here, a Blue force of 150 soldiers is assembled and sets out to reach a goal within Red territory. The smaller Red force (50 soldiers) tries to defend this home location. All soldiers (both Blue and Red) have a sensor range of 1km and a weapon range of 1km. The sensor has a 100% probability of detection and the weapon has a 40% probability of kill.

Agents in ABSNEC move based on affinity forces to various targets. These affinities can be integer values from -10 (strong repulsion) to +10 (strong attraction), and can be uniquely defined for different states of the agent. In the Ben Hasty scenario, each Blue agent has two states: *Advance* and *Attack*. In the *Advance* state, Blue agents move towards their waypoint goal in Red's territory with a waypoint affinity of +10. When a Blue agent detects a Red agent, it enters the *Attack* state where it maintains the waypoint affinity of +10, and adds to it an enemy affinity of -5. Each Red agent also

Table 1 - NOLH sampling points for Red agents

	Advance			Defend		
	Wp	Fr	En	Wp	Fr	En
1	-4	10	6	-5	9	-3
2	-9	-5	8	-10	-4	1
3	-8	-1	-9	3	6	-5
4	-6	3	-4	1	-8	10
5	5	9	-1	-4	-10	-8
6	10	-4	-3	-9	5	6
7	3	-6	10	8	-1	-4
8	1	8	5	6	3	9
9	0	0	0	0	0	0
10	4	-10	-6	5	-9	3
11	9	5	-8	10	4	-1
12	8	1	9	-3	-6	5
13	6	-3	4	-1	8	-10
14	-5	-9	1	4	10	8
15	-10	4	3	9	-5	-6
16	-3	6	-10	-8	1	4
17	-1	-8	-5	-6	-3	-9

has two states: *Advance* and *Defend*. If Red agents wait in their starting location in the *Advance* state, and have an enemy affinity of +5 when an enemy is detected in the *Defend* state, then *all* Red agents will be killed, taking out an approximately equal portion of Blue agents before being wiped out. The goal of the scenario is to "farm" the space of possible Red affinities for these two states and look for an emergent behaviour that is beneficial to Red. Three possible target affinities (Enemies, Friends, and a Waypoint at Red home/Blue goal) can be varied for two different states (*Advance* and *Defend*) creating a 6-dimensional space to be explored. A Nearly Orthogonal Latin Hypercube (NOLH) sampling technique (Cioppa and Lucas, 2002) was used to limit the possible combinations of affinity choices and reduce the number of computations.

The sample points in the NOLH are listed in Table 1. It should be remarked that the space-filling and nearly orthogonal properties of the chosen sampling points have been independently verified. The mean number of kills for Blue versus the mean number of kills for Red is plotted in Figure 1 with the corresponding sample numbers. Ellipses show the

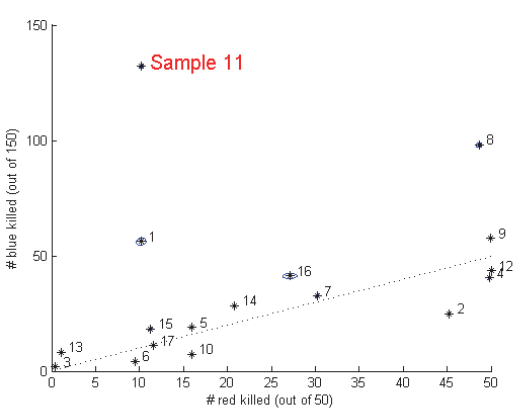


Figure 1 - Results for NOLH sample points

corresponding standard error of the mean about each point. The dotted line represents a one-to-one kill ratio. Points above this line correspond to tactics where Red is more efficient than Blue at killing. Samples in the top left corner of the figure are ideal for Red, i.e. large amount of Blue killed with low amount of Red killed.

Sample 11 is an outlier in upper left quadrant, and it is a beneficial solution for Red. The screen plot/animation results for Sample 11 (see Figure 2) reveals that the Red agents cluster together, back into the Red home location, and fight Blue. If Blue continues to push through, Red has a negative affinity to the Blue agents and will run away.

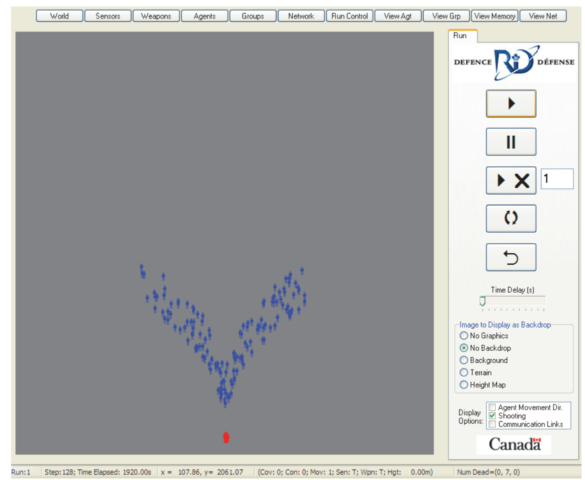


Figure 2 - ABSNEC screenshot for sample 11 of the Ben Hasty scenario

The behaviour of Red in Sample 11 exemplifies the fortified defence tactic identified by SME as a way to improve Red's battlefield outcome. The strong agreement between the ABSNEC simulation output and SME recommendation therefore brings us one step closer to accepting ABSNEC as a credible simulation solution.

2.2. Model-to-Model Comparison

Using model-to-model comparison (Balci, 1998), also known as docking or back-to-back testing, we compare various results of ABSNEC to results of MANA on the Ben Hasty scenario. Real world phenomenon can be represented by different conceptual models, and different research groups or individuals can implement conceptual models differently using a variety of programming languages or different simulation toolkits. These computational models may also be run on different platforms. The intent of this validation test is to investigate whether different simulations using similar input data produce similar results, trends and agent behaviours. One should be cautioned not to expect the simulation results to be identical, since agent-based simulation is built on simple probability and cellular automata principles. The essence of the model comparison exercise is to examine, upon applying data farming techniques on Ben Hasty

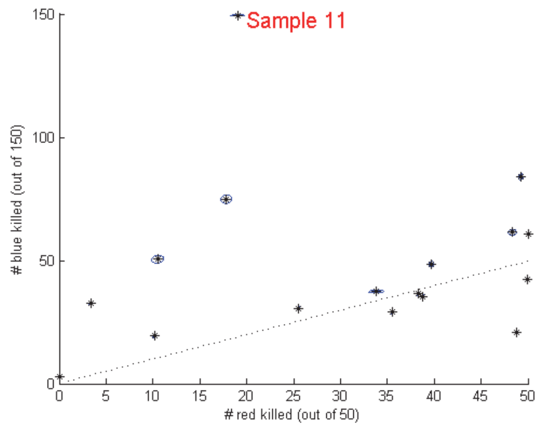


Figure 3 - Ben Hasty scenario results using MANA

scenario using ABSNEC and MANA, whether or not the simulations produce a similar trend and kill pattern. Most importantly, can the different platforms identify similar outliers for Red? Figure 3 highlights the Ben Hasty results at the same NOLH sample points using MANA.

Comparing Figure 3 with ABSNEC output Figure 1, it is gratifying to notice that they both exhibit similar trends for the number of Red and Blue killed. Of particular interest is that the two platforms simultaneously identify the same sample outlier - SAMPLE 11, albeit the differences in Blue and Red killed.

2.3. Comparison with Lanchester Equations

Finally, we compare the output data of ABSNEC with the output data of the classical Lanchester equation (Lanchester, 1956). The Lanchester model has been the fundamental model for developing theories of combat and for calculating attrition of forces in military engagements. The governing differential equations are subject to fairly stringent assumptions (Przemieniecki, 2000). For example, both forces are homogeneous and are continually engaged in combat; each unit or individual weapon is within the maximum weapon range of all the opposing units; and the effective firing rates are independent of the opposing force level. As a result, there is no shortage of criticism on the Lanchester assumptions and on the accuracy of the predicted force strength for any real military engagement (Helmbold, 1994). Nonetheless, the Iwo Jima Battle provides a case where the Lanchester model does provide an excellent agreement between the Lanchester predicted results and the actual American troop strength (Engel, 1954). The success is attributed to the fact that the governing parameters in the Iwo Jima scenario are fairly consistent with the assumptions in the Lanchester equations. In view of this, in this final validation test, we will introduce identical Lanchester assumptions in ABSNEC simulation. The generalized Lanchester equations incorporating C4ISR efficiency (Ng, 2006) are given as follows:

$$\frac{du}{dt} = \frac{-\beta v u}{a - f(a-u)} \qquad (1)$$

$$\frac{dv}{dt} = \frac{-\lambda u v}{b - e(b-v)} \qquad (2)$$

where u, v are the number of surviving units in each force at a time after the battle begins; λ, β are the corresponding kill probabilities; a, b are the initial values of u and v, respectively; e, f are the C4ISR efficiency of each force. For $e = f = 1$, these equations reduce to the direct fire scenario. For $e = f = 0$, these equations reduce to the area fire scenario.

To track force strength between the Lanchester and ABSNEC output, we need to understand how the two models represent area fire. In the Lanchester model, area fire represents the case where target information is not updated, and targeting weapons cannot tell if they are firing on live or dead target, i.e. C4ISR efficiency of zero. In ABSNEC, weapons are not related to the C4ISR efficiency of the force. Therefore, to simulate area fire Lanchester equations in ABSNEC, agents in each force do not die when hit by an enemy weapon, but lose the ability to fire back. Essentially, each force is given an initial picture of the enemy forces positions, but is unable to update that initial picture to show when an enemy is killed. To simulate different levels of C4ISR efficiency, agents in the simulation will die when hit by a weapon with a probability equal to the C4ISR efficiency. Therefore, agents that are hit with probability λ, β and die with e, f and are no longer targeted. This is the desired effect of C4ISR efficiency.

To summarize, in the direct fire scenario, an agent hit by the opponent weapon will be killed, whereas in the area fire scenario, the same agent represented is considered dead and its weapon is removed, but remains alive as a target for the enemy force. In the 25% C4ISR efficiency case ($e = f = 0.25$), if an agent is hit by opponent weapon there is a 25% chance that the agent will die, and a 75% chance that the agent will change to a weaponless state and remain a target. Table 2 summarizes the comparison of ABSNEC and Lanchester model results for a scenario with 50 Blue agents against 20 Red agents that was run 1000 times. The force size for both Red and Blue was averaged over all runs for each simulation time step, and the difference

Table 2 - ABSNEC/Lanchester comparison results

	Average Absolute Difference (Blue)	Average Absolute Difference (Red)	Average difference at final time (Blue / Red)
Direct fire	0.171	0.121	0.213 / 0.000
50% C4ISR efficiency	0.290	0.306	0.479 / 0.007
Area fire	0.156	0.021	0.171 / 0.006

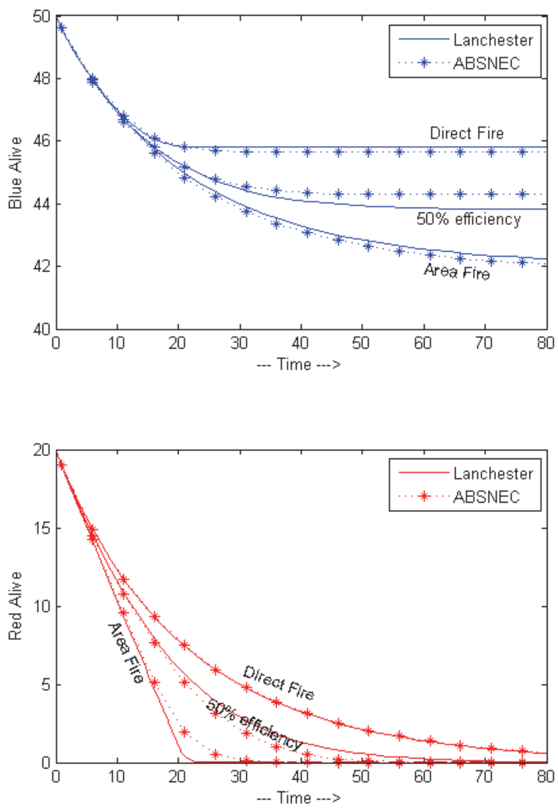


Figure 4 - Comparison of Lanchester's equations with ABSNEC

between this average and the solution to the Lanchester equation was used to calculate the average absolute difference for each side. Also shown, is the end state (steady state) difference in number of agents for each side. The differences between the two models are very small (much less than a single agent). Figure 4 is provided as a visual comparison of these results.

In summary, the affirmative results of the three validation techniques: face validation (using validation by SMEs), model-to-model comparison (ABSNEC versus MANA) and simple statistical analysis/test against the well known Lanchester equations provide clear evidence on the validation of ABSNEC to date. Furthermore, it instills an underlying confidence in the fidelity of ABSNEC in generating meaningful insights to complement complex military decision-making.

3. EXAMPLES

3.1. Example 1: Fratricide versus Situation Awareness

The objective of this example is to study networked operations with humans in the loop. The unique feature of ABSNEC – the ability to track intangible parameters such as morale, fatigue and combat stress – can be fully exploited to learn about why and how friendly fire happens and hopefully to prevent similar incidents in the future. Friendly fire casualties can have direct impact on troop morale, mission success, and public perception. A number of things can lead to friendly fire. One of the most common is miscommunication, which can result in unclear orders or lack of knowledge about troop movements. When allied troops are added to the mixture, maintaining lines of communication can be even more difficult, especially if language barriers and differing rules of engagement are being surmounted. Poor weather conditions and combat stress can also lead to a friendly fire incident in which a soldier mistakenly believes that he or she is shooting at the enemy. When a leader issues unclear or ambiguous orders, this can also be problematic when combined with conditions which prevent soldiers from using their own judgment. Incidents of friendly fire abound, however, it is extremely difficult to collect/gather relevant statistics for analysis.

Our fictitious scenario is based on the incident reported by the United States Department of the Army (2007) with modifications. An aerial gunship (e.g. the AH-64 Apache) has mistakenly identified a neutral target as an enemy. The gunship crew waits for additional information to confirm (or refute) the target, but the surrounding infrastructure concealing the target, the physical location of surveillance assets, and the structure and reliability of the available communication network can delay this confirmation/refutation. The gunship crew believes that this target is a threat, and they are in prime position to strike, so while they wait for additional information, the gunner's finger is on the trigger and the crew's stress level begins to rise rapidly. Beyond some threshold, the gunship crew will no longer feel justified in waiting for a target confirmation and choose to kill the target. The commander in charge is capable of ordering the gunship to stand down, but will not make that decision until a sufficient level of situational awareness is obtained. The commander receives discrete pieces of the situational awareness map from its two surveillance assets: a UAV (air surveillance asset) and a Humvee (ground surveillance asset). The commander also requires very up-to-the-minute information with a certain degree of synchronicity between the two sources. Beyond a lower threshold level of situational awareness, the commander will send an intermediate command to the gunship. The intermediate command will ease the stress of the gunship crew, but will only provide a temporary solution. Without further intervention, the gunship crew's stress level will still rise beyond the threshold level, causing them to kill the target. Once the commander has a complete situational awareness map, a "stand down" order is sent to the gunship, and if it is received prior to the gunship crew reaching their threshold stress level, then the neutral target will be saved. A visual description of the scenario is summarized in Figure 5.

In general, value of information is comprised of two main attributes: timeliness and quality of information. Timeliness is the degree to which mission performance depends on timely and perhaps perishable information (Perry 2005). Quality of information refers to the completeness and accuracy of information. That is,

$$\begin{aligned} \begin{matrix} Value\ of \\ Information \end{matrix} &= \begin{matrix} Joint\ Probability\ Function\ of \\ Timeliness\ \&\ Quality\ of \\ Information \end{matrix} \\ &= f(Timeliness, Quality) \end{aligned}$$

For this scenario, this concept simplifies to the following expression:

Let T = event where timely information is received prior to threshold stress level is reached,
Q = event where information is received prior to threshold stress level is reached is both complete and accurate

$$P(useful\ information) = P(T \cap Q) \qquad (3)$$

In our context, fratricide will occur whenever no useful information is received prior to gunship crew's stress threshold value is reached.

$$\begin{aligned} P(fratricide) &= 1 - P(useful\ information) \\ &= 1 - P(T \cap Q) \qquad (4) \end{aligned}$$

Since useful information refers to the degree of successfully correcting the situation awareness map, lowering the stress level of gunship crew and stopping the attack, the remainder of this example will demonstrate using ABSNEC to compute the probability of successfully correcting the SA map such that the stress threshold level will not be reached.

In this fictitious scenario, it is assumed that the gunship agent's stress level starts to rise at an arbitrary 2 units per second. Also, at the start of the scenario, the UAV and Humvee are in the area and begin sending their SA to the Commander. The Commander then makes decisions based on the percentage of the total picture received from both surveillance assets. The sensor at the gunship sees a single target (that it has mistakenly identified as an enemy); however, the UAV and Humvee each see five separate targets and will attempt to send a single packet through the network for each target. When these packets arrive at the Commander, they populate the Commander's SA map, but are obsolete after they reach a certain age. The UAV and Humvee continue to send each of these five packets at regular intervals, but some of the packets might be lost in transfer.

Once the Commander's SA map has received 6 of the total 10 packets concerning the misidentified neutral target, the *Hold* command will be sent to the gunship. (It is to be recalled that of the 10 packets, 5 are from the UAV, and 5 are from the Humvee). When the *Hold* command is received, the gunship's stress level will only increase at 1 unit per second. Out of the commander's SA map of 10 packets, whenever 8 of them on the neutral target have been received, the *Stand Down* command is sent and the gunship's stress level will stop increasing. If the *Stand Down* order is received before the gunship reaches its threshold level of stress, then the gunship will not fire its weapon, and the neutral target will be saved.

In this simple example, the SA map will be enriched more quickly if fewer packets are being dropped in the communication link. That is, useful content of the SA map is inversely proportional to probability of packets being dropped in link. To a first order approximation then, stress versus the enrichment of the SA map is analogous to the behavior of stress versus the probability of dropped packets.

The described scenario was run using ABSNEC for different probabilities of packets being dropped in the communication links between the surveillance assets and the commander. For each probability of a dropped packet, the scenario was run for 1000 replications and the average success of the mission was recorded. Figure 6 shows the probability that the neutral target was killed (i.e. fratricide) against the probability that a

Figure 5 - "Fratricide and Intangible Parameters" scenario layout

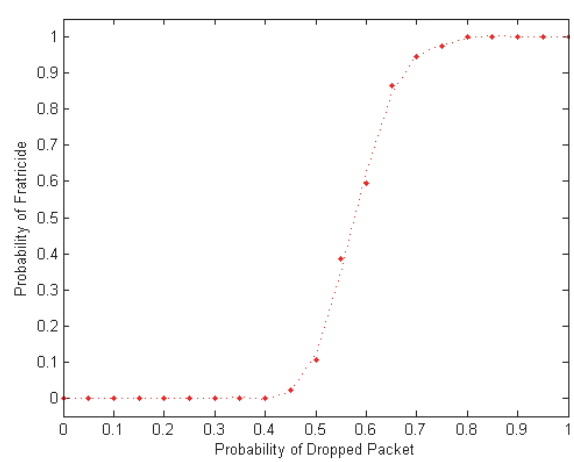


Figure 6 - Probability of fratricide when packets are dropped in the network

packet was lost.

Letting x = probability of dropped packets, Figure 6 reveals the following

$$P_{fratricide}(x) \begin{cases} = 0 & if\ x\ \leq 0.4, \\ > 0 & if\ x\ > 0.4. \end{cases} \qquad (5)$$

The chance of fratricide can thus be lessened if one can use technology to reduce the packet drop rate between communication links, a recommendation that hardly comes as a surprise for this example.

A salient feature of ABSNEC is that it can track up to 6 intangible parameters. Using this feature, a more realistic scenario can be explored in which the gunship crew experiences both stress and an immense fear of being shot at by the wrongfully identified enemy. A simple screenshot from ABSNEC in Figure 7 illustrates how two human factors (stress and fear) can be used to trigger an alternate state for the agent when they reach a specified level.

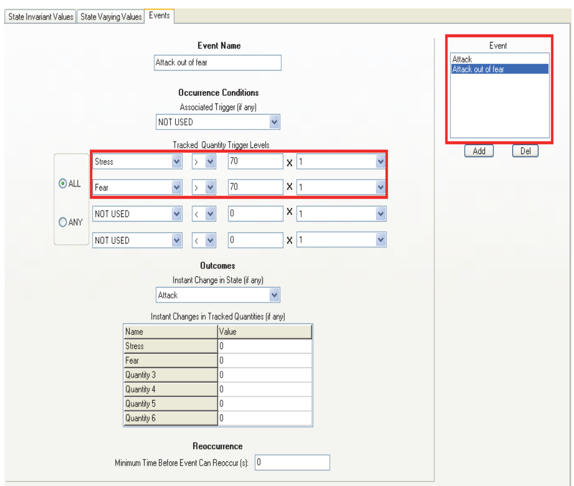


Figure 7 - Screenshot from agent events tab; used to trigger agents to a new state.

Previously, the gunship would fire on the target if its stress levels rose above a set threshold (arbitrarily defined at 120). In the revised scenario, a secondary condition was set where the gunship would now fire if both stress and fear levels rose beyond a lower threshold of 70. Both of these events trigger the gunship into the *Attack* state, where it proceeds to kill the target.

For illustrative purposes, fear levels rise at 1 unit per second. When the *Hold* order is received from the commander, the gunship agent's fear then rises at 2 units per second. When either *i)* fear and stress both rise beyond 70, or *ii)* fear is less than 70 and stress is above 120, then the gunship will enter the *Attack* state and kill the target. An example of fear and stress levels is shown in Figure 8. At 21 seconds, the *Hold* order is received and both stress and fear rates change. At 47 seconds, fear rises beyond 70, and at 56 seconds stress

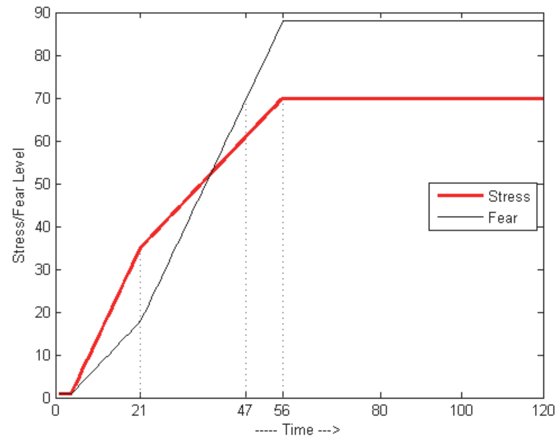


Figure 8 - Sample stress and fear levels

hits 70, which triggers the *Attack* state and ends the simulation.

Figure 9 shows the summary of results for the revised scenario. To achieve the same mission success as the previous case, the probability of a dropped packet in the communication links must be lowered due to the creation of the additional fear trigger. In other words, for the same probability of dropped packet, there is now a higher probability of fratricide. (It was verified from the simulation output data that both event triggers were activated throughout the simulation, i.e. sometimes stress levels triggered an attack with low fear levels, and sometimes elevated fear levels caused the gunship to attack at lower stress levels.)

Even though the example is fictitious, it provides us a means to understand the relationship between fratricide, stress, fear and situational awareness. Similarly, our approach can easily be adapted to study other aspects of networked operations with humans in the loop. Last, but not the least, our approach provides us a means to quantify the effect of training on combat outcomes. It has been acknowledged that training might reduce fear among military operators. Reduced fear would result in a less steep (more flattened) fear curve in Figure 8, which in turn would shift of the

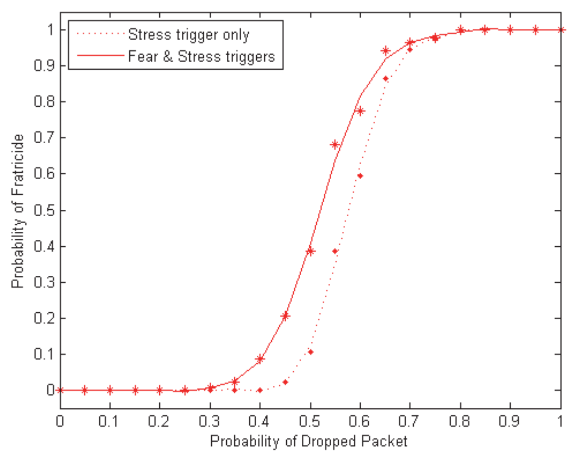


Figure 9 - Effect of additional triggers causing the gunship to fire

Probability of Fratricide curve (Figure 9) to the right, reducing the chance of fratricide.

3.2. Example 2: Ben Hasty revisited

The aim of this example is to demonstrate that by combining data farming techniques and the distinctive feature of ABSNEC, i.e. the ability to represent realistic force structures with tiered C2 architectures, we can generate valuable, and perhaps unpredictable, combat strategies. Moreover, the simplicity of ABSNEC provides a viable method of performing quick turnaround studies to NEO/NEC scenarios.

The Ben Hasty scenario as was presented in the Validation Section, ended with the demonstration that the Red tactic identified using ABSNEC was one of the recommendations by the SMEs. In this example, ABSNEC was used to repeat the data farming process for Blue and search for beneficial tactics that would help Blue defeat the improved Red tactics demonstrated in Sample 11. To do this, Red affinities found in Sample 11 were held constant while the Blue parameter space of affinities, intermediate waypoints, and issued commands are explored. The initial setup of the scenario is shown in Figure 10. The Blue force is divided into two teams: Blue1 and Blue2. Each team starts in the Advance state and moves towards its team's intermediate waypoint. When a Blue agent reaches its intermediate waypoint, it changes to the Flank state and continues to move towards the final goal. If a Red agent is detected, the Blue agent that detected it changes into the Attack state. Each Blue team maintains a situation awareness (SA) map. This SA map is sent to the Blue Leader at a constant rate with a fixed delay of 150 seconds (arbitrarily chosen). The Blue Leader combines the two SA maps and uses the ratio of total friends to total enemies to make command decisions. These command decisions are then sent to both of the Blue teams with a fixed communication delay of 150 seconds.

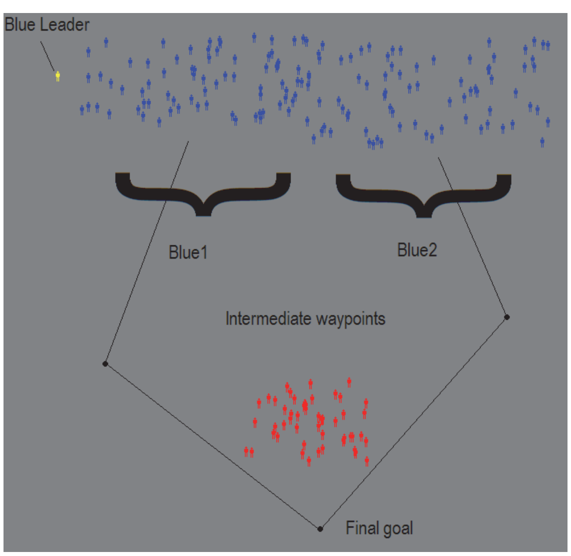


Figure 10 - ABSNEC screenshot (with labels) for example 2

Sixteen agent affinities, four positional variables, and four command variables, were used in the data farming procedure to find a suitable strategy for Blue to counterattack Red. These 24 variables are listed below, along with their associated ranges. As stated earlier, agent affinities can be integer values from -10 (strong repulsive) to +10 (strong attractive). Waypoint affinities have been restricted to only attractive forces to ensure agents do not simply run away from their final goal.

<u>Blue States and Affinities</u>

Advance

- Next waypoint…………………….........[1, 10]
- Others in own group ….…...……......... [-10, 10]
- Friendly group …………….................. [-10, 10]
- Target assigned by commander.......... [-10, 10]

Flank (entered when intermediate waypoint is reached)

- Next waypoint..………….…………….. [0, 10]
- Others in own group ………….………… [-10, 10]
- Friendly group ………………………... [-10, 10]
- Target assigned by commander .... [-10, 10]
- Enemy sensed by own sensors………... [-10, 10]
- Enemy positions sent via network…….. [-10, 10]

Attack (entered when enemy detected with organic sensor)

- Waypoint ……………………………... [-10, 10]
- Others in own group..…………………. [-10, 10]
- Friendly group ………………………... [-10, 10]
- Target assigned by commander .... [-10, 10]
- Enemy sensed by own sensors ………...[-10, 10]
- Enemy positions sent via network ...…... [-10, 10]

- Intermediate waypoint location for Group1 …… ($0 < x_1 < 10{,}000$, $0 < y_1 < 15{,}000$)
- Intermediate waypoint location for Group2 …… ($0 < x_2 < 10{,}000$, $0 < y_2 < 15{,}000$)
- If friend/enemy < *A*, Then *CommandA*
- If friend/enemy > *B*, Then *CommandB*

The last two listed items are of particular interest. Using these logic statements, command decisions (*CommandA* and *CommandB*) can be made based on friend to enemy ratios *A* and *B*. For our example, the commands are chosen from one of the three states of the Blue agents, i.e. *Advance*, *Flank*, or *Attack* for values of *A* and *B* of 1, 2 or 3.

The 24 possible factors are used to create a NOLH sampling pattern and the runs were performed in ABSNEC. The resulting number of kills for each run is shown in Figure 11. Similar to Figure 1, the one-to-one kill ratio is plotted in Figure 11 as a dotted line. Points above this line correspond to scenarios where more Blue were killed than Red. Note, there are no points where Blue was able to kill a large number of Red and receive less casualties than Red, i.e. points below the dotted line. In summary, no advantageous tactic was found for Blue when Red adopts the tactic of Sample 11 (from Section 2).

Now, assume the 150 Blue agents are given, in addition, a Beyond Line of Sight (BLOS) weapon (e.g.

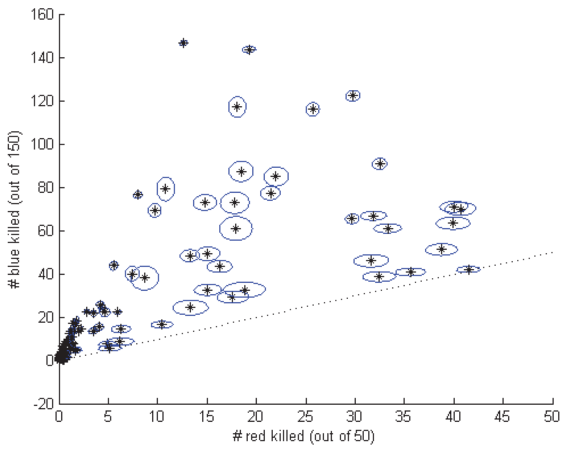


Figure 11 – Data farming results for Blue tactics against Red.

artillery). Data farming was performed using these sample points. The resulting kill ratios for each sample point are shown in Figure 12.

The points in Figure 12 are the average of 100 iterations at each of the sample points in the NOLH. There is a single outlier that exists below the dotted one-to-one kill ratio line and to the right of the graph. In other words, Sample 9 provides an advantageous tactic for Blue, where Blue is able to inflict more kills than casualties received. A screenshot for a single iteration of Sample 9 is shown in Figure 13 and the agent affinities, waypoint locations, and command rules are shown below.

In Sample 9, the agents start out at the top of the screen, spread out (due to their negative affinity towards friends), and move towards their intermediate waypoints (*Wp1* and *Wp2*). There is also repulsion between the two groups, *Blue1* and *Blue2*. This repulsion between groups causes the two groups to split up and surrounding Red.

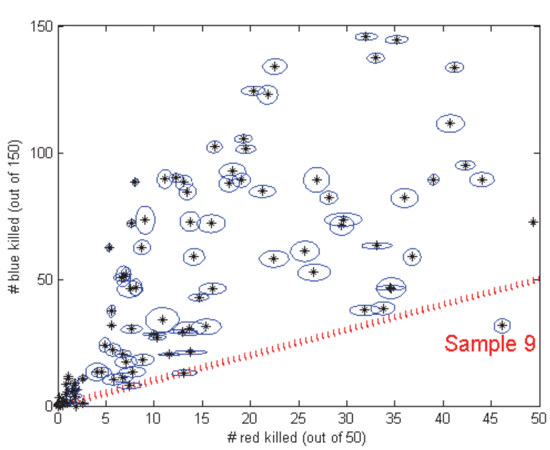


Figure 12 – Data farming results for Blue with a BLOS weapon added to Blue's side.

<u>Blue States and Affinities</u>

Advance

Waypoint …..………………………... 10
Others in own group ……...………..... -10
Friendly group ……………................. -9

Flank (entered when intermediate waypoint is reached)

Waypoint ………………….…………. 10
Others in own group ……….………... -1
Friendly group ……………………….. 5
Enemy sensed by own sensors ...……. 7
Enemy positions sent via network ..…. 5

Attack (entered when enemy detected with organic sensor)

Waypoint ……………………………. -5
Others in own group ………………... -6
Friendly group ………………………. -6
Enemy sensed by own sensors ……… 10
Enemy positions sent via network ..…. 5

<u>Blue Leader Command Choices</u>

- Intermediate waypoint location for Group1 …… (2739, 9156)
- Intermediate waypoint location for Group2 …… (2583, 11617)
- If friend/enemy < 3, Then *Flank*
- If friend/enemy > 3, Then *Attack*

Next, *Blue1* reaches its intermediate waypoint, *Wp1*, switches to the Flank state, and moves towards the final goal (where Red is located). *Blue2* continues to move towards *Wp2* by winding around Red's location. As members of *Blue2* reach *Wp2*, members in *Blue1* that have not reached *Wp1* move up over the north end of Red's location in order to maintain a separate distance from *Blue2*. Now, Red is completely surrounded and remains in one place, and as Blue agents reach their intermediate waypoints they move towards Red and come within sensor range. Then they send sensed Red locations to the Group Leader, who then sends it to the BLOS weapon. The BLOS weapon

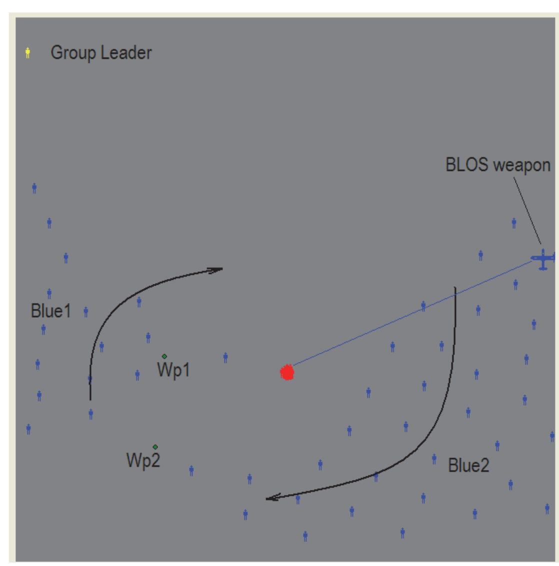


Figure 13 – ABSNEC screenshot from Sample 9

fires on the location of the Red agents it sees on its SA map which is 30 seconds old by the time it commences firing. Red and Blue agents have the same sensor and small arms weapons range. As a result, some Blue agents are killed as they get too close to the clustered Red agents, but because they approach the Red agents in small numbers and are spread out, the Blue casualties are kept small. In summary, an alternative Blue strategy has been developed to counter an effective Red strategy by exploring the parameter space using ABSNEC.

By combining data farming techniques and the distinctive feature of ABSNEC to represent realistic force structures with tiered C2 architectures, we have demonstrated the capacity to generate valuable, and perhaps unpredictable, combat strategies. Moreover, the simplicity of ABSNEC provides a viable method of performing quick turnaround studies to NEO/NEC scenarios.

CONCLUSION

We have introduced and highlighted some of the distinctive characteristics of the Canadian agent-based model known as ABSNEC (Agent-Based System for Networked Enabled Capabilities).

Next, the paper presents affirmative results of three validation techniques applied to date on the model: face validation (using validation by SMEs), model-to-model comparison (ABSNEC versus MANA) and simple statistical analysis/test against the well known Lanchester equations. The validation results instill an underlying confidence in the fidelity of the model in generating meaningful insights to complement complex military decision-making.

In the first illustrative example, we utilize ABSNEC to investigate the interrelationship between fratricide, combat stress, fear and situation awareness. The example opens the door to other 'human in the loop' operational studies. The second example combines ABSNEC's ability to represent realistic force structures with tiered C2 architectures and data farming techniques to generate unpredictable insights on combat strategies. In summary, the examples furnish compelling evidence to establish ABSNEC as a valuable analytical tool with which to better understand NEO/NEC concepts under the influence of the human error (human in the loop effect).

REFERENCES

Balci, O., 1998. Verification, Validation, and Testing, In: Banks, J., Ed. *The Handbook of Simulation*, New York, NY, John Wiley & Sons, Chapter 10, pp. 335-393.

Cioppa, T.M., and Lucas, T.W., 2007. Efficient Nearly Orthogonal and Space-Filling Latin Hypercubes. *Technometrics*, 49 (1), 45-55.

Engel, J.H., 1954. A Verification of Lanchester's Law. *Journal of the Operations Research Society of America*. 2 (2), 163-171.

Helmbold, R.L., 1994. The Constant Fallacy: A Persistent Logical Flaw in Applications of Lanchester's Equations. *European Journal of Operational Research*, 75 (3), 647-658

Horne, G., 2011. Data Farming for Canada. *Presentation at the Naval Postgraduate School*, Arlington, VA, February 2011

Ilachinski, A., 2000. Irreducible semi-autonomous adaptive combat (ISAAC): An artificial-life approach to land combat. *Military Operations Research*, 5 (3), 29-46.

Lanchester, F.W., 1956. Mathematics in Warfare, In: Newman, J.R., Ed. *The World of Mathematics, Vol. 4*, New York, NY, Simon and Schuster, pp. 2138-2157

McIntosh, G.C., Galligan, D.P., Anderson, M.A., and Lauren, M.K. 2007. MANA (Map Aware Non-Uniform Automata) Version 4 User Manual. *DTA Techical Note 2007/3*, NR 1465, ISSN 1175-6608. May 2007

Millikan, J., Brennan, M., and White, G. 1996. BactoWars Project Brief. Defence Science and Technology Orginisation, Land Operations Division, Ediburgh, Australia. Unclassified.

Ng, K., 2006. Complexity Analysis, Presentation at TTCP JSA, Melbourne, Australia.

Perry, W., 2005. Network-Based Operations for the Swedish Forces, *RAND*

Przemieniecki, J.S., 2000. *Mathematical Methods in Defense Analyses*, Washington, DC, American Institute of Aeronautics and Astronautics.

Railsback, S. F., Lytinen, S.L., and Jackson, S.K.. 2006. Agent-based simulation platforms: review and development recommendations. *Simulation* 82, 609-623

United States Department of the Army. 2007. Investigation into Civilian Casualties Resulting from an Engagement on 12 July 2007 in the New Baghdad District of Baghdad, Iraq. Washington DC, United States: United States Central Command

Yang, A., Curtis, N., Abbass, H.A., Sarker, R., and Barlow, M., 2006. WISDOM II: A Network Centric Model for Warfare. *ANU E Press*, Chapter 8, 149-173.

Richard McCourt received the B.A.Sc. degree (honours) in engineering physics in 2002, and the M.A.Sc. degree in mechanical engineering in 2004 from the University of British Columbia, Vancouver, BC, Canada. He is an Operational Research Scientist with the Department of National Defence where he is a member of the Air Staff Operational Research team.

Kevin Y. K. Ng received the B.Sc. degree (honors) in 1970, the M.Sc. degree in geophysics in 1971, the M.Sc. degree in applied math in 1972, and the Ph.D. degree in applied math/fluid dynamics from McGill University, Montreal, QC, Canada. He is a Senior Operational Research Scientist with the Department of

National Defence and an Adjunct Professor at the University of Ottawa, Ottawa, ON, Canada. Prior to joining the National Defence, he was an Engineer with the Canadian Aviation Electronics Ltd. and a Natural Sciences and Engineering Research Council of Canada Postdoctoral Fellow at the Canada Center for Remote Sensing. His research interest includes military operations research, operational logistics, and optimization in single and multiple objectives. He has published more than 40 refereed journal papers. He was an associate editor with the *Canadian Journal of Operational Research and Information Systems* and is currently an associate editor with the *Military Operations Research* Journal. Dr. Ng's publications in military operations research have earned him honorable mentions at the *INFORMS Koopman Prize* and the *Canadian OR Society Practice Prize* competitions.

Roy E. Mitchell received the Ph.D. degree in experimental atomic physics from the University of Western Ontario, London, ON, Canada. He is currently the Head Maritime and Joint ISR Operational Research at Defence Research and Development Canada's Centre for Operational Research and Analysis (DRDC CORA). He manages and conducts research on behalf of the Canadian Navy and the Department of National Defence. His research interests include C4ISR, Navy interoperability Joint C2 and Network Enabled Operations.

INCORPORATING REFLECTION INTO LEARNER AND INSTRUCTOR MODELS FOR ADAPTIVE AND PREDICTIVE COMPUTER-BASED TUTORING

ADAPTIVE AND PREDICTIVE COMPUTER-BASED TUTORING TRACK

Elaine M. Raybourn

Sandia National Laboratories* and Advanced Distributed Learning Initiative

emraybo@sandia.gov, elaine.raybourn@adlnet.gov

ABSTRACT

In the present paper the act of learner reflection during training with an adaptive or predictive computer-based tutor is considered a learner-system interaction. Incorporating reflection and real-time evaluation of peer performance into adaptive and predictive computer-based tutoring can support the development of automated adaptation. Allowing learners to refine and inform student models from reflective practice with independent open learner models may improve overall accuracy and relevancy. Given the emphasis on self-directed peer learning with adaptive technology, learner and instructor modeling research continue to be critical research areas for education and training technology.

Keywords: adaptive, predictive tutoring, games, learner model, Reflective Observer/Evaluator, instructor model, reflection

1. INTRODUCTION

Transforming education in the United States is a White House priority and a national security challenge. Homeland Security, Defense, and other Government agencies recognize the need to boost learner performance. According to Secretary of Education, Arne Duncan (2010), the staggering numbers of students that either fail to graduate or drop out of high school altogether is "economically unsustainable and morally unacceptable." Many educators attribute a one-size-fits-all approach to contributing to the failings of the education system (U.S. Department of Education, 2010). Without a strong foundation in education the United States will be unable to meet global challenges alongside other nations.

In response to the inadequacies of a one-size-fits-all approach, a National Science Foundation (NSF) study identified grand challenges for education technology such as personalization, assessment, and supporting social learning among several others (Woolf et al., 2010). In the cases of personalization and assessment the NSF study indicated that computational technologies might one day match the ability of a human tutor to understand individual learner's strengths and weaknesses and remove perceived boundaries between learning and assessment. Social learning could one day ensure continuous inputs from team members when the learner needs feedback the most. Additionally, the Army Learning Concept (ALC) 2015 envisions a future for education that leverages peer-based and self-directed learning. According to ALC 2015 "the future learning model must offer opportunities for Soldiers to provide input into the learning system throughout their career" as well as account for Soldiers' prior knowledge and experiences (ALC 2015, pages 6-7).

In each of these cases, real-time reflection plays a vital role. As we reflect we hone in on our strengths and weaknesses. Constructive criticism blurs the boundaries between learning and assessment, and each day we receive continuous inputs from our instructors and peers as we interact with them. Reflection is a dynamic activity. The challenge for adaptive and predictive computer-based tutoring is to take lessons learned from real-life reflection and incorporate them. However in order to do this, challenges in authoring, instructional strategy selection, and learner modeling for predictive and adaptive systems will need to be addressed to achieve desired outcomes from education technology (Sottilare, et al., 2011).

Adaptive tutoring has been defined as "the ability of an intelligent computer-based system to adjust to needs of the learner" (Sottilare et al., 2011). According to Sottilare and others, "adjustments to learner needs may be based on learner performance, behavioral and physiological sensor data, demographic data, personality profiles, mood surveys, and learner-system interaction" (2011, page 1). These systems often utilize what is known as a *student model* for purposes of description or prediction (Woolf, 2009). These student models are usually local to the application—that is, they are often treated as a component of a computer-based tutor, and not as an open, negotiated representation of learning. Adaptive and predictive computer-based tutors are typically standalone systems for individual users although research goals exist to extend these systems for eventual use with teams and groups (Sottilare, 2010).

The present paper proposes that incorporating aspects of social learning theory such as reflection and real-time observation and evaluation may support the automation of adaptive student models and instructional strategy selection. Student models can be static and simplistic, therefore quickly becoming inadequate as the complexity of ill-defined, cross-domain problems increase (Woolf, 2009). However learners who are given opportunities to refine their student models by reflecting on their own performance before and after training as well as reflecting on others' performance during training by performing real-time, peer evaluations may contribute information to learner models otherwise difficult to come by.

The potential opportunities to incorporate learner reflection that occurs before, during, and after training into adaptive and predictive computer-based tutoring systems are explored in the present paper. Two examples of making use of learners' reflection in adaptive systems are discussed: 1) reflection on peer performance occurring during adaptive or game-based training and 2) reflection on one's own performance before or after training via an independent open model such as an e-portfolio. The following section describes a role designed for real-time reflection in multi-player or team-based adaptive training and games.

2. INCORPORATING REAL-TIME REFLECTION INTO PREDICTIVE TUTORS

2.1. Reflection in Military Training

Reflection is a large part of all military training whether live or virtual. However, in most military training reflection largely occurs as a byproduct of face-to-face interactions with others, during group debriefings, or when lessons learned are taken from the classroom into the field. Military field training is often considered to be the most rigorous training experience available as it usually consists of mentally, emotionally, and physically challenging live exercises. These live exercises are role-play scenarios that can last up to two weeks and may involve a cast of approximately 1,600 role-players who collectively provide learners with experiences that rival a real-life situation (Tressler, 2007). Following the exercise, the learners and instructors discuss and debrief individual and/or team performance, and the consequences of actions taken in the scenario (Gredler, 1992). Military debriefings and hot washes are generally large or small group discussions in which performance is analyzed for what went wrong, what went right, and what could have been improved. It is common practice for sense-making to occur in a debriefing outside of the exercise context after one's performance has concluded. The challenge for the military is to engender real-time habits of reflection so that learners can debrief their own actions in situ, while they still have an opportunity to influence outcomes (Raybourn, 2007).

2.2. Player Role for Real-time Reflection

Very few computer games are designed to specifically engender habits of reflection even though this ability is a key metacognitive skill for successful learning. General perceptions of what constitutes computer game interaction or what behaviors constitute a "player" tend to closely align with trends in the entertainment industry even though opportunities exist for serious games and adaptive training to conceptualize both game play and "player" roles completely differently. According to Salen and Zimmerman (2004), rules as we know them in games can be broken and are sometimes transformed through the experience of social interaction. A unique opportunity therefore exists for serious games and military training systems to support real-time reflection and evaluation in-game with novel roles and new approaches to multi-player interaction.

As noted in the previous section, the opportunity to reflect on game-based training experiences or performance largely occurs after learning exercises have concluded, and outside of the exercise. Likewise in multi-player military training games the challenge of designing compelling learning opportunities that replicate live exercises is also usually met by separating action from reflection.

In the sections below we address the following questions: What are the implications of real-time reflection for team training? What are the affordances of using real-time reflections on peer performance for fine-tuning computer-based predictive tutoring models utilizing machine learning techniques?

2.3. Reflective Observer/Evaluator Role

A player role for real-time reflection based on the United States Government-owned Real-time In-Game Assessment, Evaluation and Feedback system was invented for a military training game developed for the U.S. Army Special Forces (Raybourn, 2007, 2009a). The design of the Reflective Observer/Evaluator role for multi-player games was inspired by the Special Forces' desire to hone intercultural competence and adaptive thinking through the practice of real-time reflection on actions taken, and the practice of providing constructive peer performance feedback. Operating competently in intercultural settings constitutes an ill-defined domain for predictive computer-based tutors and requires that the learner develop the ability to be aware of oneself and others, reflect on salient experiences, evaluate or assess situations, and act purposefully on those evaluations.

Early instantiations of the Real-time In-Game Assessment, Evaluation and Feedback system involved the instructor in-the-loop (Raybourn, 2009a,b) while a subsequent instantiation was developed for teams to work in pairs (Raybourn, et al., 2011). For example, in one of the scenarios for the Special Forces game a team conducts an area assessment of a local leader's courtyard. As the Detachment Commander communicates with the local leader, the instructor notices a behavior that she would like Reflective Observer/Evaluators to score. The instructor chooses the topic for evaluation (e.g. ethics) from a drop down menu on the instructor interface and instantly the request for evaluation appears in the Reflective Observer/Evaluators' interfaces. They enter a numerical evaluation and write comments in the text box as desired. The Detachment Commander is simultaneously evaluated by any number of Reflective Observer/Evaluators (e.g. 20 or more) on behaviors such as whether he exhibited cultural awareness, used appropriate nonverbal gestures, effectively built rapport, used clear communication, etc. The role-play scenario does not stop while Reflective Observer/Evaluators score their peer's performance. They can also enter annotations in the interface text field. Their feedback is quantitative and qualitative, logged by the game, and time-stamped. The evaluations across all Reflective Observer/Evaluators are aggregated and statistical analyses performed on their performance evaluations. The Reflective Observer/Evaluator interface has a scale bar in the lower left-hand corner that allows Reflective Observer/Evaluators to tap the space bar to raise or lower the bar to indicate team performance. Reflective Observer/Evaluators are able to evaluate both individual and team performance without becoming overwhelmed. Team and individual assessments can be displayed either in real-time or during group debriefings.

This approach to learning places real-time reflection directly in the training event, and gives Reflective Observer/Evaluators the ability to assess other players' performance and comment on events as they unfold. Following the game session, learners' roles can be switched and game play repeatability is preserved.

2.4. Reflective Observer/Evaluator Role for Learner Skill Development and Automated Knowledge Capture for Model Refinement

Learners in the Reflective Observer/Evaluator role observe, reflect, and evaluate the performance of another learner in real-time during role-play. The rationale for introducing the Reflective Observer/Evaluator role is fourfold. As described below the first two reasons primarily benefit the learner, while the latter two benefit the learner-system interaction with the express purpose of enhancing predictive capability and refining machine learning algorithms. In essence, the Reflective Observer/Evaluator role serves to train learners as well as train the system's predictive capability.

First, real-time reflection and assessment are introduced into training without having to stop or pause action. Several theories of reflection or reflective practice have been advanced (Kolb, 1984; Schön, 1983; Gibbs, 1988; Atkins & Murphy, 1994). Each includes reflecting, thinking, feeling, evaluating, and acting as key components. Reflection in the Reflective Observer/Evaluator role for games or adaptive training systems is treated the same way one would expect to exercise this skill in real-life.

Second, the introduction of this role allows different people to hone different cognitive processes at the same time, together. This also may increase content reuse and game play repeatability. By playing roles that allow learners to act (conventional player roles) and observe, reflect, & act (Reflective Observer/Evaluator role) different cognitive tasks are executed. Experiential Learning Theory's combined modes for grasping experience (watching or doing) via reflective observation and active experimentation and transforming experience (thinking or feeling) via abstract conceptualization and concrete experience provided a solid framework for the development of the Reflective Observer/Evaluator role (Kolb, 1984). For example more concrete, active experimentation (e.g. negotiating from a different cultural point of view) takes place with role-play itself, while abstract conceptualization and reflection is fostered by the Reflective Observer/Evaluator role (e.g. pause, observe the negotiation performance in light of the cultural context, critically evaluate best practices, and communicate feedback).

Third, large numbers of learners can participate as Reflective Observer/Evaluators in small group scenarios simultaneously. It therefore becomes possible to train an entire class on an intimate, small group exercise. It is also possible to obtain an aggregate evaluation of performance across a large number of participants to include experts, peers, and instructors. Reflective Observer/Evaluators may be anonymous and their feedback may assist the intelligent system in learning when, how, and why system feedback is appropriate.

Fourth, learners in the Reflective Observer/Evaluator role provide continuous inputs of subjective, value judgments and constructive feedback on performance that can be captured by the system. This may further refine computational models of human performance. For example, adaptive and intelligent tutoring systems often utilize a student model to "provide knowledge that is used to determine the conditions for adjusting feedback" for purposes of description or prediction (Woolf, 2009; p. 49). These systems often rely on discrete performance on well-defined problems, generalizations of expertise, and in worst cases, stereotypes of learners. The limitations of student models unfortunately contribute to computer-based games, simulations, tutors, and adaptive systems that are limited in perception and adaptability. The Reflective Observer/Evaluator role can assist with system capture of naturalistic data to include perceptions on whether, or to what degree, human performance is perceived by others to be good, effective, valuable, etc. Instead of refining a learner model based solely on inputs from the learner, now multiplayer inputs on how the human performance is perceived by others can be captured and incorporated into models that aim to predict learner or system performance, and select instructional strategies. This topic is explored more deeply in the subsequent section on reflection before and after training with independent, open, and negotiated models for adaptive systems and computer-based predictive tutors.

3. INCORPORATING REFLECTION FROM INDEPENDENT OPEN LEARNER MODELS INTO PREDICTIVE TUTORS

3.1. E-portfolios for Naturalistic Knowledge Capture

The rationale presented above serves to illuminate the different ways reflection can inform learner and instructor models in adaptive and predictive computer-based tutoring systems. The present section describes how independent open learner models can be used in military training to automatically populate student models for intelligent tutoring systems and adaptive training environments.

Platforms for aggregating and managing personal data residing on different desktop applications and internet services are an active area of computer science research (Kay & Kummerfeld, 2010). These prototypes aggregate pervasive computing sensors, online portals, and direct user input about personal health data. This approach characterizes a more naturalistic capture of social learning via *cognition in the wild*. Cognition in the wild refers to human cognition as it naturally occurs and adapts in the everyday world—situated in culturally constituted human activity (Hutchinson, 1995; Holland et al., 2000). Learner models can be informed by data capture via sensors that are typical in our learning environment such as desktop search aides, mobile devices, biometric sensors, social media, and the integration of these sensors into learning applications, as well as integration with e-portfolios.

E-portfolios (a.k.a. electronic or digital portfolios) are independent open learner models that are an education technology of interest in that they can provide opportunities for learner self-reflection before and after training with adaptive and predictive tutoring systems. An independent open learner model is an open learner model that is used independently of or external to a system (Bull, 2010). Open learner models are defined as student models that are accessible to the learner being modeled and possibly to teachers, peers, or others who may be able to enhance the model (Bull & Kay, 2007).

E-portfolios are under review by the International Standards Organization (ISO). E-portfolios enable learners to populate quantitative records, monitor, share skills, educational goals, competencies, outcomes, and achievements. E-portfolios are learner-managed and can aid decision-making as well as provide personal reflections beyond the abilities of most assessment systems typical of performance-based simulations/training environments and Learning Management Systems (LMSs) representative of formal learning and training. An example e-portfolio would have a variety of data fields for learner-generated quantitative and qualitative entries, as well as hooks to data sources for tracking formal and informal learning experiences (e.g. social media, Google Mail, Withings body scale and blood pressure monitors, etc.).

E-portfolios offer opportunities to infer learner attributes through data mining and statistical analyses. These data can set the initial challenge level in intelligent tutoring systems or adaptive systems avoiding the cold start problem where the system initially knows nothing about the user (Durlach, personal communication June 13, 2010; Bull & Kay, 2007) or where learner stereotypes are used (Woolf, 2009). E-portfolio components may also be used to enhance adaptation. Durlach and Ray (in press) distinguish between local and model-based adaptation. Local adaption involves providing feedback in adaptive and predictive computer-based tutors without taking explicit learner information into account whereas model-based adaptation takes the student model information into account to influence the sequence of instruction.

E-portfolio data in the form of peer or instructor evaluations may also serve to inform Negotiated Learner Models. Negotiated models may be preferred in instances when learners want the system to initiate interaction and negotiation. If the learner and the system have differing beliefs about knowledge representation, the negotiation process is initiated. Negotiated models can result in more accurate learner models and boost learner reflection. Incorporating the reflections (and perceptions) of others' performance into negotiated learner-model interaction could have implications for refinement of performance measurement in ill-defined domains and automating shared mental models for teams.

4. CONCLUSIONS

Learner modeling, instructional strategy selection, and authoring research will benchmark how personalized education and training is delivered to meet international security challenges. The present paper discussed how a role for real-time reflection and evaluation in military training games can be incorporated into adaptive and predictive computer-based tutors. Incorporating learner-system interaction such as reflection and real-time evaluation of peer performance can support the development of automated adaptation. Allowing learners to refine and inform student models from reflective practice with independent open learner models may improve overall accuracy and relevancy. It is the position of the present paper that we have only scratched the surface regarding leveraging reflection in adaptive and predictive computer-based tutoring. Military game training need not follow general game play assumptions and rules but rather can set the bar for how critical learning and meaningful social interaction is achieved through adaptive systems. The challenge for international militaries is to engender real-time habits of reflection such that learners can debrief their own actions in situ, while they still have an opportunity to influence outcomes.

ACKNOWLEDGEMENTS

*Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DEAC04-94AL85000.

REFERENCES

Atkins, S. and Murphy, K. (1994). Reflective practice. *Nursing Standard* 8(39) 49-56.

Bull, S. (2010). Features of an independent open learner model influencing uptake by university students. In P. De Bra, A. Kobsa, and D. Chin (Eds), User Modeling, Adaptation, and Personalization (UMAP) Lecture Notes in Computer Science (LNCS), Volume 6075, 393-98. Heidelberg: Springer.

Bull, S. (2004). Supporting learning with open learner models. 4th Hellenic Conference with International Participation: Information and Communication Technologies in Education, Athens, Greece.

Bull, S. & Kay, J. (2007). Student models that invite the learner in: The SMILI Open Learner Modelling Framework. *International Journal of Artificial Intelligence in Education*, 17(2), 89-120.

Department of Education (2010). Transforming American education: Learning powered by technology. National Education Technology Plan. Last retrieved from http://www.ed.gov/technology/netp-2010 on June 27, 2011.

Durlach, P. and Ray, J. (in press). Designing adaptive instructional environments: Insights from empirical evidence. Army Research Institute Report.

Gibbs, G. (1988). *Learning by doing: A guide to teaching and learning methods*. Further Education Unit, Oxford Brookes University, Oxford.

Gredler, M. (1992). *Designing and Evaluating Games and Simulations: A Process Approach*. Kogan Page, London.

Hollan, J., Hutchins, E., and Kirsh, D. (2000). Distributed cognition: Toward a new foundation for human-computer interaction research. ACM Transactions on Computer-Human Interaction, Vol. 7, No. 2, 174–196.

Hutchinson, E. (1995). Cognition in the wild. The MIT Press: Cambridge, MA.

Kay, J. & Kummerfeld, B. (2010, January). PortMe: Personal lifelong user modeling portal. Technical Report 647. Retrieved on May 21, 2011 from: http://sydney.edu.au/engineering/it/research/tr/tr647.pdf

Kolb, D.A. (1984). *Experiential learning: Experience as the Source of Learning and Development*. Prentice-Hall, NJ, USA.

Raybourn, E. M. (2007). Applying simulation experience design methods to creating serious game-based adaptive training systems. Interacting with Computers, 19, Elsevier, 207-14.

Raybourn, E.M. (2009a). In-Game Peer Performance Assessment Role that Fosters Metacognitive Agility and Reflection. In S. Natkin and J. Dupire (Eds.). International Conference on Entertainment Computing, International Federation for Informatio Processing 5709 (IFIP), 304-6. Springer-Verlag.

Raybourn, E. M. (2009b). Intercultural competence game that fosters metacognitive agility and reflection. In A.A. Ozok and P. Zaphiris (Eds.): Online Communities, Lecture Notes in Computer Science (LNCS) 5621, Springer-Verlag. 603–612.

Raybourn, E.M., Fabian, N., Glickman, M., Tucker, E., Willis, M. (2011). Real-time individualized training vectors for experiential learning. SAND 2011-0166, Albuquerque, NM, Sandia National Labs.

Salen, K., & Zimmerman, E. (2004). *Rules of Play: Game Design Fundamentals*. MIT Press, Cambridge, MA.

Schön, D. (1983). *The Reflective practitioner: How professionals think in action*. Temple Smith, London.

Sottilare, R. (2010). A perspective on the role of AI and digital media in military training. Keynote presentation to AI and Interactive Digital Entertainment Conference (AIIDE), October 11-13, Stanford University, Palo Alto, CA.

Sottilare, R., Goldberg, S., & Durlach, P. (2011). Research gaps for adaptive and predictive computer-based tutoring systems. International Defense and Homeland Security Simulation Workshop, 12 –14 September, Rome, Italy.

The U.S. Army Learning Concept for 2015. (2011). TRADOC Pamphlet 525-8-2.

Tressler, D. (2007). Negotiation in the new strategic environment: Lessons from Iraq. Strategic Studies Institute, Carlisle Barracks, PA.

Woolf, B. (2009). *Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning*. Burlington, MA; Morgan Kaufmann.

Woolf, B. (2010). A roadmap for education technology. National Science Foundation # 0637190.

AUTHOR BIOGRAPHY

Dr. Elaine Raybourn is a Principal Member of the Technical Staff at Sandia National Laboratories where she leads research in multi-role experiential learning, in-game reflection, and designing games that stimulate intercultural adaptability. She earned her Ph.D. in intercultural communication with an emphasis in human-computer interaction from the University of New Mexico and has been designing computer games for experiencing cultural dynamics for the past 11 years. Since 2003 Elaine has led teams that have developed, deployed, or transitioned cross-cultural engagement game-based training for the U.S. Army Special Forces, DARPA, USMC, and US Army PEOSTRI. Her team's work was identified by the Defense Science Board 2006 Summer Study on 21st Century Strategic Technology Vectors as "critical capabilities and enabling technologies for the 21st century that show promise." Elaine was an ERCIM (European Consortium for Research in Informatics and Mathematics) fellow at laboratories in Germany and France, and is a National Laboratory Professor at the University of New Mexico's Department of Communication. She is currently on assignment to PEOSTRI Games for Training, Advanced Distributed Learning Initiative, and Defense Acquisition University. Elaine is a recipient of the Department of the Army Award for Patriotic Civilian Service, awarded to her by the U.S. Army Special Forces Training & Doctrine Command.

VERIFICATION METHODOLOGY FOR SIMULATION MODEL BASED ON SYSTEM MORPHISM

Changbeom Choi[(a)], Jangwon Bae[(b)], Minwook Yoo[(c)], Tag Gon Kim[(d)] and Soohan KIM

[(a) (c) (d)]Dept. of Electrical Engineering, Korea Advanced Institute of Science and Technology
[(b)]Dept. of Industrial and System Engineering, Korea Advanced Institute of Science and Technology
[(e)] Video Display Biz. Technical Planning Group, SAMSUNG Electronics HQ

[(a)]cbchoi@sim.kaist.ac.kr, [(b)]jwbae@sim.kaist.ac.kr, [(c)]mwyu@sim.kaist.ac.kr , [(d)]tkim@ee.kaist.ac.kr,
[(e)] ksoohan@samsung.com

ABSTRACT

As the development of modern combat system grows rapidly, the importance of a development and the verification of the simulation model are also growing rapidly in simulation based acquisition. The simulation model of the combat systems can be defined by integration of simulation models, and simulation of the models is interaction among simulation models. Therefore, the verification for each simulation model and the verification of interaction among simulation models are important. In this paper we propose system morphic verification method to support the verification of simulation model with respect to the requirement specification, and we propose the incremental system morphic verification method to verify the interaction among the simulation models. The verification method based on the system morphism was used in the development of warship simulator to support the researcher of the national defense research institute.

Keywords: Simulation based Acquisition, Combat System Verification, DEVS Formalism, and System Morphism

1. INTRODUCTION

As the development of science and technology grows rapidly, the development of modern combat system is also growing rapidly. The combat system is a composition of subsystems, such as detection, weapon, and command and control systems. Such subsystems can be the composition of various kinds of equipment, such as, the composition of various search and track radars. Therefore, a combination of a combat subsystem and its components may affect the results of the battle. Consequently, the decision maker wants to try various combat systems with various scenarios. However, developing the combat systems and experiments in reality may consume lots of time, effort, and resources. For this reason, modeling and simulations are used to tackle this problem.

Simulations have become a useful part of the mathematical modeling of various natural systems, such as computational physics, chemistry and biology, human systems in economics, psychology, and social science, in order to gain insight into the operation of those systems, or to observe their behavior (Frigg and Hartmann, 2006). In general, the natural systems are composition of subsystems, which can be defined as the composition of various subsystems, recursively. Therefore, the simulation model of the natural systems can be defined by integration of simulation models, and simulation of the models is interaction among simulation models. Accordingly, checking the effectiveness of the new combat subsystem using its simulation model is possible. As a result, domain and M&S experts may develop a simulation model for the combat system and verify the simulation model that the subsystem is working properly while interoperating with other subsystems under the military doctrine.

In this paper, we propose the verification method based on the Modeling & Simulation (M&S) theory. The proposed verification method utilizes the system morphism to show that the implementation of simulation model satisfies the requirement specification. Moreover, as mentioned above, the combat system is divide into several subsystems that interoperates with other subsystems. Therefore, it can be modeled as either a single simulation model or an interoperation of simulation models which are the subsystem of the combat system. Therefore, we propose the incremental system morphic verification method to verify from the standalone simulator to the interoperation of simulators. In addition we introduce the case study of development of warship simulator. During the development of the warship simulator, the system morphic verification was used to verify the standalone simulator while the incremental system morphic verification method was used to verify the interoperation of simulators. In our case study we adapted the discrete event system specification (DEVS) formalism (Zeigler, Kim, and Praehofer, 2000) as a modeling methodology and

HLA/RTI as an interoperation environment (IEEE Std 1516, 2000).

The rest of this paper is organized as follows: Section 2 represents the background which related to modeling formalism and the theory of simulation model verification. In Section 3, we introduce simulation model verification using system morphism, and incremental system morphic verification methodology. In Section 4, we introduce the initiative case study for the verification of combat systems using the proposed methodology. Finally, we conclude the paper.

2. RELATED WORK

This section briefly explains the background knowledge which is applied to the combat system verification framework.

2.1. DEVS Formalism

The DEVS Formalism is formalism for the discrete event system modeling based on the set theory, and it is one of the M&S theories which are applied in various military simulations (Kim, Sung, Hong, Hong, Choi, Kim, Seo, and Bae, 2011). The DEVS Formalism supports to specify the discrete event models in hierarchical and modular manner. Therefore, the user may model the target system by decomposing large system into smaller components by applying coupling scheme among them. There are two kinds of models in the formalism: Atomic model and Coupled model.

2.2. System Morphism

System algebra is a mathematical tool used to express a real world system in a specific form with an attribute set and its binary relations. In general, an attribute set can be a system set, input/output event set, and time constraint set of a discrete event system of a real world system. Also, binary relations for any two attributes of system algebra describe the behaviors of the real world system.

The system morphism maps the relation from one system algebra to other system algebra with binary relation property preservation (Zeigler, Kim, and Praehofer, 2000). Formal representation of system algebra can be defined by three-tuples as follows:

Definition 1 System Algebra

A system S_A = <I, O, F> is system S_A, together with the following conditions:

I: system attribute set I, that indicate the input event set

O: system attribute set O, that indicate the output event set

F: I → O: binary relation F, that indicate system transfer relation

Definition 2 System Morphism

Let S_A and S_B are systems. Mapping relation ϕ: $S_A \rightarrow S_B$ relates the system structure of S_A to the system structure of S_B. Mapping relation ϕ is the system morphism if and only if satisfies the following system relation preservation condition:

$$\phi = \{\phi_I, \phi_O\}$$

$$\phi_I : S_A.I \rightarrow S_B.I$$

$$\phi_O : S_A.O \rightarrow S_B.O$$

$$\phi_O(S_B.F(\phi_I(i_A))) = S_A.F(i_A) \text{ for } \forall i_A \in S_A.I$$

Figure 1 shows the system morphism between two systems: S_A and S_B. The functions ***g*** and ***k*** are the mapping relation ϕ showing that the $System_A$ is system morphic to $System_B$ under the mapping relation $\phi = \{\boldsymbol{g}, \boldsymbol{k}\}$.

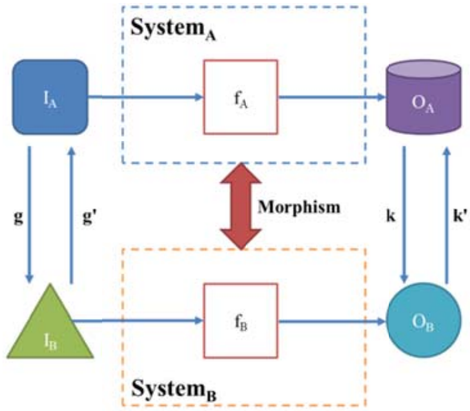


Figure 1: Morphism between two systems

In the M&S theory, the system morphism provides the foundation for simulation model verification. The system morphism assures the system structure preservation among systems. As a result, we can prove that the simulation model is valid and reflects the real system, or we can say that the simulation model is verified and meets the requirement specifications.

3. SYSTEM MORPHISM BASED VERIFICATION METHODOLOGY FOR SIMULATION MODEL VERIFICATION

In practice, implementation of the simulation model can be viewed as a system, where input is the model specification, output is the implementation of the simulation model, and the verification of a simulation model shows that implementation is flawless. This can be proved by the system morphism between the requirement and implementation of the simulation model.

In this chapter, we introduce the verification method using system morphism and incremental system morphic verification which utilizes the system morphism based verification method.

3.1. Verification Method using System Morphism

The verification of simulation model using system morphism defines two systems: requirement specification and implementation of the simulation model. Then it proves the existence of a transition rule that links two systems. As a result, the cooperation among domain experts and M&S experts is required for writing requirement specifications of simulation model and checking implementation of simulation model against requirement specification. Figure 2 describes the system morphic relation between requirement and the implementation.

First, the domain experts make a Simulation Logic Description (SLD) document that contains military domain knowledge such as field manuals and technical manuals. The SLD documents provide sufficient information for M&S experts to design and implement the simulation model. In addition, the SLD document usually embraces the UML diagrams (Booch, Rumbaugh, and Jacobson, 1999), such as use-case diagrams, class diagrams, and sequence diagrams to indicate the characteristics and behavior of a target system. When the SLD document is completed, M&S experts design the simulation model from the SLD document, and create the simulation model specification, which can be various modeling formalism.

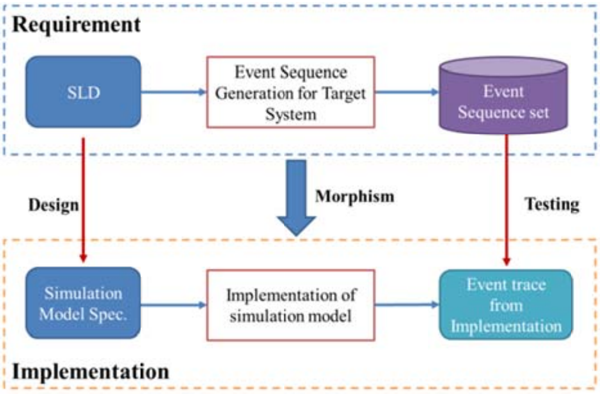


Figure 2: System Morphism between Simulation Model Requirement and Simulation Model Implementation

When the specifications of simulation model are completed, the M&S experts implement the simulation model. In order to verify the implementation of the simulation model, domain experts make the test cases, which are the desired event sequences of the simulation model. During the implementation of the simulation models, the domain experts make the Simulation Model Test Description (SMTD) document, which contains the input value for the simulation model, desired output, and the event trace among the simulation models. In other words, organizing the SMTD document generates an event sequence for the target system. Therefore, the domain and the M&S experts may verify the implementation of the simulation model using SMTD documents by checking input of the simulation model and the corresponding output result, and by monitoring the event traces among the simulation models.

3.2. Incremental System Morphic Verification

In general, adaptation of new technology may have high risk. Therefore, in order to support the decision makers, the simulation models for the target system are necessary.

The systems in the real world are complicated and consist of several sub-systems. For example, a combat system contains several sub-systems such as detection, combat fire control, and weapon systems. Therefore, in order to develop a simulation model for a combat system, the level of detail may vary. In other words, the simulation model of a combat system may contain various simulation models; some may be abstract, and others may be detailed enough to substitute for the real equipment.

As a result, during the simulation based acquisition, the developer may make various simulation models with various levels of detail and utilize the simulation models to simulate the target system. In general, there are two ways to simulate target system, building standalone simulator and organizing the interoperation of simulators. In general, the simulation models of the standalone simulator are usually hard-coded and embedded in the simulator so that the simulation model developer cannot easily modify or extend the simulation models. Moreover, the time required to develop standalone simulation is relatively less than the interoperation of simulators. On the other hand, in the interoperation of simulators, a simulation model can be mapped into a model that participates in the interoperation of simulators. Therefore, the user may extend and modify the simulation model easily, and may substitute an abstract simulation model into a detailed simulation model. Moreover, the simulation model can interoperate with real equipment so that the hardware-in-loop simulation is possible. However, the performance of the interoperation of simulators may be relatively lower and the required development time is longer than the standalone simulator. Figure 3 describes the phase of the simulation based acquisition using the interoperation of simulators. In this simulation based acquisition process, we develop and verify each simulation model in the standalone simulator. Then we extend the standalone simulator into an interoperation of simulators so that domain experts and M&S experts can develop more accurate simulators.

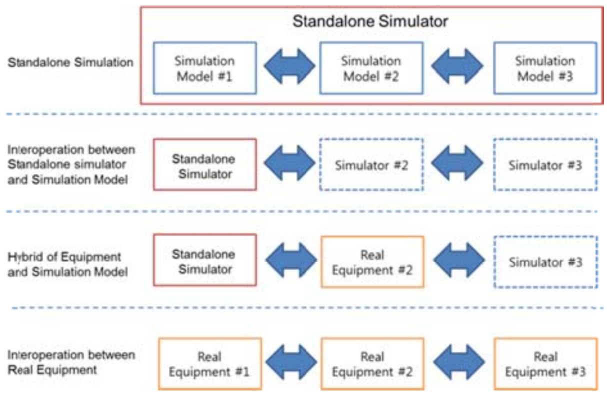


Figure 3: Phase of the simulation based acquisition using Interoperation of simulators

First above all, the M&S experts and the domain expert develop the simulation models and integrate them into standalone simulator. The standalone simulator can alter the parameters of the simulation models, and build a simulator by compositing various simulation models. In this phase the M&S experts and the domain experts can use the system morphism based verification method to verify each simulation model in the standalone simulator.

After building standalone simulator, the M&S experts and the domain experts should implement the interoperation features to support simulation between standalone simulators and other simulators. Note that in the interoperation of simulators, the simulator may be the detailed simulation model which reflects the real equipment better than the simulation models of the standalone simulator. Since the simulation models in the standalone simulators are abstract models of the real equipment, by implementing simulators for each simulation models can increase the correctness and reliabilities of the simulation results in distributed simulation environment. At the final phase, the domain expert and the M&S expert can test the real equipment with the simulation models.

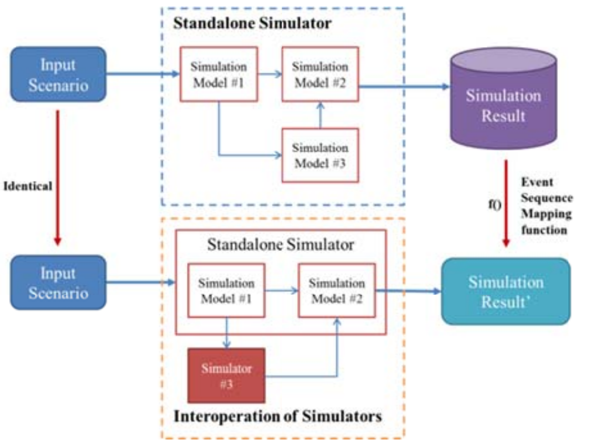


Figure 4: System Morphism between standalone simulator and the interoperation of Simulators

The figure 4 denotes the system morphism between standalone simulator and the interoperation of simulators. In order to give the basis of the verification, the domain experts and the M&S experts utilize the system morphic verification method to show that the standalone simulator and the requirements are in the system morphic relation. Then, the domain experts and the M&S experts utilize the input scenario of standalone simulator and its simulation results to verify that the interoperation of simulator satisfies the requirement specification. In other words, during the simulation based acquisition phases, the domain experts and the M&S experts find system morphism among each phase incrementally.

4. CASE STUDY: VERIFICATION OF SIMULATAION MODEL FOR WARSHIP SIMULATOR

In this section, we introduce the background and verification results of the simulation model of a warship simulator. However, this case study is related to the national defense of South Korea. Ergo, the name of the institute, the modeling results, and the simulation results are classified at the CONFIDENTIAL level. Therefore, we give the initiative information of the simulation models in this case study.

4.1. Background

The main objective of this case study is to develop a framework for the combat system of a warship, which allows engagement among the combat systems. Based on a survey of the Ship Air Defense Model of BAE Systems, the domain experts and the M&S experts have designed the M&S framework for the combat system. The characteristics of this framework are that it supports the simulation model to have various levels of details by implementing Plug and Play feature, supporting interoperation of the simulators, and providing human-in-the-loop simulation and hardware-in-the-loop simulation.

The simulation framework has two modes standalone mode and interoperable mode. The standalone mode contains a simple model of the combat system. Therefore, the user can easily check the trend of the engagement simulation of several combat systems. However, if a user wants to know the results of simulation without using an abstract simulation model, he or she can utilize the detailed simulator, such as MATLAB/Simulink models, by using HLA/RTI. Moreover, the simulation framework supports both human-in-the-loop simulation and hardware-in-the-loop simulation by using HLA/RTI.

In order to implement the warship simulator, we adopted the DEVS graph notation. Using a DEVS graph, Atomic model and Coupled model, the DEVS formalism can be expressed using various symbols. Figure 5 and Figure 6 denotes the DEVS Graph for the Atomic model and the Coupled Model.

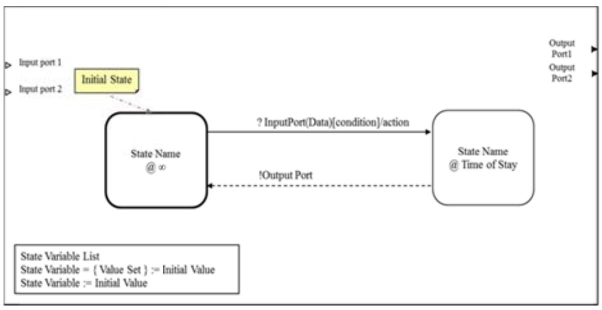


Figure 5: DEVS Graph for Atomic Model

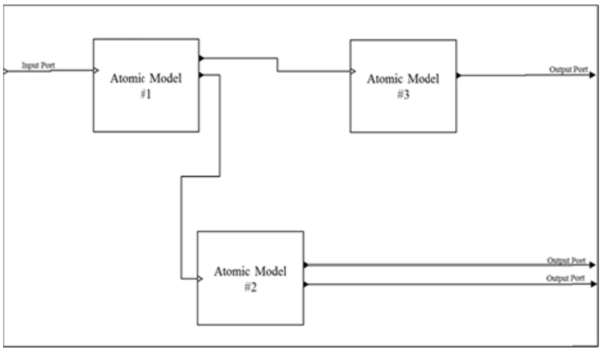


Figure 6: DEVS Graph for Coupled Model

4.2. Implementation

In order to develop the simulation model, we utilize two libraries to support the military experts and the M&S experts: the DEVSim++ and the KHLAAdaptor Library.

The DEVSim++ is a DEVS simulation environment based on C++ (Kim, 2007), and it provides several object-oriented features such as encapsulation, inheritance, and reusability. Moreover, it supports the distributed simulation environment by using HLA/RTI (Sung, Hong, and Kim, T.G., 2005; Kim, Hong, and Kim, 2006). The M&S experts have several advantages in simulation model verification when they use DEVSim++. First, if the simulation model requirements are described in DEVS Formalism, the M&S experts can implement the simulation model, which is atomic and coupled model directly. In other words, the simulation algorithm of DEVS formalism is implemented in DEVSim++, so that if the user implements the simulation model, the DEVSim++ manages the time scheduling and data handling. Therefore, the verification of simulation model using DEVSim++ can be viewed as monitoring the event sequence of the simulation model. Second, the DEVSim++ can cooperate with HLA/RTI, and helps the user to focus on the design and implementation of the simulation model rather than on the interoperation among simulators. Finally, the M&S experts can reuse the simulation model in an object-oriented fashion.

The KHLAAdaptor library enables developers to build an interoperation of simulators using HLA/RTI (Kim, Sung, Hong, Hong, Choi, Kim, Seo, and Bae, 2011). The KHLAAdaptor supports the developers to select the HLA services which will be used in the interoperation of simulators. The advantages using KHLAAdaptor library are twofold. First, the KHLAAdaptor library handles the time management and the data management, so that the developer may consider the encoding and decoding of the simulation message. As a result, the developers can focus on the design and implementation of the simulation model. Second, the KHLAAdaptor library handles the entire simulation messages of the simulation application during interoperation simulation. Therefore, the tester may collect the event sequences from the KHLAAdaptor to verify the interoperation of simulators.

4.3. Verification Result

The verification process for the simulation models of a warship combat system comprises of four phases as shown in the Figure 7. During Phase I, the military and M&S experts verify that the behavior of the simulation model based on the system morphism. In this phase, the domain experts make SLD document in natural language. Afterward, the domain experts extract the SMTD documents from the SLD; on the other hand, the M&S experts make Simulation Model Specification from SLD. In this case study, we choose the DEVS formalism as a simulation model specification to model the combat system and have drawn every DEVS graph in the warship simulator.

In order to verify the combat systems in the warship simulator, we have collaborated with the researchers of the national defense research institute. Figure 7 depicts the DEVS graph of the missile launcher. The M&S experts have developed the DEVS models based on the SLD document; both military and M&S experts inspect the DEVS graph against the SLD document. Figure 8 illustrates the DEVS graph for the missile launcher.

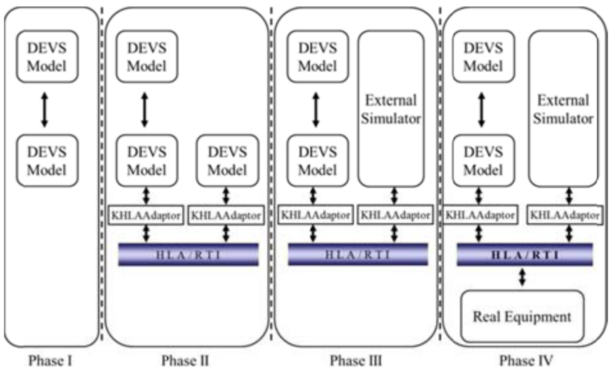


Figure 7: Verification process of simulation models of warship combat systems

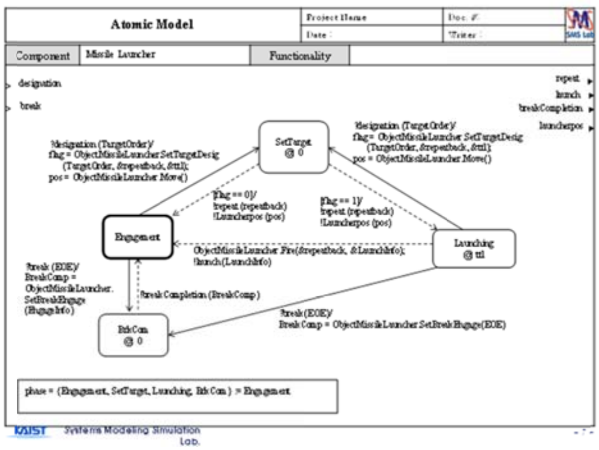


Figure 8: DEVS Graph for the Missile Launcher

Table 1: Air-to-Surface Missile Verification

Step	Action	Desired Output	Passed/ Failed
1	Model Initialized (set target combat	Check the initial parameters	Passed

	system)	though GUI	
2	Model Generation	Missile Launcher model launches the ASM	Passed
3	ASM follows the target combat system	Check the trajectory of the ASM	Passed
4	Model Destroy	Check through GUI	Passed

After drawing each DEVS graphs in the warship simulator, the M&S experts begin implementing the simulation models; the domain experts must extract test cases from the SLD documents and make the SMTD documents. Table 1 shows this part of the SMTD document.

In Phase II, we verify the design and the implementation of HLA modules, i.e. KHLAAdaptor. In order to verify the HLA modules, we use the DEVS models to simulate the detailed simulator. During this phase, we can verify the time synchronization, i.e. verify the modules that use HLA services, and data exchange, such as data encoding and decoding. After this phase, we can guarantee that the HLA modules are stable enough to test the interoperation between standalone simulator and detailed simulators.

In Phase III, we simply exchange the DEVS model into the External Simulator, i.e. the detailed simulation model for the real equipment. After we verify the external simulator by reviewing the source code and checking the log files, we can finally interoperate the real equipment.

5. CONCLUSION

This paper introduces a verification method based on the system morphism. System morphism is the relation between systems that shows the existence of a structure preservation function between two systems. The verification of the simulation model against the requirement specification is verifying the structure of the system. In other words, if we assume that designing and testing are the ideal structure preservation functions, the verification of the simulation model is that checking the test cases which are generated based on the requirement specification. We propose a system morphic verification method to support the verification of a simulation model with respect to the requirement specification, and we propose the incremental system morphic verification method to verify the interaction among the simulation models. The verification method based on the system morphism was used in the development of the warship simulator in order to support the researcher of the national defense research institute. Regarding future research, we will extend the incremental system morphic verification method to verify the various simulation models, such as discrete the time model and continuous model.

ACKNOWLEDGMENTS

This research is supported by Ministry of Culture, Sports and Tourism (MCST) and Korea Creative Content Agency (KOCCA) in the Culture Technology (CT) Research & Development Program 2009

REFERENCES

Booch, G., Rumbaugh, J., and Jacobson, I., 1999. *The Unified Modeling Language User Guide*. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc.

Frigg, R. and Hartmann S., 2006. *Models in Science.* Stanford Encyclopedia of Philosophy, Available from: http://plato.stanford.edu/entries/models-science/

IEEE Std 1516, 2000. *IEEE standard for modeling and simulation (M&S) High level architecture (HLA) - framework and rules,* doi: 10.1109/IEEESTD.2000.92296,

Sung, C.H., Hong S.Y. and Kim, T.G., 2005. Layered Structure to Development of OO War Game Models Using DEVS Framework. *Proceedings of Summer Computer Simulation Conference,* pp. 65-70. July 24-28, Philadelphia (Pennsylvania, USA).

Kim, J.H., Hong, S.Y., and Kim. T.G., 2006. Design and Implementation of Simulators Interoperation Layer for DEVS Simulator., *Proceedings of Summer Computer Simulation Conference,* pp. 195 – 199, July, Calgary, Canada.

Kim, T.G., 2007. *DEVSimHLA User's Manual.* Available from: http://smslab.kaist.ac.kr

Kim, T.G., Sung, C.H., Hong, S.Y., Hong, J.H., Choi, C.B., Kim, J.H., Seo, K.M., and Bae, J.W., 2011. DEVSim++ Toolset for Defense Modeling and Simulation and Interoperation, *Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, 8(3), 129–142.

Zeigler, B.P., Kim, T.G., and Praehofer, H., 2000. *Theory of Modeling and Simulation*. Orlando, FL, USA: Academic Press, Inc.

DETECTING DECEPTION WITHIN A PROBABILISTIC MODELLING FRAMEWORK

Ken R. McNaught

Cranfield University, Decision Analysis & Risk Modelling Laboratory, Dept of Informatics and Systems Engineering, Defence Academy, Shrivenham, Swindon, UK.

K.R.McNaught@cranfield.ac.uk

ABSTRACT

In this paper, we consider a number of analytic approaches to identifying or accounting for possible deception tactics being employed by an adversary. These are equally applicable to military or civil intelligence, or even law enforcement. As well as examining the Analysis of Competing Hypotheses (ACH) methodology, employed by some intelligence agencies, we focus on the contribution of methods for reasoning under uncertainty, particularly Bayesian networks (BNs). We also discuss the combination of these approaches as suggested by other authors. It is shown that the incorporation of negative evidence in addition to positive observations improves the performance of the BNs.

Keywords: deception, adversarial reasoning, military intelligence, intelligence analysis

1. INTRODUCTION

Deception is an integral part of human adversarial interaction. However, despite its well-proven and widely accepted value, including many historical accounts, there is relatively little scientific literature regarding its worth. In this paper, we view deception in the context of uncertain reasoning and so adopt a probabilistic modelling approach. This is aimed particularly at detecting possible deceptions.

Although deception is inextricably linked to psychology, the focus of this paper is on modelling issues rather than psychological ones. In particular, we focus on analytic approaches to detecting deception. Such approaches are applicable in a wide range of settings, including military and civilian intelligence and law enforcement situations. Weiss (2008) discusses some of the more general issues regarding uncertainty and its communication in the intelligence domain.

Deception can take many different forms. The two most general varieties, however, involve denial or hiding evidence which would be valuable to an adversary, or providing false and misleading evidence which it is hoped that the adversary will observe and believe.

The primary conceptual framework which we employ for considering deception is reasoning under uncertainty. In particular, Bayesian networks (Pearl, 1988) provide a powerful modern tool for such reasoning, based on probability theory. We also discuss the Analysis of Competing Hypotheses (ACH) methodology (Heuer, 1999) which is well known to many US intelligence analysts. Since deception frequently involves the use of misleading evidence, we also briefly consider some characteristics of evidence and the combination of evidence.

2. BAYESIAN NETWORKS

Bayesian network models are powerful and flexible decision support tools, supporting a wide range of analyses. The framework is practically proven in diverse application areas such as medicine, forensic analysis and industrial fault diagnosis. BNs permit the fusion of disparate information, combining observations with subjective expert opinion. They are robust to missing information, facilitate value of information assessments and can represent variable source credibility, a key requirement for intelligence analysis.

At heart, a BN is a compact and efficient representation of a joint probability distribution over a domain of variables of interest. What makes it so powerful and flexible is the ease with which it supports different types of reasoning or inference. Furthermore, although we might expect probabilistic calculations performed over a domain of many variables to be slow and cumbersome, BN software typically employs some sophisticated algorithms, making use of local computations (Lauritzen and Spiegelhalter, 1988). These local computations, themselves made possible by conditional independence assumptions regarding the variables in the domain, avoid the need to work with the whole joint probability distribution when making inferences, thus speeding up the task considerably.

The qualitative structure of a BN is represented by a directed acyclic graph (DAG), portraying probabilistic dependencies and independencies within the domain. Each node in the graph represents a variable in the domain of interest. Although continuous variables are permitted, they are usually discretised so that each variable typically has a small number of mutually exclusive states which it can be in. An arc between two nodes indicates a direct probabilistic dependence between them, while the absence of an arc indicates a conditional independence relation. Hence, the DAG contains a great deal of information, even before we consider any probability distributions. A fully specified BN, however, also requires the construction of conditional probability tables (CPTs) for each node. For parentless nodes, which have no arcs entering them, only a single prior marginal distribution has to be specified. For nodes with a single parent, a conditional probability distribution needs to be specified for each possible state of the parent variable. Finally, for chance nodes with several parents, a conditional probability distribution is usually required for every possible combination of parent states. While initially this may appear burdensome, in practice the requirement can often be relaxed, e.g. by making use of so-called Noisy-OR gates (Pearl, 1988) and their generalizations. This amounts to making certain reasonable independence assumptions, in exchange for a much simpler parameterization of the model.

There are many potential orderings of variables in a network, and the ordering chosen for a BN should represent the assumed dependencies and independencies as efficiently as possible. This usually means that the direction of an arc should follow the direction of causality when the relationship between two variables is causal. So, for example, it is the activities (or intent indicators) undertaken by a combat force which cause reports to be generated, the reports do not cause the activities to take place. Not all relationships in a BN have to be causal - weaker probabilistic dependencies will often be present. Exactly how such relationships should be represented and which way the arcs should be directed usually becomes clearer once the modeller has thought through their dependency implications. An invaluable guide in this respect is the d-separation criterion. See Pearl (1988) or Jensen (2001) for more details of this and for an introduction to Bayesian networks, more generally.

3. SCENARIO CONSIDERED

Here we employ a scenario described in McNaught et al. (2005). In it, a Blue HQ is trying to infer the intentions of a hostile Red force. The four possibilities considered are main attack (M), advance (A), defend (D) and withdraw (W). It is assumed that these are mutually exclusive, i.e. the Red force will only pursue a single course of action (CoA) at any given time. It is further assumed that each of these CoAs are equally likely at time zero, although this is not a general requirement and any prior distribution could be adopted. Several information cues or indicators of enemy intent are searched for by the Blue side in order to infer the Red CoA. Some cues may be associated with more than one CoA and some cues may be detected by more than one mechanism.

3.1. Bayesian Network of the Scenario

Space constraints prevent illustration of the entire BN so we present just a portion of it in Figure 1. This shows that the likely presence of indicators of enemy intent such as the establishment of airfields and counter-recce activities depend on the Red side's CoA. Furthermore, the probabilities of the two intent indicators displayed being observed or not by various mechanisms (e.g. air recce or ground recce) depend on the presence or absence of those indicators. Part of the timeline associated with this scenario is shown in Table 1.

Table 1: Timeline for Scenario.

Time Step	*Actions Taken by the Red Side and Indicators Detected by the Blue Side*
1	Blue establishes air and ground recce.
2	Red deploys air and ground recce as deception; Red increases counter-recce activities as deception; Red establishes dummy airfields as deception.
3	Red establishes demolition on bridges; Blue sub-unit reports sighting of Red recce (S3MA1); Blue ground recce reports Red counter-recce activities (S2MAD4); Blue air recce reports sighting of Red aux airfields (S1MA3).
4	Red conducts feint attacks; Blue ground recce report sighting of Red aux airfield (S2MA3) and demolition on bridges (S2DW15); Blue sub-unit reports local attacks (S3M8).
5	Red evacuates non-essential services; Blue sub-unit reports sighting of demolition on bridges (S3DW15).
6	Red employs smoke and jamming and a defensive frontage; Blue ground recce reports sighting of Red evacuation of non-essential services (S2W19) and Red's use of smoke (S2MW10); Blue sub-unit reports Red's use of smoke (S3MW10) and jamming (S3MW11); Blue Signals report Red's jamming (S4MW11); Blue sub-unit reports Red's defensive frontage (S3W18).
7	Red begins systematic destruction of bridges and commences withdrawal; Blue air and ground recce report sightings of Red destruction of bridges (S1W20 and S2W20).

3.2. Initial Results

As time progresses and fresh observations are made by the Blue side, the BN updates our belief in each Red CoA as shown in Figure 2.

The final probability distribution of 'Enemy Intent' is:

$$P(M \mid \text{All evidence}) = 0.39; P(A \mid \text{All evidence}) = 0.06;$$
$$P(D \mid \text{All evidence}) = 0.04; P(W \mid \text{All evidence}) = 0.51.$$

Although the BN eventually 'got it right', the true intent of the Red side only became apparent towards the end. For much of the time, 'Main Attack' seemed the likeliest CoA.

3.3. Results After Negative Evidence is Included

Events associated with one Red CoA or another which were not observed to take place were previously assumed unknown. Now, such events are treated as

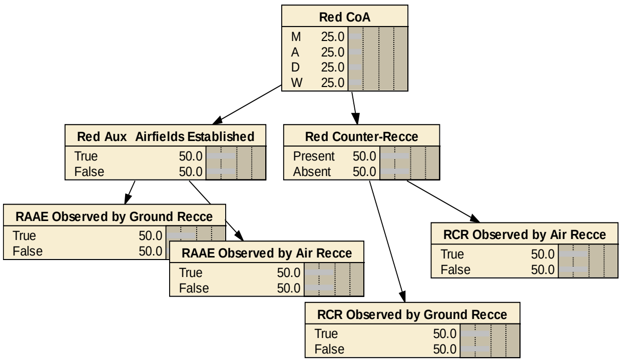


Figure 1: Partial BN of the scenario.

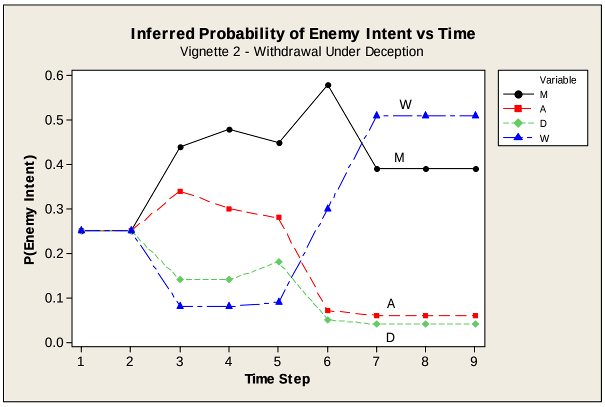


Figure 2: Probability Distribution of Enemy Intent vs Time

definitely not having occurred. The same underlying events are generated as in the first experiment, and the same positive intelligence reports are received at the same times. The difference is that in addition to the positive intelligence reports, there are now a number of 'negative' intelligence reports indicating that certain things have not been reported.

In deciding when to instantiate a report node with negative evidence, we have looked at the latest time we would expect a positive report to be received across the four possible states of Enemy Intent. If it has not been received by that time, we have instantiated a negative report for that indicator in the next time-step. The revised results for this scenario, incorporating the effects of negative evidence, are shown in Figure 3.

Clearly, this time the BN performs much better when the negative evidence is also taken into account. Firstly, the final distribution of 'Enemy Intent' is more decisive in each case. In Figure 3, the final distribution of enemy intent is now given by: $P(M \mid \text{All evidence}) = 0.01; P(A \mid \text{All evidence}) = 0.005;$ $P(D \mid \text{All evidence}) = 0.01; P(W \mid \text{All evidence}) = 0.975.$ Secondly, the correct option is identified earlier by the network. While it is difficult to quantify the benefit obtained by identifying the true enemy course of action sooner, this could be addressed in a simulation study.

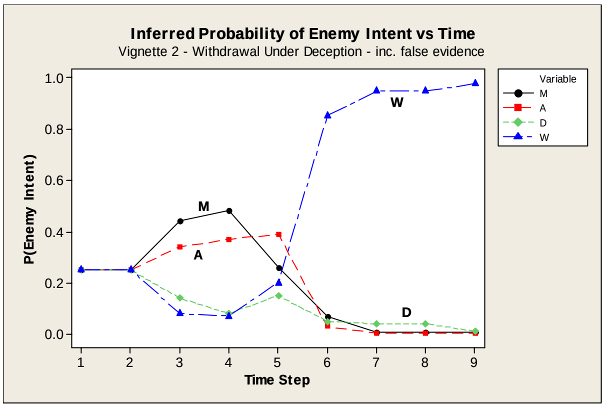


Figure 3: Probability Distribution of Enemy Intent vs Time with Negative Evidence Included

3.4. Conflict Measure as a Possible Indicator of Deception or a Missing Hypothesis

We investigate the use of a recognised BN conflict measure as a possible indicator of deception or a missing hypothesis. This measure is the ratio of the product of each piece of observed evidence's marginal probability to the joint probability of the observed evidence set, i.e.

$$\frac{P(e_1)P(e_2)...P(e_n)}{P(e_1,e_2,...,e_n)}.$$

The rationale behind this ratio is that when the observed pieces of evidence are generally in agreement, i.e. taken together they form a coherent hypothesis, the evidence will tend to be positively correlated and so the joint probability of the various observations in the denominator will be greater than the product of the marginal probabilities in the numerator, leading to a ratio less than 1. A ratio in the region of 1 would only be expected if the evidence variables were largely independent of each other. However, a ratio greater than 1 implies that the joint probability of the observations is less than the probability of their occurrence if they were independent. In other words, the evidence does not paint a coherent picture, which might indicate in an adversarial context that a deception is being undertaken or that a more realistic hypothesis has not been considered.

Figure 4 shows how this ratio changes over time in this scenario both when only positive observations are taken into account and when negative evidence is also included. The ratio is notably lower when negative evidence is included.

Using the above example, we now consider the situation where the correct hypothesis is not being considered. Hence, the Red CoA 'Withdraw' is removed as a possible hypothesis. After making this change to the BN and then entering the same set of positive evidence as before, the final probability distribution over the remaining Red CoAs is:

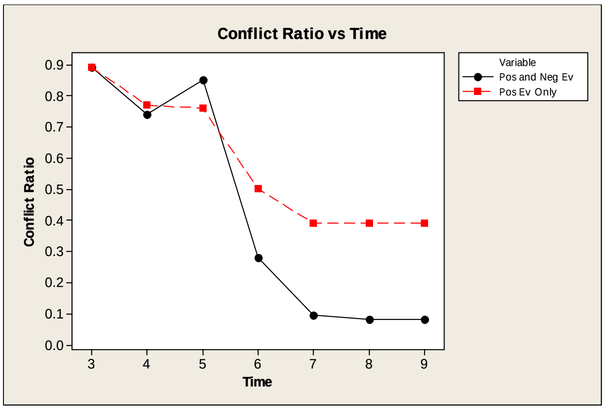


Figure 4: Conflict Ratio vs Time

P(M|Positive evidence) = 0.829; P(A|Positive evidence) = 0.103; P(D|Positive evidence) = 0.068. However, when negative evidence is also included, the final probability distribution becomes: P(M|All evidence) = 0.329; P(A|All evidence) = 0.209; P(D|All evidence) = 0.463. Clearly, when only positive evidence is considered, we may be mis-lead into believing that the Red CoA is a main attack. When negative evidence is included, however, the situation is much less clear with the single most likely hypothesis now being Red defence, the closest option to the true but unconsidered hypothesis of withdrawal. The conflict measure ratio is also plotted for these two cases in Figure 5. Note that this ratio never falls below 0.1. Although we cannot use such a threshold more widely, in a new situation we could possibly try to estimate it via simulation. A higher value could indicate, as here, that a new hypothesis needs to be considered, possibly one that is being masked by an adversary or otherwise seems implausible.

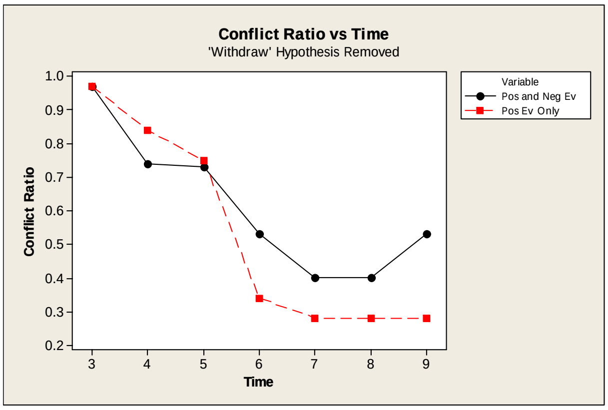


Figure 5: Conflict vs Ratio with the 'Withdraw' Hypothesis Removed

The next change that we consider is to introduce a new general hypothesis 'Other' while still leaving out the 'Withdraw' hypothesis. This represents a situation where the correct hypothesis is not among those being considered but nonetheless other possibilities are still being entertained. Such an approach is recommended within the ACH-CD method described in section 4.3. Results when only positive evidence is considered are presented in Figure 6. Results when negative evidence is included as well are presented in Figure 7.

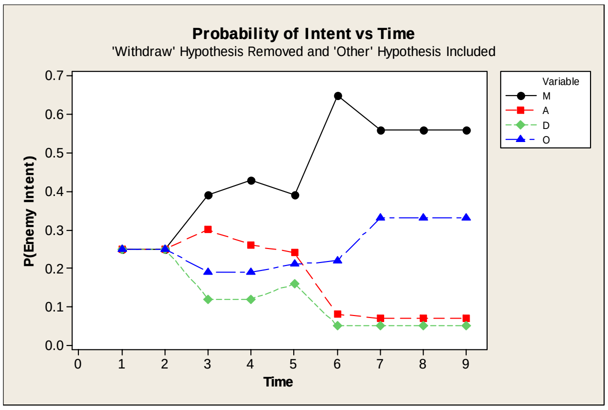


Figure 6: P(Enemy Intent) vs Time with Correct Hypothesis Removed and 'Other' Hypothesis Added

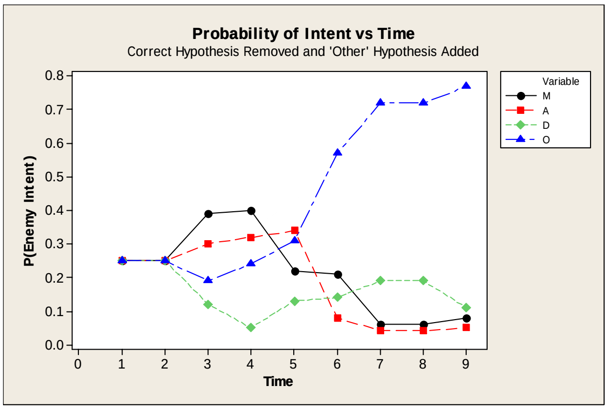


Figure 7: P(Enemy Intent) vs Time with Correct Hypothesis Removed and 'Other' Hypothesis Added

It is clear that the inclusion of negative evidence again improves the inference. The 'Other' hypothesis finishes strongly ahead of the remaining hypotheses, indicating that new possibilities need to be considered to explain the situation.

4. ANALYSIS OF COMPETING HYPOTHESES

Analysis of Competing Hypotheses (ACH) is a methodology developed by Heuer (1999) to help intelligence analysts overcome various cognitive biases, particularly confirmation bias. This is the tendency to overlook or underweigh evidence which contradicts the currently most favoured hypothesis, while overweighing supportive evidence. A matrix is developed in which the columns correspond to the set of plausible hypotheses and the rows correspond to items of evidence. The elements of the matrix record the extent to which each item of evidence supports or contradicts each hypothesis. Relevant negative evidence can and should also be included as rows in the matrix.

4.1. The ACH Framework

The basic outline of ACH is as follows:

1. Identify the alternative hypotheses to be considered.
2. Identify what evidence and assumptions are relevant to these hypotheses.
3. Construct the ACH matrix where the alternative hypotheses are the columns and each separate piece of evidence and assumption is a row.
4. In the matrix, indicate what evidence (including negative evidence) and assumptions supports or contradicts each of the alternative hypotheses, and by how much, removing that which does not discriminate between hypotheses.
5. Compare the relative likelihoods of all hypotheses, paying particular attention to evidence which contradicts a hypothesis, and identify future milestones when discriminating new evidence might come to light.

4.2. Partial ACH Example Matrix

Table 2 displays five example rows for an ACH matrix related to the scenario presented above. Evidence observations E1 and E2 are only weakly diagnostic as we might reasonably expect them to be present regardless of the Red CoA, although they are only slightly more likely for some CoAs than others. E3 is strongly associated with a Red Withdrawal. Failure to observe E4 (negative evidence) makes Defence or Withdrawal more likely than either offensive CoA while failure to observe E5 makes Defence a much less likely Red CoA. E6 is much less likely for the Withdrawal CoA but is employed as a means of deception in this scenario. While the Main Attack column may contain the most plus signs, note that overall, however, the Withdraw CoA column has the fewest minus signs and this is seen as more important in ACH. While human nature often makes us look for confirming positive signs, disconfirming negative signs may prove more valuable in many situations. This is a key motivation for the ACH framework and is particularly aimed at overcoming confirmation bias.

Table 2: Partial ACH Matrix

	Main Attack	**Advance**	**Defend**	**Withdraw**
E1: Red Radio Silence	++	+	++	+
E2: Red conducts feint attacks	++	+	+	+
E3: Red Evacuation of Various Services	-	-	-	++
¬E4:No observed forward movement of logistics	-	-	+	+
¬E5:No AT assets observed at frontline	+	+	--	+
E6: Red Counter-Recce Forces Operating	++	++	++	-

4.3. Combining ACH and BNs

ACH-CD (the CD standing for counter-deception) is an approach combining ACH and BNs, proposed by Elsaesser and Stech (2007). An example is provided concerning the Battle of Midway in which the position of US aircraft carriers is the basis of the deception. In another example concerning the D-Day landings, it is the transportable port facilities known as Mulberry which lies at the centre of the deception.

In their approach, a particular hypothesis of interest, H, is instantiated and the conditional probabilities of each piece of observable evidence given the hypothesis $P(e_i \mid H)$ recorded. Similarly, the condition 'not H' is then instantiated and the values of $P(e_i \mid \neg H)$ are obtained from the network. The ratio $\frac{P(e_i \mid H)}{P(e_i \mid \neg H)}$ indicates how important this piece of evidence is in discriminating between H and not H. In statistics, this ratio is well known as the likelihood ratio. Since it does not depend on the prior probability of the hypothesis, it is a direct measure of the weight of the evidence. For this reason, it has also become increasingly popular among forensic scientists, e.g. see Taroni et al. (2006).

5. EVIDENCE CHARACTERISTICS

A piece of evidence has many characteristics. These include relevance to the question being addressed, timeliness, since we might expect a newer observation to carry greater weight than an older one of the same type, and source credibility, particularly where human intelligence is involved. In reasoning about some adversary's intentions, we frequently need to combine multiple pieces of uncertain evidence with different degrees of relevance, different time stamps and coming from sources with varying degrees of credibility. Identifying unreliable sources is particularly important to reduce vulnerability to deception, as is identifying common or highly dependent sources.

Deception may affect the evidence marshalling process, described by Schum (2001). This concerns the organisation of evidence to make a case and might include analysis of evidence gaps, and notions of evidence thresholds to take different actions such as more intrusive surveillance or making an arrest.

These aspects could again be modelled utilizing the framework of a static BN. In the experimental, visual analytic 'Jigsaw' system (Stasko et al. 2008) developed to help intelligence analysts navigate a vast array of potentially relevant documents, provision is made for a ‘shoebox’ which is essentially an evidence marshalling tool. Such a tool can help an analyst to organise the available evidence, so aiding the construction of a coherent case.

As well as organising available evidence, such a tool can help highlight gaps in the evidential support for a hypothesis. With often very limited resources, support is required to identify the most promising gaps to investigate. Probabilistic decision support tools such as Bayesian networks can help in such situations.

6. CONCLUSION

Deception fundamentally involves reasoning under uncertainty and probabilistic modelling therefore suggests itself as a potentially useful framework for modelling and trying to detect deception activities. In this paper, we presented a scenario involving a Red adversary with four possible courses of action available, representing the set of alternative hypotheses being considered. We showed how a Bayesian network of the situation could be constructed and used to update our belief distribution over the alternative hypotheses as new observations were made by our recce units and other assets. In particular, we noted that the incorporation of negative evidence alongside positive observations improved the ability of the BN to infer the correct course of action.

We also examined what would happen if the correct hypothesis was not even under consideration. When the correct ‘Withdraw’ hypothesis was simply removed, leaving the other three, the BN performed poorly with only positive evidence. It did improve, however, when negative evidence was also included. When a general alternative hypothesis of ‘Other’ was added to the three remaining CoAs, however, the BN performed better. Although still coming to the wrong final conclusion of ‘Main Attack’ when using only positive evidence, the ‘Other’ hypothesis was considered the second most likely with a sizeable posterior probability of 0.33. When negative evidence was also included, however, the hypothesis ‘Other’ ended up the most likely with a posterior probability of 0.77, leaving all other hypotheses trailing well behind.

While deeper and more wide-ranging investigation is required, this suggests that in highly uncertain situations, particularly where deception is prevalent, we should more routinely consider adding such an alternative to the set of hypotheses being considered.

REFERENCES

Elsaesser, C. and Stech, F., 2007. Detecting deception. In: Kott, A. and McEneaney, W.M., eds. *Adversarial Reasoning – Computational Approaches to Reading the Opponent’s Mind*. Boca Raton, FL: Chapman & Hall, 101-124.

Pearl, J., 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann.

Heuer, R., 1999. *The Psychology of Intelligence Analysis*. Washington DC: Center for the Study of Intelligence, CIA.

Jensen, F.V., 2001. *Bayesian Networks and Decision Graphs*. New York: Springer. Mercer, P.A. and Smith, G., 1993.

Lauritzen, S.L. and Spiegelhalter, D.J., 1988. Local computations with probabilities in graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, Series B*, 50, 157-224.

McNaught, K.R., Ng, B. and Sastry, V.V.S., 2005. Investigating the use of Bayesian networks to provide decision support to military intelligence analysts. In: Merkuryev, Y., Zobel, R. and Kerckhoffs, E.J.H. (Eds.), *Proc. 19th European Conference on Modelling and Simulation*, 72-79. June 2005, Riga, Latvia.

Schum, D.A., 2001. Evidence marshaling for imaginative fact investigation. *Artificial Intelligence and Law*, 9, 165-188.

Stasko, J., Gorg, C. and Liu, Z., 2008. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization*, 7, 118-132.

Taroni, F., Aitken, C., Garbolino, P. and Biedermann, A., 2006. *Bayesian Networks and Probabilistic Inference in Forensic Science*. Chichester: Wiley.

Weiss, C., 2008. Communicating uncertainty in intelligence and other professions. *International Journal of Intelligence and Counter-Intelligence*, 21, 57-85.

AUTHOR’S BIOGRAPHY

Ken R. McNaught is a senior lecturer in Operational Research (O.R.) at Cranfield University’s School of Defence and Security situated at the UK’s Defence Academy in Shrivenham. He has an MSc in O.R. from Strathclyde University and a PhD in O.R. from Cranfield University. He leads the Decision Analysis and Risk Modelling Lab where his research interests include simulation, combat modelling and decision support, particularly making use of probabilistic graphical approaches such as Bayesian networks and influence diagrams. He also teaches on a number of specialized MSc courses, including Military Operational Research and Defence Simulation and Modelling.

DEVS-BASED VALIDATION OF UAV PATH PLANNING IN HOSTILE ENVIRONMENTS

Alejandro Moreno [(a)], Luís de la Torre [(a)], José L. Risco-Martin [(b)], Eva Besada-Portas [(b)] Joaquín Aranda [(a)]

[(a)]Dpto. Informática y Automática, Univ. Nacional de Educación a Distancia (Madrid, Spain)
[(b)]Dpto. Arquitectura de Computadores y Automática, Univ. Complutense de Madrid (Madrid, Spain)

[(a)]amoreno@bec.uned.es, ldltorre@bec.uned.es, jaranda@dia.uned.es [(b)]jlrisco@dacya.ucm.es, evabes@dacya.ucm.es

ABSTRACT
Discrete Event Specification (DEVS) is a sound formal modeling and simulation framework based on concepts derived from dynamic systems theory. DEVS provides a framework for information modeling with several advantages to analyze and design complex systems: completeness, verifiability, extensibility, and maintainability. Unmanned Aerial Vehicles (UAVs) are aircrafts without onboard pilots that can be controlled remotely or fly autonomously based on pre-programmed flight routes. They are used in a wide variety of fields, both civil and military. This research work is focused on taking advantage of DEVS simulation framework to build models that simulate a complex military problem. The simulator is used to validate the results of a route planner for multiple UAVs. The path planner uses several approximations to compute solutions in affordable time, whereas the simulator uses accurate models to validate those results.

Keywords: DEVS, UAV, simulation, path planning

1. INTRODUCTION AND RELATED WORK

Unmanned Aerial Vehicles are aircrafts without onboard pilots that can be controlled remotely or fly autonomously based on pre-programmed flight routes (Stevens and Lewis 2004). They are used in a wide variety of fields, both civil and military, such as surveillance, reconnaissance, geophysical survey, environmental and meteorological monitoring, aerial photography, and search-and-rescue tasks.

In military missions they work in dangerous environments, where it is vital to fly along routes which keep the UAVs away from any type of threat and prohibited zone. The threats of our problem are ADUs, which consists on detection radar to discover the UAVs, a set of tracking radars to follow their trajectories and a set of missiles to destroy them. The prohibited zones, also known as Non Flying Zones (NFZs), are certain regions that the UAVs cannot visit due to mission restrictions.

The best routes for the UAVs are those which minimize the risk of destruction of each UAV and optimize some planning criteria (such as flying time and path length) while fulfilling all the physical constraints of the UAVs and its environment, plus the restrictions imposed by the selected mission (such as forcing the UAVs to visit some points of the map).

Therefore, the motivation of this research is to validate the results of a route planner for multiple UAVs (Besada-Portas et al. 2011) applying DEVS formalism. The path planner uses several approximations to compute solutions in affordable time, whereas the simulator uses accurate models to validate those results.

In order to evaluate the quality of the planner before using it in real missions, we decide to validate the routes in multiple experiments against a simulator that contains models for all the elements of the problem.

In this problem, those elements are the lists of way points (WPs), the UAVs, the radars and the missiles; as well as the terrain, and the controllers coupled with the UAVs, this last group responsible for translating the WPs in maneuverability instructions. The models of the radars and missiles are non-deterministic, incorporating stochastic behaviors related with the probability of detection and destruction of the UAVs. So, two simulations for the same experiment and optimal trajectories can return different results.

The simulation symbolizes a scenario where one or more unmanned aerial vehicles must follow a given trajectory trying to avoid flying within air defense unit's visibility range. The trajectories are calculated prior to the simulation, each trajectory consists of a sequence of way points, each way point may also be a getaway.

To properly simulate this set of elements, correct models have to be used as the basis of the simulation. Traditional flight simulators, such as Microsoft Flight Simulator, FlightGear and X-Plane have very accurate aerodynamics models incorporated in their programs, but they do not include other elements like Air Defense Units or UAV's embedded radars.

In addition, these programs need a lot of memory and computing time to accurately calculate UAV's position and attitude. In this regard, a number of multi-UAV simulations have already been developed. In Rasmussen and Chandler (2002), the authors propose a Matlab-based model of multiple UAVs. However, their model can only be run for up to eight elements. To avoid these limitations, some approaches based on cellular automatas have been presented Glickstein and Stiles (1992), Shem, Mazzuchi and Sarkani (2008), Holman, Kuzub and Wainer (2010). For example, in the latter, the 3D cell space is divided into three 2D layers, where the first layer stores UAV position and previous travel path, the second one stores the target location probability terrain, and the last level stores the UAV restricted boundary information. Other approaches consist of agent-based models, like Lundell et al. (2006) and Karim, Heinze and Dunn (2004). However, this paper is focused in the employment of DEVS methodology for development of UAV simulators using the Component Based Development (CBD) process, which is a software development paradigm to assemble applications from reusable, executable software pieces called Components. Once developed, a component is repeatedly used in many projects via well-defined CBD interfaces, which greatly reduces the software development cost and increases the reliability (Kim et al. 2007).

This paper is organized as follows. Section 2 collects some relevant aspects of DEVS. Section 3 describes the modeling of the upper mentioned scenario. Section 4 presents the experiments and results of the simulation. Finally, in section 6 some conclusions are drawn.

2. DEVS

The Discrete Event System Specification is a general formalism for discrete event system modeling based on set theory (Zeigler et al. 2000). It allows representing any system by three sets and five functions: input set (X), output set (Y), state set (S), external transition function (δ_{ext}), internal transition function (δ_{int}), confluent function (δ_{con}), output function (λ), and time advanced function (ta). DEVS provides a framework for information modeling with several advantages to analyze and design complex systems: completeness, verifiability, extensibility, and maintainability. DEVS can also approximate continuous systems using numerical integration methods. Thus, simulation tools based on DEVS are potentially more general than others including continuous simulation tools (Kofman 2004).

DEVS defines system behavior as well as system structure. System behavior in DEVS is described using input and output events as well as states. To this end, DEVS has two kinds of models to represent systems: atomic model and coupled model. The atomic model is the irreducible model definition that specifies the behavior for any modeled entity. The coupled model is the aggregation/composition of two or more atomic and coupled models connected by explicit couplings between ports. The coupled model itself can be a component in a larger coupled model system giving rise to a hierarchical DEVS model construction. The top-level coupled model is usually called the root coupled model.

DEVS models can be simulated with a simple ad-hoc program written in any language. In fact, the simulation of a DEVS model is not much more complicated than the simulation of a Discrete Time Model. The problem arises with models composed by many subsystems where ad-hoc programming becomes very hard. One of the simplest ways to implement these complex models is writing a program with a hierarchical structure equivalent to the hierarchical structure of the model to be simulated. This is the method used in (Zeigler et al. 2000), where a class called Simulator is associated to each atomic DEVS model and a different class called Coordinator is related to each coupled DEVS model. At the top of the hierarchy there is a Coordinator, usually called the Root Coordinator that manages the global simulation time.

3. MODEL

To develop a more formal simulator, we redefine the behavior of all the elements of the system following the DEVS modeling formalism. Although different DEVS tools can be used for this purpose, from the modeling point of view, they are based on atomic and compound model definitions presented in section 2.

For this problem, each example is constructed upon multiple DEVS atomic components that exemplify an UAV dynamics and behavior, together with multiple DEVS coupled components that characterize the line of action of an ADU (see Figure 1). Each ADU is composed by various heterogeneous DEVS atomic components: detection radar; several tracking radars; and certain number of missiles. Wiring rules are depicted by Figure 2.

Below, we describe first the couplings type, and afterwards the structure, behavior and couplings layout of each model.

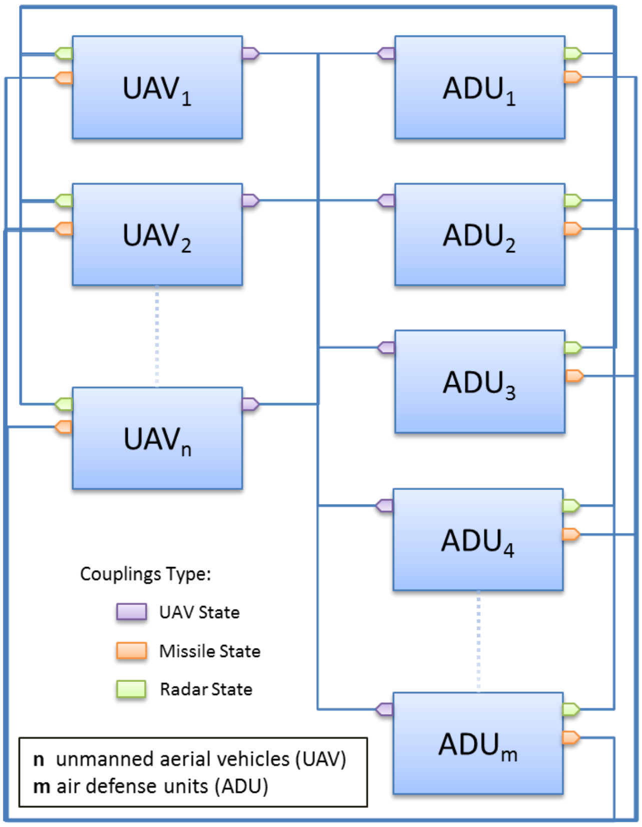


Figure 1 Root Coupled Model

```
For each uav in UAVs
    For each adu in ADUs
        coupling (UAV State) from uav out(0) to adu in(0);
        coupling (Radar State) from adu out(0) to uav in(0);
        coupling (Missile State) from adu out(1) to uav in(1);
```

Figure 2 Couplings pseudo code

3.1. Couplings

Couplings are directed edges that link output ports to input ports, defined by a type that bounds the data exchanged between models. The DEVS model described in this document uses up to four different coupling types.

- **UAV state**: UAV data type that encapsulates the necessary information to know who, when, where and how.
- **Missile state**: carries the missile information as an UAV, current missile phase and target identifier.
- **Radar tracking state**: covers the tracking radar identifier, the ADU where it belongs, and the data of the UAV to track.
- **Lost target**: the identifier of the UAV lost during the tracking process.

```
UavState {
  String id;              //identifier
  Time t;                 //computed in time
  Double X,Y,Z;           //X, Y and Z coordinates
  Double Vx,Vy,Vz;        //X, Y and Z velocities
  Double theta,phi,psi;     //roll,pitch,yaw
  Double Vtheta,Vphi,Vpsi;//roll,pitch,yaw vels
}
MissileState {
  UavState uav;       // unmanned aerial vehicle
  String phase;       // phase
  String target;      // target identifier
}
RadarTrackingState {
  String id;          //identifier
  String adu;         //air defense unit
  UavState uav;       //unmanned aerial vehicle
}
String lostTarget: // identifier of lost target
```

Figure 3 Couplings types

3.2. UAV

Unmanned aerial vehicles are represented as models assembled with two input ports that receive states of tracking radars and missiles of each ADU correspondingly. One output port that sends its state gathering the computed time, identifier, position, orientation and velocities. Additionally, keeps an internal state variable with an array of these UAV states reflecting changes in time of the UAV dynamics. Whenever an UAV realizes that is been detected by an ADU, starts if possible an evasion maneuver to escape from the ADU fire power range and prevent from being shot down.

Basically, the UAV model works in following way. Every time the internal time event function is triggered (simulation time is equal to sigma), the next necessary collection of states is computed, unless this collection is not empty or the UAV has reached the end of the trajectory. These states store intermediate values of position, orientation and velocities that describe the UAV's movement across the current coordinate to the next trajectory point. Then, sigma (time of next internal time event) is updated to the next computed time or set to ∞ only if the UAV reached the end of the assign path. Whenever the external transition is executed (received an input), the UAV verifies if any radar is tracking his path and whether the distance from a missile aimed at overthrowing it, is less than the established minimum. If the former case is positive, attempts to escape through an intersecting trajectory to flee away from the corresponding ADU and afterwards updates sigma. If the latter is positive and according to a certain probability of destruction, the UAV is destroyed and sigma is set to ∞. On every occasion that the output function is activated the current UAV state is sent thought the output port. Figures 4 and 5 depict this behavior.

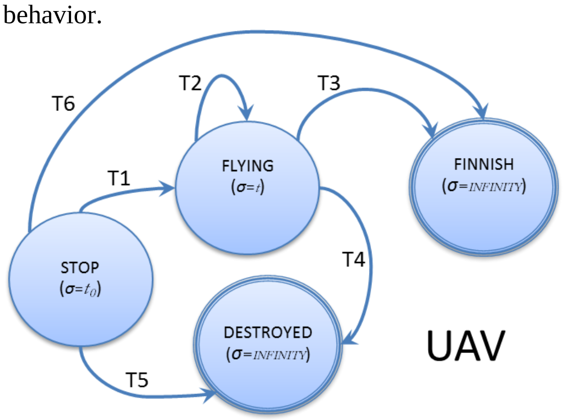


Figure 4 UAV State Diagram

	FLYING	FINNISH	DESTROYED
STOP	T1	T6	T5
FLYING	T2	T3	T4

	δ_{int}	δ_{ext}
T1	*states is NOT empty*	
T2	*states is NOT empty*	*dist. to missile is > MIN*
T3	*states is empty*	
T4		*dist. to missile is <= MIN*
T5		
T6	*states is empty*	

Figure 5 UAV State Transitions

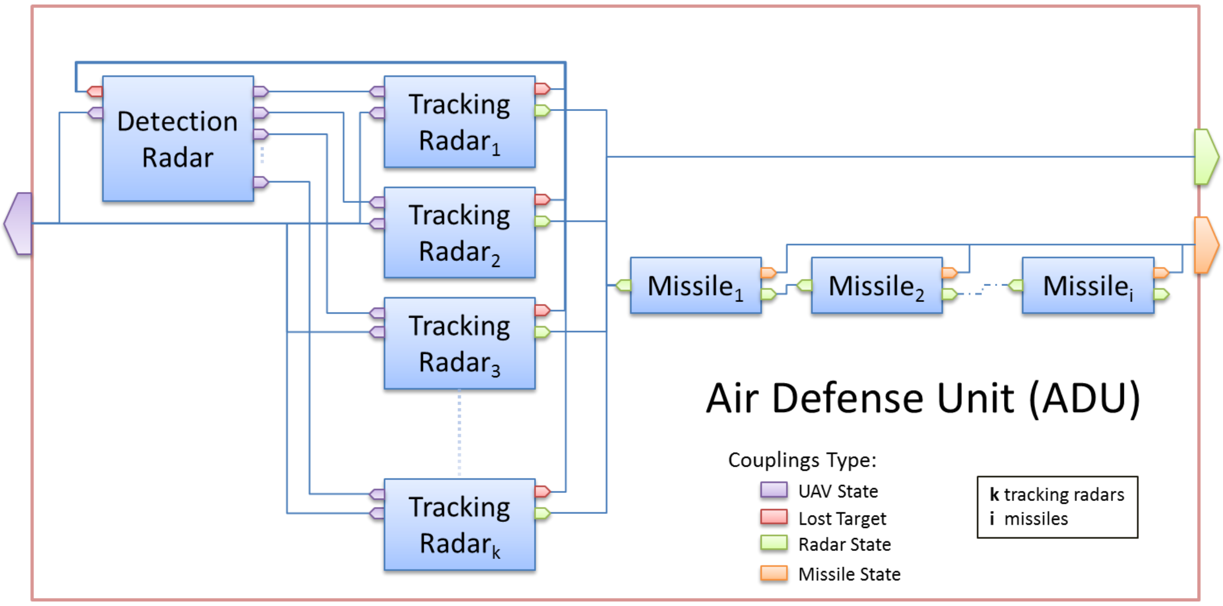


Figure 6 ADU DEVS model

3.3. ADU

Air defense units are coupled models formed by one detection radar, multiple tracking radars and multiple serial connected missiles. Figure 6 depicts the DEVS based ADU model structure and Figure 7 illustrates the pseudo code that clarifies how the wiring is set given the atomic models of all mentioned elements. Detection radars scan the skies seeking for UAVs. If a detection radar detects an UAV in its proximity then looks for task-free tracking radars. Assigns the spotted target UAV to an unoccupied tracking radar. The tracking radar attempts to detect the target UAV, if successful, alerts the first missile model in the row with the corresponding target. If the missile state is already fired, it hands the target to next missile in the row. In essence, on every occasion an ADU detects an UAV, after a specified period of time, shoots a missile to attempt to knock down the UAV. In the subsequent sections, each component is described in more detail.

```
p = 1;
For each tr in Tracking Radars {
    coupling (UAV State) from DetectionRadar out(p) to tr in(0);
    coupling (Lost Target) from tr out(0) to DetectionRadar in(0);
    coupling (Radar State) from rt out(1) to Missile1 in(0);
    coupling (Radar State) from rt out(1) to ADU out(0);
    p++;
}
j=1;
For each m in Missiles {
    coupling (Missile State) from m out(0) to ADU out(1)
    if (j < i)
        coupling (Radar State) from m out(1) to Missilej+1 in(0)
    j++
}
```

Figure 7 ADU couplings pseudo code

3.3.1. Detection Radar

The radar detection component consist of one input port that receives UAV's states, another input port that alerts if any tracking radar has lost its target, and one output port per each tracking radar model to transmit the UAV state to follow.

The radar operation is based on its inputs, stores as an internal state variable a collection that maps the assignment between incoming UAVs to tracking radars. When the detection radar receives one or more UAV's states, first, attempts to discover them in its visibility field, if they are within its range and according to a certain probability of detection, checks if they have not been already assign to any tracking radar, and then, in that case, searches for a task free tracking radars to send them, and sets sigma to a certain response time. Otherwise, when it receives notification of a lost target removes the mapping relationship from memory.

3.3.2. Tracking Radar

The tracking radar DEVS based model is design to operate as follows. Through the input port linked to detection radar of the corresponding ADU, obtains the state of a UAV to track, stores its value as an internal state variable and waits for the reception of the same UAV state from the coupling wired to UAV's models. Then, verifies whether the UAV is within its detection field, if it fails and the elapsed time doesn't exceed the defined maximum, estimates its position, orientation and velocities and finally sends the UAV state to the first model of the series of missiles. Otherwise, reports to the Detection Radar that the target has been lost.

3.3.3. Missile

Missiles models are composed by one input port that accepts states of UAVs intended to be blown down. One output port to give over the UAV state to the next missile only if their status is "fired". And another output port to communicate its state to the UAVs so they can check whether they are destroyed or not. Like UAVs, they also keep an internal state variable with an array of missile states (same as UAVs) reflecting changes in time of the missile dynamics.

Essentially, as seen on Figures 8 and 9, missiles wait for an external command from any tracking radar model of its corresponding ADU to shift from the initial state stop to be fired. Then, sigma is updated from infinity to the next immediate state time. Afterwards, every time the internal time event function is triggered, jumps to the next computed state. Sigma is updated to the next computed time, unless the array of states is empty, reached its goal, or exceed limits or distance sigma is set to ∞. This behavior is exemplified by Figures 8 and 9.

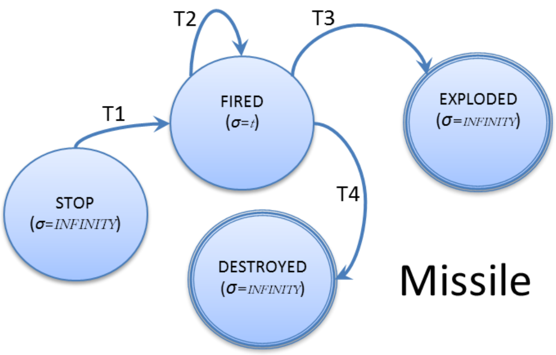


Figure 8 Missile State Diagram

⇨	FIRED	EXPLODED	DESTROYED
STOP	T1		
FIRED	T2	T3	T4

	δ_{int}	δ_{ext}
T1		*NO assigned target*
T2	*states is NOT empty*	*assigned target*
T3		*reached goal*
T4		*exceed limits or distance*

Figure 9 Missile State Transitions

4. SIMULATION

In order to evaluate the quality of the planner before using it in real missions, we decide to validate the routes in multiple experiments against a simulator that contains models for all the elements of the problem.

The models of the radars and missiles are non-deterministic, incorporating stochastic behaviors related with the probability of detection and destruction of the UAVs. So, two simulations for the same experiment and optimal trajectories can return different results.

The simulation symbolizes a scenario where one or more unmanned aerial vehicles must follow a given trajectory trying to avoid flying within ADUs visibility range. Trajectories are calculated prior to the simulation, each trajectory consists of a sequence of way points, each way point may also be a getaway, i.e., an intersection between the current trajectory and an alternative trajectory intended to be used in evasion maneuvers. Whenever an UAV apprehends that is been detected by an ADU, starts if possible an evasion maneuver to escape from the ADU fire power range and prevent from being shot down. Correspondingly, every time an ADU detects an UAV, after a specified period of time, shoots a missile to attempt to knock down the UAV.

4.1. Experiments

The experiment consists of 3 UAVs and different number of ADUs and NFZs. They also differ in the initial and final positions of the UAVs, in the position of the ADUs and NFZs, and in the number of initially known ADUs. There are 4 different types (A, B, C, and D) schematized in Figure 11. The initial and final positions of each UAV are represented as green and magenta crosses. The yellow crosses in experiment type C represent intermediate points that the UAVs are forced to visit. The ADUs are represented by the big blue dashed circles which show the maximum distance of detection of their radars, and by the small red solid circles which enclose the zones where the probability of destroying an UAV can be greater than 0. The NFZs are represented with the rectangular green areas. For this experiment, the offline path planner is configured to consider no ADU's and the initial routes are only by NFZs, terrain and UAVs maneuverability.

Figure 11 represent the 4 experiment types (A, B, C and D), the offline routes for the experiments, and one of the online alternative routes.

4.2. Results

In this section we analyze the results of the aforementioned experiments. For each of the 4 cases, we carry out 30 simulations. Based on their results, we measure the consistency of the simulations. The consistency is characterized by the percentage of successful arrivals of each UAV in each of the 4 cases during the 30 simulations. The performance is measured according with the total simulation time needed in each of the 4 cases.

Table 1 Percentage of success for each example type

	UAV_1	UAV_2	UAV_3
A	100%	100%	10%
B	100%	100%	30%
C	100%	77%	100%
D	13%	100%	17%

The results presented in Table I depend on the UAV, experiment type (A, B, C, or D). For instance, if we focus on the experiment B (second row of Table I), UAV_2 always survives because its initial trajectory is always safe, UAV_3 has a good chance to be destroyed because its initial trajectory stays in the non-safe zone too long. Similar explanations apply to the rest of the UAVs in the remaining experiments.

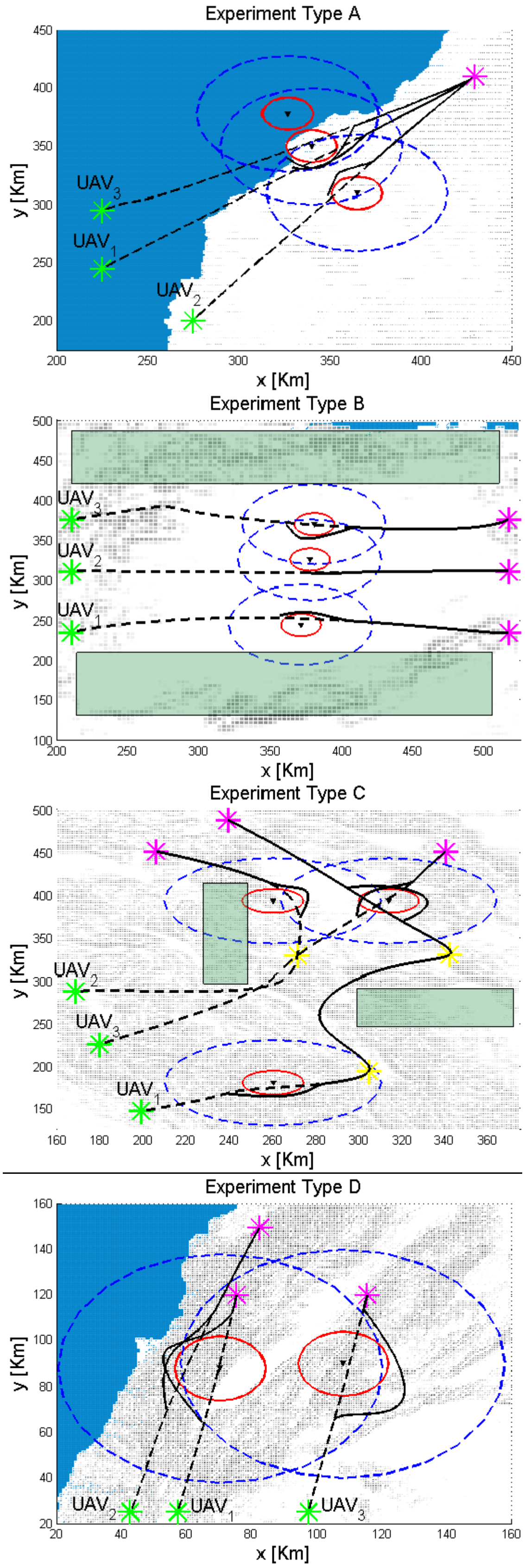


Figure 10 Experiments A, B, C and D

5. CONCLUSIONS

The work presented in this document takes advantage of DEVS simulation framework to build models that simulate a complex military problem. The simulator validates successfully the results of a route planner for multiple UAVs. Builds accurate models to verify the computed solutions of the path planner obtained through several approximations.

ACKNOWLEDGMENTS

This work is supported by the Spanish grants DPI2009-14552-C02-01 and DPI2009-14552-C02-02.

REFERENCES

Stevens, Brian L., and Lewis, Frank L., 2004. *Aircraft Control and Simulation*. 2nd ed. Wiley.

Besada-Portas E., de la Torre L., Moreno A., and Risco-Martín J.L., *Performance Analysis of Multiobjective Bio-inspired UAV Path Planners*. In GECCO 2011.

Rasmussen S. and Chandler P., 2002. Multiuav: a multiple uav simulation for investigation of cooperative control. *Proceedings of the Winter*, vol. 1, pp. 869–877.

Glickstein I. and Stiles P., 1992. Situation assessment using cellular automata paradigm. *Aerospace and Electronic Systems Magazine*, IEEE, vol. 7, no. 1, pp. 32–37.

Shem A., Mazzuchi T., and Sarkani S., 2008. Addressing uncertainty in uav navigation decision-making. *Aerospace and Electronic Systems, IEEE Transactions*, vol. 44, no. 1, pp. 295 –313.

Holman K., Kuzub J., and Wainer G.A., 2010. Uav search strategies using cell devs. *Proceedings of 2010 Spring Simulation Conference*, ANSS Symposium, pp. 192–199.

Lundell M., Tang J., Hogan T., and Nygard K., 2006. An agent-based heterogeneous uav simulator design. *Proceedings of the 5th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases*. WSEAS, pp. 453–457.

Karim S., Heinze C., and Dunn S., 2004. Agent-based mission management for a uav. *Intelligent Sensors, Sensor Networks and Information Processing Conference*, pp. 481 – 486.

Kim J.H, Kim T.G., and Jeong J., 2007. Embedding devs methodology in cbd process for development of war game simulators. *Proceeding of summer computer simulation conference*, pp. 35:1–35:8.

Bernard P. Zeigler and T. Kim and H. Praehofer, 2000. *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*. Academic Press.

Kofman E., 2004. Discrete Event Simulation of Hybrid Systems. *SIAM Journal on Scientific Computing*, 25(5), 1771-1797.

AUTOMATIC MARITIME SURVEILLANCE WITH VISUAL TARGET DETECTION

Domenico Bloisi[(a)], Luca Iocchi[(b)], Michele Fiorini[(c)], Giovanni Graziano[(d)]

[(a)(b)] Department of Computer and System Sciences Sapienza University of Rome
[(c)(d)] SELEX Sistemi Integrati S.p.A. - Roma

[(a)]bloisi@dis.uniroma1.it, [(b)]iocchi@dis.uniroma1.it, [(c)]mfiorini@selex-si.com, [(d)]ggraziano@selex-si.com

ABSTRACT
In this paper an automatic maritime surveillance system is presented. Boat detection is performed by means of an Haar-like classifier in order to obtain robustness with respect to targets having very different size, reflections and wakes on the water surface, and apparently motionless boats anchored off the coast. Detection results are filtered over the time in order to reduce the false alarm rate. Experimental results show the effectiveness of the approach with different light conditions and camera positions. The system is able to provide the user a global view adding a visual dimension to AIS data.

Keywords: coastal surveillance, data fusion, border control, territorial waters.

1. INTRODUCTION
The increasing requests for safety, security and environmental protection, reveal the deficiency of the classic on-shore traffic management centres to satisfy the needs and requirements of the maritime domain. The traditional devices (Aids to Navigation, AtoN) and sensors (radar and Automatic Identification System, AIS) based on microwave tubes are starting to be replaced by solid-state equipments (Amato, Fiorini, Gallone, and Golino 2010) .

The centres are, as often as not, equipped with long range surveillance cameras enslaved to the radar to enhance the early recognition of targets. This approach is used in many Vessel Traffic Services (VTS) systems installed by SELEX Sistemi Integrati in Europe and abroad. It is useful both for monitoring vessels traffic, increasing safety at sea and facing illegal and hostile activities along the coastline.

SELEX Sistemi Integrati installed VTS and coastal systems in Yemen, Turkey, Serbia (fluvial), Panama, Poland, Russia (St. Petersburg), China and Italy.

In the VTS context, the target classification is done automatically by the system only for the cooperative targets equipped with AIS, while remaining targets should be identified manually by the operator.

Cameras are used for monitoring and providing pictures of the tactical situation inland the Territorial Waters (12 nautical miles).

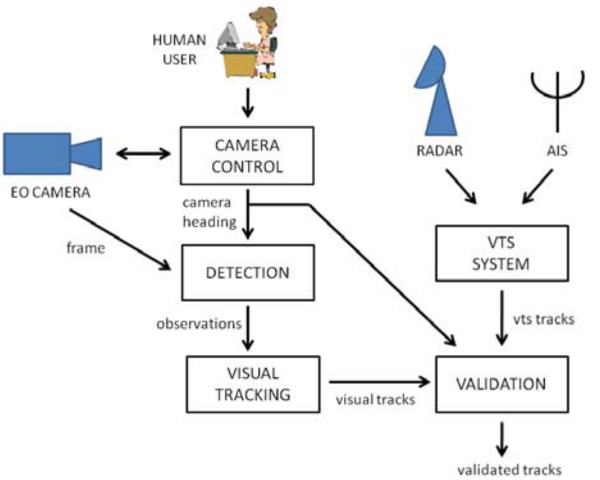


Figure 1: Architecture of a VTS/Coastal Surveillance Site

The cameras can operate in manual (under the operator control) or automatic (enslaved to a radar track) mode, but are not used for target validation, i.e., identity verification of AIS targets and classification of non-AIS targets.

The main contributions of the paper concern the description of a complex implemented system and the evaluation of its critical components. In particular, the detection approach is specifically designed in order to deal with the characteristic of the scenario.

The paper is organized as follows: after discussing related work in Section 2, the general architecture of the system is described in Section 3. In Section 4 we detail the visual detection module, while Section 5 shows quantitative results obtained by our approach on real data. Section 6 provides conclusions and future directions.

2. RELATED WORK
Maritime environment represents a challenging scenario for automatic video surveillance due to gradual and sudden illumination changes (e.g., clouds), motion changes (e.g., camera jitter), high frequency background objects (e.g., waves, raindrops) and reflections.

Although it is possible to develop systems based on electro-optical (EO) sensors only (e.g., ARGOS (Bloisi and Iocchi 2009)), the use of additional sensors such as infrared (IR), radar, Global Positioning System (GPS), and AIS can reduce the false positive detections thus improving the performance.

SeeCoast (Rhodes, Bomberger, Seibert, and Waxman 2006) is a system for coastal surveillance using EO and IR cameras, radar, and AIS. The detection is carried out by estimating the motion of the background and segmenting it into components. However, motion-based vessel detection can experiment difficulties when a boat is moving directly toward the camera or is anchored off the coast due to the small amount of inter-frame changes.

ASV (Pires, Guinet, and Dusch 2010) is an automatic optical system for maritime safety using IR, GPS, and AIS. To detect relevant objects, the sea area is segmented and its statistical distribution is calculated. Any irregularities from this distribution are supposed to correspond to objects of interest. However, due to wakes, such an approach can produce false positives.

A method for visual surveillance in maritime domain with non-stationary camera installed on an untethered buoy is presented in (Fefilatyev, Goldgof, and Lembke 2010). After the detection of the horizon line, a color gradient filter is applied to obtain a gray-scale image with intensities corresponding to the magnitude of color changes. Detection of the objects of interest is performed through thresholding of such gray-scale image into a binary map. The algorithm is limited by the assumption that all marine targets are located above the horizon line.

An object detection system for finding ships in maritime video is detailed in (Wijnhoven, van Rens, Jaspers, and de With 2010). The used approach is based on the Histogram of Oriented Gradients (HOG) (Dalal and Triggs 2005). Since the calculation of the detection features involves a significant amount of computational resources, real-time performance can be obtained only by means of hardware acceleration with programmable components such as FPGAs.

Maximum average correlation height (MACH) filters are employed for vessel classification in (Rodriguez Sullivan and Shah 2008). Vessel detections are cross-referenced with ship pre-arrival notices in order to verify the vessel's access to the port. As reported by the authors, such an approach tends to misclassify small vessels like speed boats and fishing boats.

The analysis of the literature shows a series of criticisms related to the port scenario:

- the use of Pan-Tilt-Zoom (PTZ) cameras;
- the presence of targets having very different size;
- reflections and wakes on the water surface;
- the presence of apparently motionless boats anchored off the coast.

Our approach takes into account all of the above mentioned problems.

Figure 2: Boat detection

3. SYSTEM OVERVIEW

The general architecture of the system is depicted in Fig. 1.

3.1. Camera Control Module

The EO camera is a PTZ FLIR SR-TV with a 26x optical zoom, that can be moved by a human user through a control module. Furthermore, the control module is able to provide camera orientation and field-of-view.

3.2. Detection Module

The detection module takes as input the current frame acquired by the camera and the current heading of the camera. It is the most critical part of the system, since detection accuracy must be as high as possible while maintaining an acceptable computational load. The output of the detection module is a list of observations.

Each observation is a bounding box representing a detected boat (Fig. 2). All the details about the detection method are reported in Section 4.

3.3. Visual Tracking Module

The visual tracking module principal role concerns the temporal filtering of the false positives. The output of the tracking module is a set of visual tracks, i.e., bounding boxes with an identification number as in Fig. 3. Only tracks that present a sufficient number of observations are considered of interest (we set this threshold to 10).

The association between tracks and observations is made on the basis of a nearest-neighbor policy with the Bhattacharyya distance between the HSV value histograms of the track and observation as measure.

We experimented also a CamShift (Bradski 1998) based approach obtaining poor results, probably due to the fact that CamShift creates color models taking histograms only from the hue channel in the HSV space which alone is not discriminative in the maritime scenario.

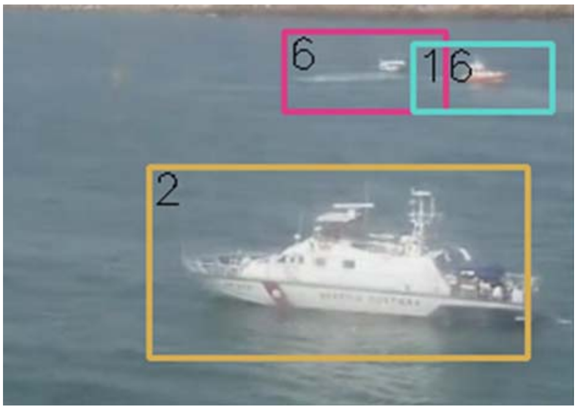


Figure 3: Visual tracking

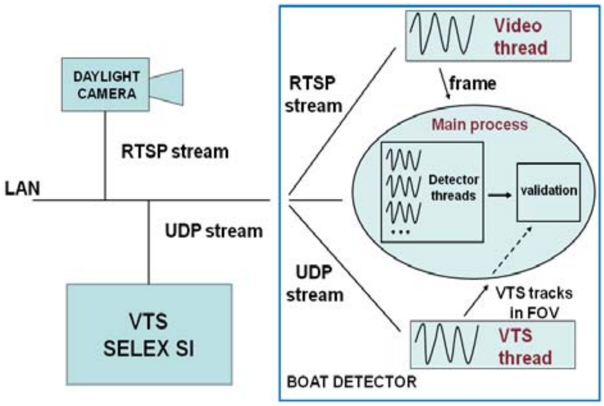


Figure 5: Communication between modules.

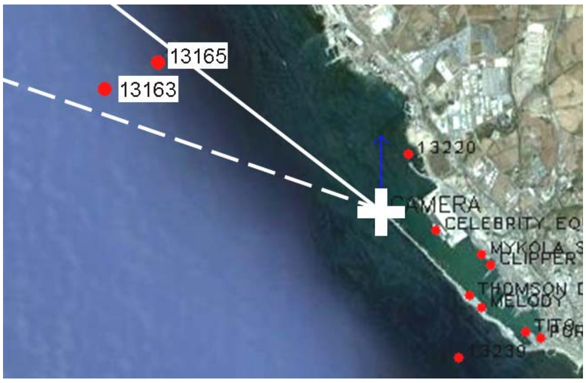


Figure 4: Geographic projection of the VTS tracks. The camera (white +) is located at the entrance of the port

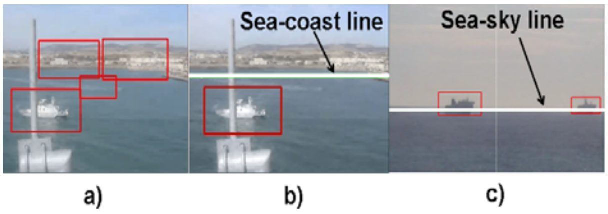


Figure 6: Sea-sky line and sea-coast line.

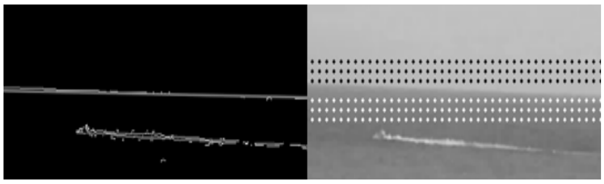


Figure 7: Sea-sky line detection

3.4. VTS Systems

A Coastal Surveillance Site (CSS) is generally responsible for detecting and tracking vessels in a range up to 12 nautical miles (NM), the defined “Territorial Waters” (United Nation 1982). Radar information are merged with AIS data in order to obtain a geographic view (see Fig. 4).

3.5. Validation Module

The validation module aims to give the user a real-time visual image for the tracks (Fig. 8), which is not available for AIS and non-AIS targets.

Data fusion between video and radar data is performed on a probabilistic base. VTS and visual tracks are projected onto a two dimensional common space in order to perform the association (see Fig. 9c). For the video data, the first dimension (x) is the distance (in pixels) of the bounding box from the left margin of the frame and the second one (y) is the distance from the bottom of the frame. For the radar data, the distance (in pixel) of the VTS track from the left side of the field of view and the distance from the camera position in the geographic projection represent the x and y dimensions respectively.

Since the video frame and the geographic view present different scales, the dimensions are normalized with respect to the common space width and height. The projected visual and VTS tracks are associated on the basis of a nearest-neighbor policy.

The proposed data fusion scheme is justified by the fact that the boats are moving on a planar surface, thus a boat closer to the camera than another will appear in a lower position in the image. This is the case depicted in Fig. 8, corresponding to the geographic view in Fig. 4.

In Fig. 9 the output for the detection (Fig. 9a), the visual tracking (Fig. 9b), and the data fusion (Fig. 9c) in case of partially overlapping targets are shown.

3.6. Inter-module Communication

The inter-module communication scheme is reported in Fig. 5. The VTS system, the EO camera, and the detector are connected through a LAN. Video frames are transmitted by the camera using RTSP, while VTS data are transmitted by the SELEX Sistemi Integrati’s VTS system via UDP.

The detector is made of a main process managing the detection algorithm (performed by a set of threads in order to speed up the detection) and the validation module and two threads managing the different refresh rates of the incoming data. Indeed, video data are transmitted at 25 fps, VTS data have a refresh period of about 2 seconds, and the detector is able to compute 10 fps (see Section 5).

In the rest of the paper we analyze in detail the detection algorithm, that represents the most innovative part of the system.

Table 1: Visual detection results

Classifier	Coastline Detection	DR	FAR
Without wake examples	NO	0.892	0.475
Without wake examples	YES	0.892	0.265
With wake examples	YES	0.928	0.251

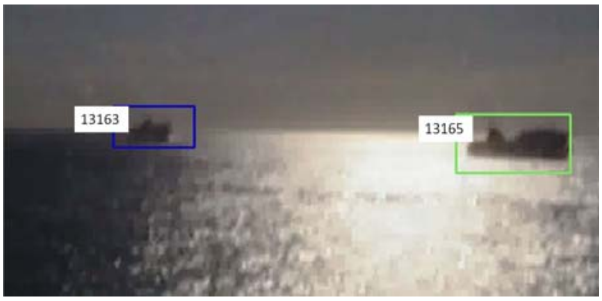


Figure 8: Validated tracks. Visual and tracking data are fused in a single view

4. VISUAL DETECTION

The detection module aims to find the boats in the current frame obtained from the PTZ camera. Since the camera is frequently moved by the user, a foreground/background modelling approach to detect vessels is ineffective. Thus, we decided to adopt a classifier based detection. In order to obtain real-time performance, computationally expensive methods (e.g., (Dalal and Triggs 2005; Tuzel, Porikli, and Meer 2008)) were discarded and a Haar-like features based approach (Viola and Jones 2004) has been adopted. Indeed, its main advantages are the computational speed and the possibility of combining different classifiers in a cascade.

However, Haar-like classifier has been originally designed for face recognition, thus we verified the applicability of the method for boat detection. At this aim, we used the OpenCV HaarTraining functions.

A set of 4000 images not containing boats and a set of 1500 images depicting different types of boats taken from the internet have been used as input for the offline training stage obtaining a 24 level classifier (the training was stopped when the false alarm rate reached 5×10^{-6}).

Given a single frame, the boats in the image are detected. Along with the boats, also the limit of the sea surface is detected. Depending by the heading of the camera, the system differentiates between sea-coast line (Fig. 6b) and sea-sky line (Fig. 6c).

Since in presence of the coast the probability of finding false positives increases (top of Fig. 6a), it is possible to filter out the erroneous detections laying above the sea-coast line (Fig. 6b).

In order to detect the limit of the sea surface, the Hough transform is applied to the edge map of the frame and the candidate lines are validated with respect to a set of points belonging to a rectangular region containing the line (right part of Fig. 7). If at least the 90% of the corresponding points above and under the line present different intensity values, then the line is considered valid. In this way, it is possible to filter out misdetections due to long wakes (left part of Fig. 7).

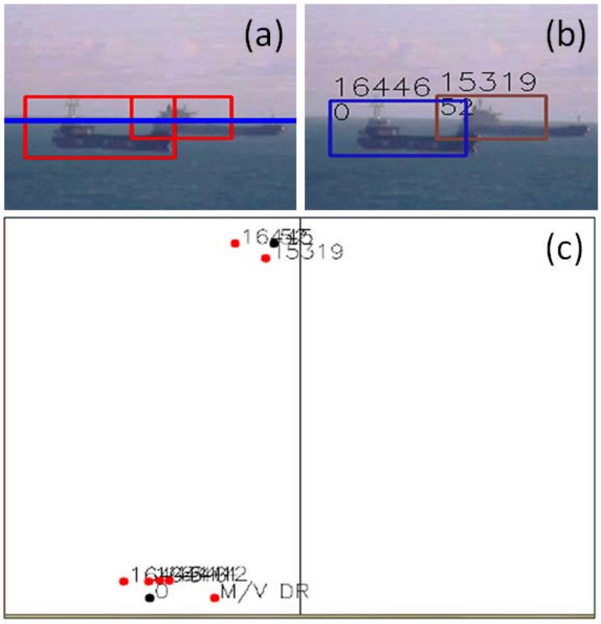


Figure 9: Visual analysis can help in correcting erroneous detection made by the radar. Detection results (a) are filtered over the time to obtain visual tracks (b) that are projected over a common space together with radar and AIS tracks (c)

5. RESULTS ON VISUAL DETECTION

A set of 100 randomly chosen images taken from 11 videos recorded with different light conditions and camera positions has been used to test the accuracy of the Haar detection.

To filter out false positives due to wakes (e.g., in the middle of Fig. 6a) and reflections (e.g., in Fig. 8), an additional weak-classifier has been created by means of a negative set made of 4000 images of wakes and other false positive detections obtained by the original 24 level classifier. The results are reported in Table 1, in terms of detection rate (DR) and false alarm rate (FAR)

$$DR = \frac{TP}{TP + FN} \qquad FAR = \frac{FP}{TP + FP}$$

where TP are the correctly detected boats, FN is the number of not detected boats, and FP are the incorrect detections.

Visual tracking and data fusion modules can drastically reduce the FAR as well as improve the DR of the whole system thanks to temporal filtering and radar data. However, it is fundamental that frame-by-frame detection provides reliable results.

Visual analysis can help also in correcting radar errors. In presence of big targets (e.g., oil tankers in Fig. 9), it is possible to obtain multiple radar tracks for the same object (red dots in Fig. 9c). Those multiple radar detections can be merged if a single visual track has been individuated (the black dots in Fig 9c referring to the visual tracks in Fig. 9b). The computational speed for the system using an Intel Core 2 Duo SU7300 CPU, 4 GB RAM is 10 fps computing 320x240 images.

6. CONCLUSIONS

In this paper an automatic video surveillance system for maritime environment has been presented.

The system is able to deal with user defined movements of the cameras, targets having very different size, reflections and wakes on the water surface, and apparently motionless boats anchored off the coast.

The main goal of the system is to provide the user a global view of the situation at hand adding a visual dimension to AIS data.

The results on real data show the effectiveness of the proposed detection approach maintaining a 10 fps computational speed.

As future work, we intend to complete the evaluation of the whole system (that is currently underway) and to add the IR data to the data fusion scheme in order to further improve the performance of the system.

REFERENCES

[1] Amato, F., Fiorini, M., Gallone, S., Golino, G., 2010. Fully Solid State Radar for Vessel Traffic Services. *Proceedings of International Radar Symposium (IRS)*, pp. 1-5. June 16-18, Vilnius, Lithuania.

[2] Bloisi, D., Iocchi, L., 2009. Argos - a Video Surveillance System for boat Traffic Monitoring in Venice. *IJPRAI*, vol. 23 (7), pp. 1477-1502.

[3] Bradski, G. R., 1998. Computer video face tracking for use in a perceptual user interface. *Intel Technology Journal, Q2*.

[4] Dalal, N., Triggs, B., 2005. Histogram of oriented gradients for human detection. *Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, pp. 886-893. June 20-26, San Diego, CA, USA.

[5] Fefilatyev, S., Goldgof, D.B., Lembke, C., 2010. Tracking Ships from Fast Moving Camera through Image Registration. *Proceedings of the Int. Conf. on Pattern Recognition*, pp. 3500-3503. August 23-26, Istanbul, Turkey.

[6] Pires, N., Guinet, J., Dusch, E., 2010. ASV: An Innovative Automatic System for Maritime Surveillance. *Navigation*, vol. 58 (232), pp. 1-20.

[7] Rhodes, B.J., Bomberger, N.A., Seibert, M., Waxman, A.M., 2006. SeeCoast: Automated Port Scene Understanding Facilitated by Normalcy Learning. *Proceedings of IEEE Conf. on Military Communications*, pp. 3656-3662, October 23-25, Washington, DC, USA.

[8] Rodriguez Sullivan, M.D., Shah, M., 2008. Visual surveillance in maritime port facilities. *Proc. SPIE 6978*, pp. 697811-697811-8.

[9] Tuzel, O., Porikli, F., Meer, P., 2008. Pedestrian Detection via Classification on Riemannian Manifolds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30 (10), pp. 1713-1727.

[10] Viola, P., Jones, M.J., 2004. Robust Real-Time Face Detection. *Int. Journal of Computer Vision*, vol. 57 (2), pp. 137-154.

[11] Wijnhoven, R.G.J., van Rens, K., Jaspers, E.G.T., de With, P.H.N., 2010. Online Learning for Ship Detection in Maritime Surveillance. *Proceedings of the 31st Symposium on Information Theory in the Benelux*, pp. 73-80. May 11-12, Rotterdam, The Netherlands.

[12] United Nation (UN), 1982. Convention on the Law of the Sea: Territorial Sea and Contiguous Zone.

AUTHORS BIOGRAPHY

Dr. Domenico Bloisi received his Ph.D., M.Sc., and B.Sc. degrees in computer science engineering from Sapienza University of Rome, Italy in 2010, 2006, and 2004 respectively. He is currently a post-doc in the Department of Computer and System Science, Sapienza University of Rome, Italy. His main research interests are related to object detection, visual tracking and data fusion in automatic video surveillance.

Dr. Luca Iocchi received his Master (Laurea) degree in 1995 and his Ph.D. in 1999 from Sapienza University of Rome. He is currently Assistant Professor in the Department of Computer and System Science, Sapienza University of Rome, Italy. He is the author of more than 100 referred papers in international journals and conferences. His main research interests are in the areas of cognitive robotics, action planning, multi-robot coordination, robot perception, robot learning, stereo vision and vision based applications.

Dr. Michele Fiorini was educated at Ancona (Italy) and Bath (UK) Universities, receiving his PhD in 2006, and is now a senior systems engineer at SELEX Sistemi Integrati, a Finmeccanica company, in Rome, Italy. His working interests are mainly concerned with command, control, communications, computers, intelligence, surveillance and reconnaissance (C4ISR), homeland protection and vessel traffic service (VTS) as well as in electromagnetism theory and antenna design.

Giovanni Graziano received his Master (Laurea) degree in 2007 from Federico II University, Naples, Italy. Since 2007 he is a systems engineer at SELEX Sistemi Integrati, a Finmeccanica company, in Rome, Italy. His working interests are related to localization systems and tracking algorithms for radar and electro-optical sources.

CAPRICORN: Using Intelligent Agents and Interoperable Simulation for supporting Country Reconstruction

Agostino G. Bruzzone, Marina Massei, Alberto Tremori
MISS DIPTEM University of Genoa
Via Opera Pia 15, 16145 Genova, Italy
Email {agostino, massei, tremori}@itim.unige.it - URL www.itim.unige.it

Enrico Bocca
Mast srl
Via Biga 26, 16144 Genova, Italy
Email enrico.bocca@mastsrl.eu - URL www. mastsrl.eu

Francesca Madeo, Federico Tarone
Simulation Team
Via Cadorna 2, 17100 Savona, Italy
Email {madeo,tarone}@simulationteam.com - URL www.simulationteam.com

ABSTRACT
This paper proposes the development of conceptual models to used in a simulator for supporting operational planners in country reconstruction within complex scenarios affected by natural disasters, asymmetric warfare, crisis etc. The authors developed for this context a a special kind of IA (Intelligent Agents) driving CGF (Computer Generated Forces) able to consider the human aspects as well as the impact of social networks on this context as further development of their IA-CGF. In this paper the authors propose the general architecture of the simulator objects, in their structures, attributes and methods as required to model this kind of applications.

INTRODUCTION

Currently the existing mission environments propose new challenges to military operational planning; it emerge that there is no so much experience and/or support tools to support decision makers and not consolidated simulation models; in fact the current scenarios are characterized by new kinds of threats (i.e. terrorist attacks, information warfare etc.), new kinds of operations (such as country reconstruction peace keeping or stabilization and normalization operations) and they involve several not conventional actors both military and civilian (such as humanitarian organization, insurgents, guerrilla forces, terrorists, hackers, media warriors): due to this fact it emerge that a special attention need to be dedicated to operations that have a strong impact on the local population such as CIMIC (Civil Military Cooperation), PSYOPs (Psychological Operations), Country Reconstruction, Disaster Relief, Crisis Response Operations etc (Bruzzone, Massei, Caussanel 2006; Bruzzone et al.2010; Galula 2006).

The paper focuses on a new generation of CGF able to incorporate IA for reproducing the population behavior combining social network and individual reaction to the country reconstruction activities; by this approach it becomes possible to simulate complex scenarios and to evaluate the most effective decision to succeed considering the reaction/support of the population; obviously this aspect becomes even more critical in stabilization or normalization scenarios.

The authors developed this research within the CAPRICORN project (CIMIC and Planning Research In Complex Operational Realistic Network), sponsored EDA (European Defense Agency), for the development of capabilities in the complex and critical sector of Military Operation Planning, specifically for Not-Article5/San Petersburg Operations (Bruzzone, Frydman, Tremori 2009); in fact the CAPRICORN project get benefits of previous researches carried out by Simulation Team (Avalle et al.1999; Tremori et al. 2009; Bruzzone, Tarone 2011); in particular among the first application in defense of these concept the authors developed a first generation of Intelligent CGF and Objects within in the framework of a previous R&D Project (Research and Development) named PIOVRA (Polyfunctional Intelligent Operational Virtual Reality Agents) (Bruzzone, Massei 2007;). The model and results presented in this paper are unclassified, therefore any sensible information and data have been modified due to their confidential nature.

COUNTRY RECONSTRUCTION AND OPERATION AFFECTING THE LOCAL POPULATION: A COMPLEX SCENARIO

The success of country reconstruction deals with the capability to properly handle the relations with the population and local authorities (Bruzzone et al.2011); therefore there is currently a lack of resources for simulating these operation that requires deep capabilities in modeling and multidisciplinary approach; the authors are working currently to develop specific simulation tiles to cover these aspects such as CIMIC, PSYOPS and Country Reconstruction (Bruzzone, Frydman, Tremori 2009; Bruzzone, Madeo, Tarone 2010).

For instance CIMIC operations refer to an operative function that, combined with other operational areas (i.e. Command and Control, combat, support to combat, intelligence, logistics support etc) enable to support harmonic and effective development of combined operations focusing on the interaction among military forces and civil components in crisis areas; usually the main CIMIC objectives are (AJP 9):

- Facilitate the relation between military and civilians
- Create synergies among governmental/international projects and those carried out by NGOs
- Create an interface between the command and the civil environment
- Prepare favorable conditions for operations on the area

Another interesting kind of activity is related to PYSOPS: these operations refer to Psychological Activities planned during peace, crisis or war and directed towards target groups of enemies, friends or neutrals in order to influence attitudes and behaviors that are related to their capability to achieve political and military objective; the main goal of PSYOPS is win the mind and the heart of local population by neutralizing enemy influence and by promoting coalition forces actions (AJP 3.10.1; Chauvancy 2011).

These contexts are very complex due to the presence of several actors (both civilians and military forces) affected by many stochastic components, due to the wide number of interactions among the different actors and due to the strong relevance of human behaviors and attitudes on military actions impacts and effects. As matter of fact in such contexts scenario perception and emotional (i.e. fear, stress) and social aspects are very critical (Amico et. al 2000).

Therefore Modeling and Simulation (M&S) is one of the most effective approaches in order to model unconventional environments and frameworks (Bruzzone, Madeo, Tarone 2010).

In fact for properly apply M&S to this sector of operations, it becomes critical to model Human behavior ModifiersBruzzone (HBM) (Frydman, Cantice, Massei, Poggi, Turi 2009).

Modeling Complex Scenario such as CIMIC and Country Reconstruction requires specific types of models:

- Civilian Attitude Model for describing changes in people attitude
- Social Network Model for simulating people interactions and information and feelings spreading.
- Economic Model to reproduce economic development effects and impacts on the entities attitude.

In particular the authors are involved in the generation of a new intelligent agent driven simulator to support operational planning as well as training for these applications based on the IA-CGF (Intelligent Agents Computer Generated Forces) developed by the Simulation Team. In fact one of the main issues is the development of new CGF (Computer Generated Forces) that guarantee more realism and major fitness to the real operational planners needs by reproducing human behaviors and cultural and social aspects (i.e. religious faith, beliefs, values, education, etc.) (Bruzzone, Massei, Caussanel 2006). The authors achieved a successful result with the development of PIOVRA (Poly-functional Intelligent Operational Virtual Reality Agents) Intelligent Agents. In fact new CGFs, able to simulate "Intelligent" behavior were developed (Bruzzone et al. 2004); as further steps the authors and the Simulation Team complete the creation of IA-CGF Libraries, Units and Non Conventional Frameworks (NCF) (Bruzzone 2008).

In this paper the authors present conceptual models to be implemented in CAPRICORN Simulator based on IA-CGF (Intelligent Agents Computer Generated Forces) for country reconstruction considering support operations related to CIMIC and PSYOPS. The proposed solution evolve in an executable HLA Federation (High Level Architecture) and their specific models result able to operate as federate in other federations as well as stand alone simulator. In fact the use of IA-CGF modules allow to consider both psychological and social aspects and to have agents in the interoperable simulation able to react based on their situation awareness and on their current and previous status.

CAPRICORN PROJECT OVERVIEW

As anticipated CAPRICORN is an EDA Research & Development Project devoted to develop capabilities in the complex and critical sector of Military Operation Planning, specifically for Not-Article5/San Petersburg Operations, by using Simulation & CGF (Computer Generated Forces) based on Intelligent Agents (IAs). In order to be effective in operational planning, CAPRICORN integrates new advanced models, developed by the authors that are able to operate stand alone or integrated within an HLA Federation; this guarantee that CAPRICORN will have a potential both for users interested in operational planning decision support (usually short term and limited resources within real mission environments) and for being integrated with other systems for experimentation over complex scenarios.

The main outcomes of CAPRICORN Project are:

- Investigation on the use of Intelligent CGF and Simulation for Operational Planning
- Advanced Human Behavior Hierarchical Models for Intelligent Agents and CGF
- Enhanced Interoperable Multilevel Models for Training & Operational Planning
- Modeling CIMIC & PSYOPS for Interoperable Simulation

Among the different researches carried out by tThe previous the authors in agent driven simulation and intelligent agent development an interesting experience is related to PIOVRA Project that allowed to develop a first Generation of new CGF able to simulate "Intelligent" behavior in order to model friends, foe (including terrorists), neutrals forces, but especially civilians that represent an additional category reacting not based on their party attribute (Friend, foe, civil, neutral), but based on the perception of the situation (Caussanel et al.2007). In fact in particular, PIOVRA project was focused on modeling neutral units representing civilians and their specific behaviors and logic in civil disorders. PIOVRA focuses on the definition of conceptual models able to simulate a co-operative behavior of PIOVRA CGF allowing their aggregation or separation depending on the situation, clearly keeping in mind the command hierarchies for military units and managing their dynamic evolution during actions.

A very critical advance in this area was provided by the new additional elements, developed by the Simulation Team in term of RATS (Riots, Agitators & Terrorists by Simulation) and IA-CGF Modules; these modules was articulated in order to have a structure of human behavior libraries and unconventional units and to create ad hoc non conventional frameworks to simulate specific cases (i.e. IA-CGF Earthquake to reproduce humanitarian support during Haiti Earthquake in term of food distribution analysis) (Tremori et al.2009; Bruzzone, Massei 2010); for instance in IA-CGF is included the capability to reproduce social network representing families and working connection, creating dynamics interactions among people and among groups of people with their own sociological and cultural characteristics (Bruzzone et al.2009).

By moving from Intelligent Agents developed in PIOVRA to IACGF, the authors create new specific CGF that, after their integration with innovative models are able to cover critical aspect (i.e. Human Behavior Modifiers and Cultural Issues); these agents are able to support CIMIC and PSYOPs operations and they represent the building blocks for CAPRICORN Demonstrator, the federation devoted to experiment with users and scientists the capabilities of these agent driven simulation in supporting operational planning and training in this area.

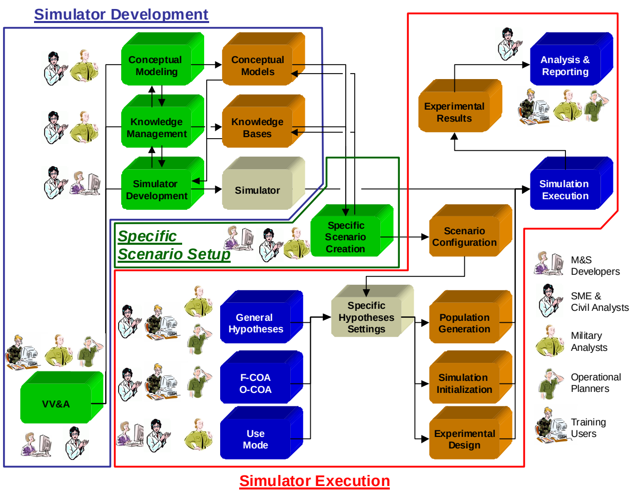


Figure 1: Processes and Key Elements

CAPRICORN DEVELOPMENT, CONFIGURATION AND EXECUTION

The general process to develop CAPRICORN are based on the procedures synthesized in figure 1.; in fact the The developing processes include activities distributed among three main phases: simulation development, specific mission environment setup and simulation experimentation over the mission environment environment. The proposed scheme allows to define a dynamic approach to reuse CAPRICORN developments even in further scenarios or over extended application domains.

The processes include the following functional activities:

- Modeling: this function is devoted to create new models able to reproduce specific phenomena, actions and element. In fact, in the case of CAPRICORN this focuses mostly on modeling population behavior as well as CIMIC/PSYOPs actions and their interactions in the first development phase, but it will be extended for further applications in the future. In this function are involved simulation experts therefore it is run in strong contact knowledge management; the main goal related to this function is the creation of the Conceptual Models.

- Knowledge Management: this function is focusing on acquiring and organizing the knowledge relevant to the specific problems; in CAPRICORN, SME (Subject Matter Experts) in operational planning, CIMIC as well as military simulation centers and software science experts interact with scientists operating in knowledge management as well as in M&S. The goal of this activity is to create a set of knowledge bases, where each one is devoted to contain specific information concerning a particular region/context; in this case the human behavior model configuration as well as characteristics of the humans (i.e. population) need to be properly tuned
- Simulators Development: in this process the simulators are implemented involving all the related components; the work is carried out mostly by simulation developers.
- VV&A: Verification, Validation and Accreditation run concurrently the all the processes in order to guarantee that the conceptual models, knowledge bases are consistent and correct as well as the fact that the user needs are satisfied by proper implementation solution; this activity requires continue and effective communications among users, scientists and developers.
- Specific Mission Environment Creation: this process start when it becomes necessary to create a specific instance for a region/scenario; during this phase a set of specific models has to be instantiated starting from conceptual models and knowledge bases available in simulator implementation. By this approach it will be possible to create a specific mission environment (i.e. Afghanistan, Lebanon, Ivory Cost) even getting benefits of previous available CAPRICORN scenarios.
- Specific Hypotheses Setting: during this phase the planners and/or analysts are allowed to define the different hypotheses over a specific mission environment including the possibility to choose alternative F-COA (Friend Course of Action) and O-COA (Opposite Course of Action) as well as general hypotheses (i.e. population). By this approach it is possible to initialize all the conditions to run the simulator and to carry out experimental analysis (i.e. to generate a specific population with characteristics compliant with hypotheses, or to define actions to be defined).
- Simulation Execution: in this phase the simulator is executed in order to generates the results that will be analyzed by the users and decision makers in the following phase; the simulation execution is possible with different operative modes (stand alone or federation) based on user needs and context
- Analysis & Report: it is the phase where the simulation results are available and focus on their analysis by applying techniques and methodologies as well as ad hoc support modules developed to quickly address these issues on operational sites; these approaches are based on comparing and evaluating the different alternatives based on simulation results; obviously despite the application area (training or operational support) it is critical to provide even self automated reports for speed up the final decision making and implementation processes for users.

In order to succeed with the development of a complex simulation framework as that one proposed by CAPRICORN project it is critical to properly address the requirement definition; in fact a simulator devoted to support country reconstruction, CIMIC and PSYOPS should address different utilization requirements considering its possible application on the field by decision makers without strong background in M&S and with limited resources in term of skills, time, data and computing capabilities.
As first step it is interesting to identify the main process for operating CAPRICORN simulator from user point of view as presented in figure 2.

- Specific Mission Environment Definition is a procedure devoted to elaborate the CAPRICORN knowledge bases and models in order to have a reference scenario corresponding to a Mission Environment ready to be used on the specific context (i.e. human, social, geographic)
- Simulation Fine Settings are procedures for changing the simulation parameters related to all the CAPRICORN boundary conditions for analysis over a specific mission environment; this allows to finalize the definition of the parameters in relation to the hypotheses and data within scenario related to a specific Mission Environment

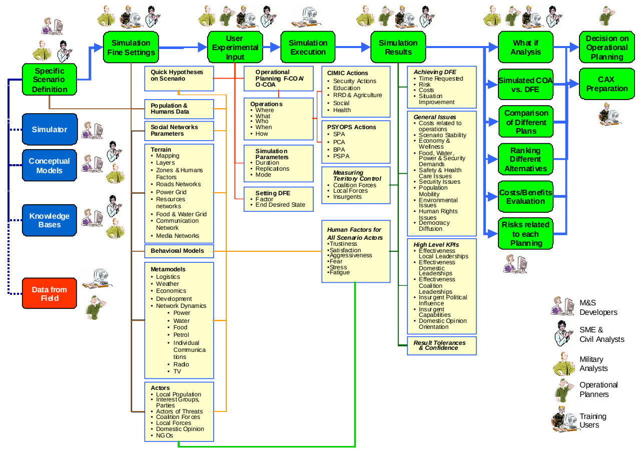


Figure 2: Procedure for Operating with CAPRICORN Simulator and main Input, Output and Elements present in the simulation

- User Experimental Input are procedures for quick changes on the scenario related to mission environment parameters, operations definition (i.e. F-COA, O-COA) and CAPRICORN simulation parameters
- Simulation Execution is the process for executing the CAPRICORN simulator on the predefined experimental settings and campaigns
- Simulation Results procedures include access to the all the output results of the CAPRICORN simulator
- What If Analysis is the analysis procedure based on testing different hypotheses, one by one, by simulating them; for instance if an analyst simulate the scenario with the hypothesis that tribe factors don't have influence it could re-run the simulator enabling tribe influence and then compare the results of both simulated scenarios.
- Simulated COA (Course of Action) vs. DFE (Desired Final Effect) is the analysis procedure based on comparing COAs results respect desired final effect; for instance if the user define as DFE the reduction of the insurgent in the area, then he fixes as COAs the schedule related to supporting local population by creating food storage for winter time in different villages, the simulator is executed and it is analyzed the measures related to this DFE.
- Comparison on Different Plans is the analysis procedure based on comparing final results of different planning respect different KPIs (Key Performance Indexes); the user
- Ranking Different Alternative is the analysis procedure based on creating a ranking list of the different alternatives respect a weighted target function based on KPIs
- Cost/Benefits Evaluation is the analysis procedure based on analyzing costs and benefits of a planning solution respect the different KPIs and MOP
- Risk Related to Each Planning is the analysis procedure based on estimating risk related to each planning based on replicated simulation runs
- Decision on Operational Planning is the procedure to use the synthetic and detailed results of CAPRICORN in the decision making process for achieve the final decisions on the operational Planning
- CAX Preparation (Computer Assisted Exercise) is the procedure to use the synthetic and detailed results of CAPRICORN in the CAX Preparation

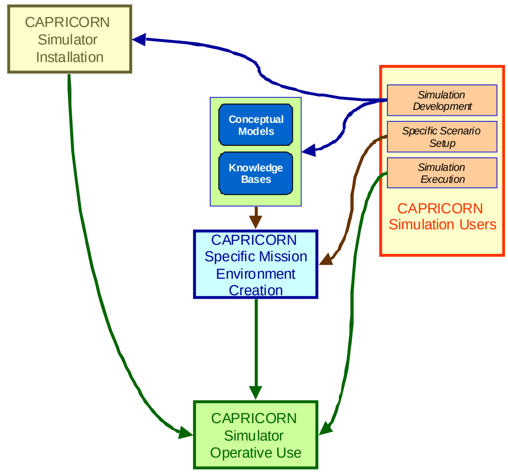


Figure 3. Main Issues from User Point of View

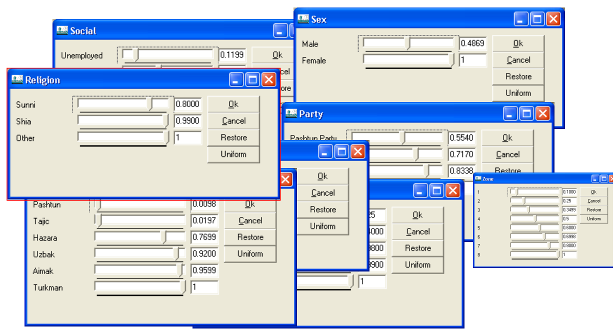


Figure 4. Population Characteristics defined in IA-CGF

It is evident that the general process is based on different main assets and aspects that are necessary to successfully run CAPRICORN; these are related to issues such as:

- CAPRICORN Simulator Installation
- CAPRICORN Configuration and Specific Mission Environment Creation
- CAPRICORN Operative Use
- CAPRICORN Users

CONCEPTUAL MODELS FOR CAPRICORN

Human Behavior Modeling is a very critical issue and in CAPRICORN represents a main aspect, so it is fundamental to consider the following aspects and factors (Bruzzone 2010):

- Ethnic and Religion
- Psychology
- Cultural and Political Situation

In particular in CAPRICORN the population is driven by agents and the virtual people are generated by applying Montecarlo Technique to specific group statistics in order to reproduce a region. The individual are created by defining in this way their:

- Location: geographic position where the object is located (latitude, longitude and altitude)
- Gender: male or female
- Age
- Health Care Status
- Ethnic Characteristics
- Social Status: level of unit wealthy
- Religion
- Education Level
- Tribe
- Political Party
- Nationality
- Psychological & Sociological Modifiers: characteristics of the object in term of stress, fear, aggressiveness, fatigue, stress and trustiness

The simulated people are driven by Intelligent Agents on the map and over their interactions within themselves and with different groups, they are able to move, to sleep, to wake up, to work, to have lunch, to have social events, to relax, to escape; they react dynamically during the simulation based on their situation awareness, their status and in some way in relation to their previous experiences (Bruzzone 1998).

People objects are connected among themselves through family relationships, friendships, working relationships generating a social network over the map; these social networks among people objects as well as their location over the map are generated by Montecarlo techniques applying specific algorithms; the location of located on the map by implementing and computing their compatibility considering all the different aspects (social status, education, ethnic groups, tribe religion etc.) through social economic algorithms. The Distribution of people on the map is indirectly defined through Zone Objects that define compatibility profile over the areas created over the terrain; all these data are stored in the CAPRICORN Database where relations and object are defined based on available knowledge from military or civilian reports. In addition each people object is connected in term of link and strength of the connection to different group objects; the group objects represent the interest groups and include a certain number of people characterized by same or similar features such as social status, religion, ethnic type, educational level, etc; for instance in rural areas group objects represent farmers, settlers, local authorities, young generation, old generation, a specific tribe, a specific ethnic group etc.

During the simulation the agent drive people objects and groups and allow their interaction to evolve dynamically; people object are enable to reinforce or resolve links with group objects based on their specific interests, while their actions and their perception are driven by the groups they still belonging and by the activities run by units and entities on the map (i.e. a CIMIC activity related to construct a school or an hospital). The groups characteristics are also generated by using Monte Carlo technique based on Configuration Group Objects (CGO); in fact the CGOs define the characteristics of the specific mission environment in term of group of interest and their mutual connections (i.e. mutual friendship, hostility etc.).

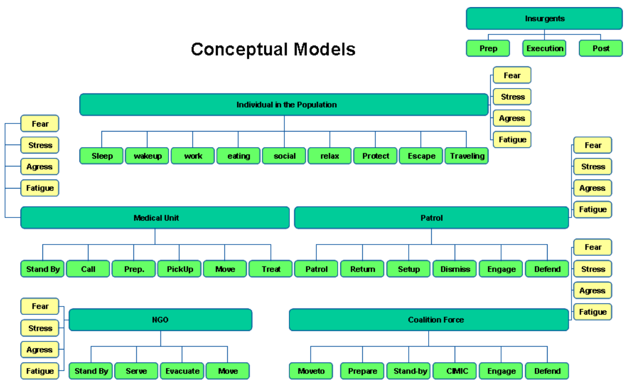


Figure 5. Example of Objects Functions

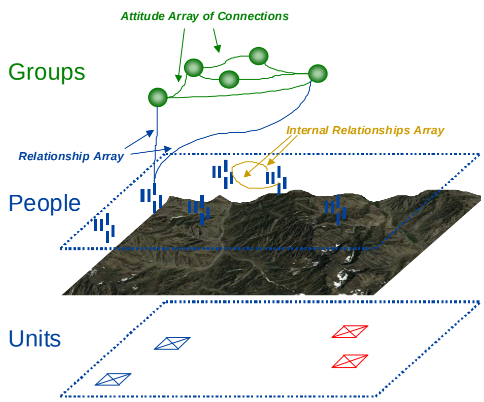


Figure 6. Scheme of Objects such as Units, People and Groups

In synthesis the definition of the population within a mission environment is based on Configuration Groups;

these identify the different groups in term of social, religious, political, ethnic and physiological aspects. The Population is created for each statistical group in references with their statistical characteristics based on Montecarlo Technique extracting data from defined statistical distributions based on CGOs; CGOs are defined by historical data and hypotheses by the users.

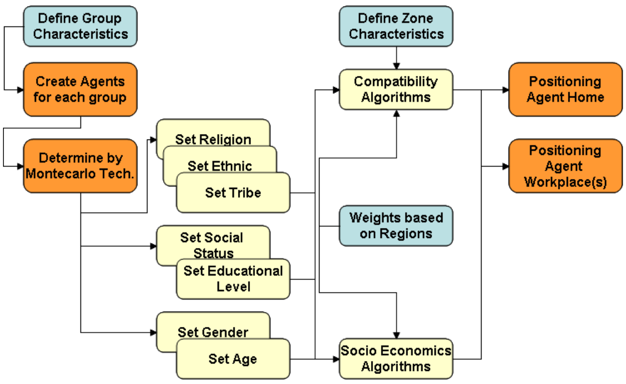


Figure 7. CAPRICORN Population Generation Process

In addition to these objects there are units (i.e. ambulances, convoy, military unit, insurgents etc.) that are part of the simulation; even these Unit Objects are driven by the Intelligent Agents and they are able to move and to perform actions autonomously referring to their tasks and scenario awareness or based on specific pre-assigned orders. The current scenario under analysis is related to South Asia and specifically to a region in Afghanistan. In particular it is possible to distinguish the following type units:

Units/Functions	Move	Stand by	Commit specific Actions	Execute specific Actions	Support specific Actions	Control an Area	Defend	Provide Logistic Support	Perform Surveillance	Promote Riots	Perform Attacks	Recruit Resources	Provide Social Aid	Support Locals
Coalition Forces	X	X	X	X	X	X	X	X	X					
ANA	X	X	X	X	X	X	X	X	X					
ANP	X	X	X	X	X	X	X	X	X					
Insurgents	X	X				X				X	X	X	X	
NGO	X	X	X	X	X								X	X

Figure 8. Example of Units Functions

- Coalition Forces are able to:
  - Move
  - Stand by
  - Commit/execute/support specific CIMIC or PSYOPs actions
  - Control an Area
  - Defend
  - Provide Logistics Support
  - Perform Surveillance
- ANA (Afghan National Army) are able to:
  - Move
  - Stand by
  - Commit/execute/support specific CIMIC or PSYOPs actions
  - Control an Area
  - Defend
  - Provide Logistics Support
  - Perform Surveillance
- ANP (Afghan National Police) are able to:
  - Move
  - Stand by
  - Commit/execute/support specific CIMIC or PSYOPs actions
  - Control an Area
  - Defend
  - Provide Logistics Support
  - Perform Surveillance
- Insurgents are able to:
  - Move
  - Stand by
  - Promote Riots
  - Perform Attacks
  - Control an Area
  - Perform Surveillance
  - Recruit Resources
  - Provide Social Aid
- NGO (Non Governmental Organization) are able to
  - Move
  - Stand by
  - Commit/execute/support specific CIMIC or PSYOPs actions
  - Support Locals

CIMIC or PSYOPs Action Objects are devoted to represent actions on the field and their evolution (i.e. definition, preparation and execution) and are characterized by the following attributes:

- *Id*: identification number
- Action type
- Locations latitude, longitude and altitude
- Time Plan & Duration
- Resources assignment (costs, units)
- Promoting Group: Operative Agent Actors
- Targets: Groups to get benefits of the action and its impact and Groups to get negative effect of the action and its impact

The Zone Object represents an area in CAPRICORN Environment where specific social, political, cultural conditions are applied. It could be sized as a single village or a rural area; zone could include a set of villages and in fact different zone are able to be overlapped in order to represent different mixes of people and groups over the same areas.

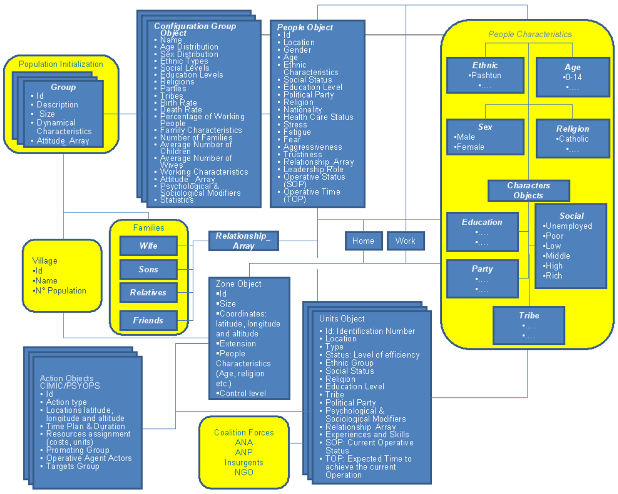


Figure 9. CAPRICORN Objects

CAPRICORN CAPABILITIES

The authors identify the main capabilities required for CAPRICORN simulator to support country reconstruction; these features are synthesized in the following list:

- Generation of the population
  - o Generation of People
  - o Creation of Networks for People and Group: Generation of families and inter-people connections (i.e. friendship) by building relationships within people, relationships between people and groups and attitudes among groups.
- Basic Visualization of social network for groups and single persons
  - Basic Visualization of Geographic zones and their features
  - Basic Visualization of human factors (i.e. psychological elements such as aggressiveness, fatigue, stress, fear, trustiness)
  - Basic Visualization of Resources and Infrastructures (i.e. food and water demand, power infrastructure)
  - Graphical representation of entities on the map
  - Interoperability with other Simulation systems through HLA

  In particular CAPRICORN Demonstrator allows users to configure a specific mission environment by:

  - Simulating the Mission Environment
  - Changing Hypotheses and Parameters of Human Elements in the Simulation by setting population characteristics
    - Social aspects in terms of percentages of people belonging, for instance, to the following class: Unemployed, Poor, Farmer, Worker, Middle Class, Wealthy, High Society
    - Religious aspects, including for instance the following classes: Sunni Muslim, Shia Muslim, Other.
    - Ethnic aspects by setting percentage of people belonging to ethnic groups in Afghanistan such as: Pashtun, Tajik, Hazara, Uzbek, Aimak, Turkmen, Baloch, Other
    - Tribe aspects by identifying clans (i.e. Safi, Ghilzai) and subclans (i.e. Gharghasht for Safi clan and Kharoti for Ghilzai clan)
- Implementing human behavior models (fear, fatigue, stress, aggressiveness, trustiness)
  - Reporting on statistics about human factors
- Grouping people in different groups depending on social, religious, educational, ethnic, political characteristic
- Simulating social network evolution
  - Reporting on social network changes due to the simulation
- Setting up different F-COAs and O-COAs by defining CIMIC/PSYOPS/Combat or other actions in term of type, location, target, approach, time, resources and specific attributes
- Reporting on Output and Target functions (i.e. KPIs)
  - Reporting actions performance indicators
  - Reporting costs, time

Additional capabilities guaranteed by using CAPRICORN are related to the following aspects:

- Capability to conduct replicated runs for measuring the tolerance and confidence band on the output variables and KPIs
- Capability to conduct runs corresponding to an experimental campaign runs for completing Sensitivity Analysis among a set of input variables and their interaction
- Capability to attribute to each out variable a confidence band
- Capability to rank different Simulation Runs and to compare costs/benefits, KPIs in absolute and statistical terms
- Capability to generate simulation evolution log for supporting scenario analysis and CAX preparation

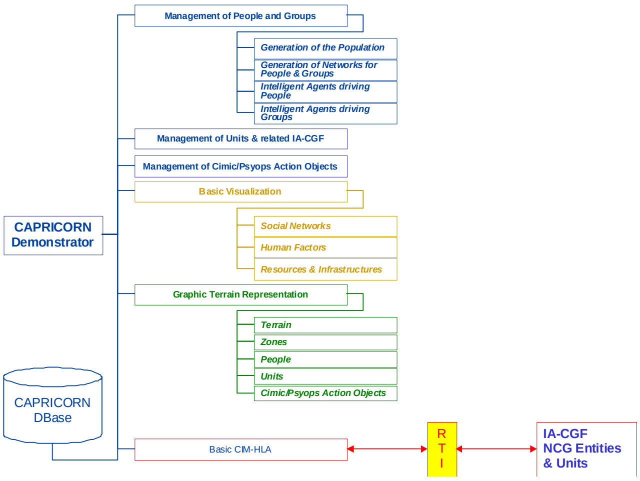


Figure 10. CAPRICORN General Architecture

In order to satisfy the needs and the requirements mentioned above, the authors designed the CAPRICORN architecture including different elements:

- A module for People and Groups Management:
  - Generation of the Population
  - Generation of Social Networks
  - Intelligent Agents driving People
  - Intelligent Agents driving Groups
- A module for Units Management & related IA-CGF
- A module for CIMIC/PSYOPs Action Objects Management
- A module for Basic Visualization of:
  - Social Network
  - Human Factors
  - Resources & Infrastructures
- A module for Graphic Terrain Representation:
  - Terrain
  - Zones
  - People
  - Units
  - CIMIC/PSYOPs Action Objects
- Basic CIM-HLA Module: CAPRICORN Interoperability Module

By this architecture it becomes possible to simulate a complex mission environment for the experimentation by simulating country reconstruction/CIMIC/ PSYOPS and other actions; CAPRICORN objects management and evolution is directed by IA-CGF while a specific NCF allows to define the mission environment by setting up configuration files and by changing aggregate variables; by this approach it is also possible to quickly setup the initial parameters for a complex scenario execution. The graphic representations of People, Groups and Units over the map while they evolve, movie and act is useful for VV&A as well as to support understanding of the different scenario evolution. The proposed CAPRICORN Federation support two operative modes:

- Stand alone Mode: by this approach the CAPRICORN is able to run independently on the mission environment.
- HLA Mode: in this case CAPRICORN guarantees the Mission Environment Management within an HLA (High Level Architecture Federation) interoperating with other simulators; in this case it is proposed to use IA-CGF Entity & Units, a specific NCF constituted by a constructive simulator able to show the situation of the units operating over a town and fully interoperable in HLA.

In particular the Interoperability Module HLA (Basic CIM-HLA) is a Module internal to CAPRICORN Demonstrator for Interoperability and it allows the creation, join and interoperation with a federation involving even other models. Its main functions are related to HLA Federation Management: Create, Join, Resign and Destroy Federation.

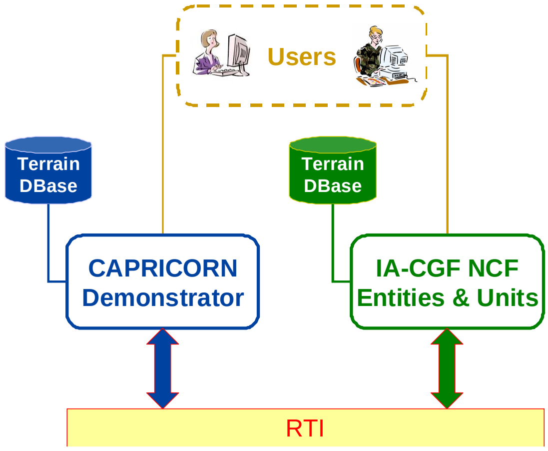


Figure 11. CAPRICORN Federation Architecture

CAPRICORN SCENARIO DEFINITION AND EXPERIMENTATION

Currently the authors are working on the Scenario Definition for CAPRICORN Experimentation; as anticipated the area of interest is a region in Afghanistan. CAPRICORN Demonstrator simulates people life in this afghan region focusing on CIMIC and PSYOPs operations impacts and effects on local population.

In particular, by analyzing users requirements, the following actions are included in the Demonstrator Development for the Experimentation phase:

- Digging Wells
- Canals Irrigation
- Roads and Infrastructures Security
- Police Station Buildings

- Messages Spreading by radio

CAPRICORN Experimentation is based on involvement of SME and military users and it includes the following phases:

- Phase A: Panels and Review of Concepts
  Reviewing Requirements and Discussing in Panels involving CAPRICORN Partners, Subject Matter Experts, Operational Planners on the different features:
  - Operational Planning Requirements for Stabilization
  - Synthesis on Conceptual Models of HBR, CIMIC and PSYOPS
  - Presentation of M&S Concepts developed for PMSEII
  - Review of CAPRICORN concepts in relation to:
    - Operational Planning overseas
    - Decision Making
    - Scenario Analysis
    - Risk Analysis
    - Training: CAX Preparation
    - Training: CAX execution
    - Concept Development & Experimentation
    - Education: Classes and Interactive Exercises

This Phase will be carried out continuously during the project during meetings and workshop as well as during the experimentation tests; a specific meeting is schedule in DHSS (Defense & Homeland Security Workshop) and Nato CAX Forum during I3M 2011 in Rome (www.liophant.org/i3m).

- Phase B: Execution of CAPRICORN
  - Users, Analysts and CAPRICORN Team working with subject matter experts and operational planners supporting Operational Planning and Scenario Analysis on the key features identified in term of:
    - Specific Mission Environment:
      - Afghan Region
      - Population, Social Networks & Groups Parameters
      - Alternative F/O-COA
      - Set CIMIC and PSYOPS

The CAPRICORN Experience is based on the execution of CAPRICORN Demonstrator over a predefined Mission Environment and to test capabilities and functions in both operation modes (HLA/Stand Alone).

This Experience should be performed by Users and Developers that works together; considering that CAPRICORN main users are in Italy and France the Experience is designed to satisfy the expectations of the two Countries, obviously respecting Project Resource original goals.

METRICS DEFINITION AND VV&A PROCESS FOR CAPRICORN

VV&A (Validation, Verification and Accreditation) process is very critical for simulators of complex scenarios and the authors defined the steps that must be followed along the whole project, especially in relation with SME and military users:

- Conceptual Model Validation: validation of conceptual model through Specifications, objects and State of Diagrams validation by SME
- Execution Testing: validation of Simulator Features and Functions by allowing SME to make tests and to use the demonstrator
- Integration Testing: Testing for evaluate HLA integration and interoperability benefits
- MSpE (Mean Square Pure Error), DOE (Design of Experiment), ANOVA (Analysis of Variance): Application of validation methodologies such as MSpE, analysis of variance and Design of Experiments in order to measure confidence band, optimal duration and optimal replications
- Sensitivity Analysis: Sensitivity analysis in order to identify critical factors for the simulated scenario
- Turing Test:the test measure the fidelity of the results by estimating the capability to discriminate real data from simulated data: for instance a subject matter expert is required to classify as real or simulated a set of reports provided over documented scenarios; in fact obviously Military SME have to propose a real case to be studied and simulated.

In order to support CAPRICORN VV&A the authors defined MOE and MOP; for instance, to measure the evolution of population satisfaction, it is possible to observe stress level, fear level, aggressiveness level, while to measure economic development it could be useful to define parameters in the simulation that provide estimation about the number of established new company, new markets or stores within a time windows.

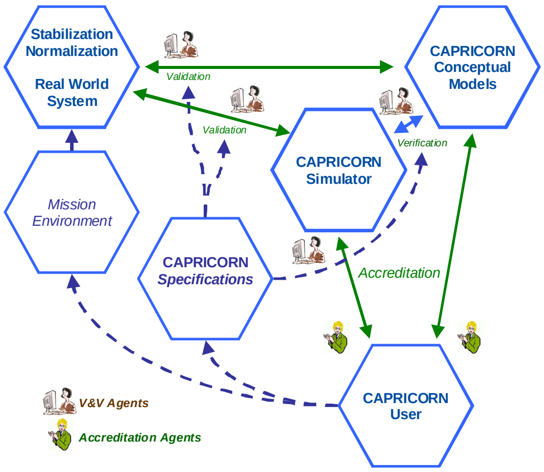


Figure 12. CAPRICORN VV&A Process

CONCLUSIONS

The authors propose a procedure for the development of a simulator as support for country reconstruction; this simulator is designed for being used both by operational planners and trainers.
The development of the conceptual models and general architecture result consistent and it was successfully validated and verified by SME during first phase of CAPRICORN project.
Currently the authors are working on the Scenario Definition for the CAPRICORN Experimentation; the area of interest will be a region in Afghanistan and CAPRICORN Demonstrator is expected to experiment different alternative reproducing people life in afghan villages over a region and to evaluate the CIMIC and PSYOPs operations evolution, impacts and effects on local population.
During the whole project great attention was devoted to VV&A even if this is just a R&D project; not too much to create a real mission environment, but to demonstrate the capabilities of such kind of simulators.
Based on author experience even in real case the data available are affected by strong uncertainty, very quick evolving nature (quick obsolescence) so it is expected that these population simulator should rely more on their capability to quickly test the different hypotheses from decision makers than on acquiring many data (that probably are incorrect and inconsistent); this is a classical example where we move from technology sensor data collection to intelligence report estimation and operational people estimations.
The new models such as CAPRICORN should be able to deal with these aspects, so it is very critical to develop even new VV&A (Validation, Verification and Accreditation) techniques; currently the authors are involved in several projects related to these aspects as further development.

AKNOWLEDGEMENT

The authors are very glad to thank the EDA and the Italian and French MoD (Minister of Defense) that are sponsoring CAPRICORN Project; special thanks to Giuseppe Menga and Lionel Kimeche as well as the Italian and French CAPRICORN User Groups and Champions; for further information please visit www.liophant.org/projects/capricorn.
The authors thanks the Simulation Team for the support provided by all the partners active in this initiative (www.simulationteam.com).

REFERENCES

[1] AJP-3.10.1, “NATO psychological operations doctrine”, NATO/PfP unclassified

[2] AJP-9, NATO civil-military co-operation (CIMIC) doctrine, NATO/EAPC unclassified

[3] Amico Vince, Guha R., Bruzzone A.G. (2000) "Critical Issues in Simulation", Proceedings of SCSC, Vancouver, July

[4] Avalle L, A.G. Bruzzone, F. Copello, A. Guerci, P.Bartoletti (1999) "Epidemic Diffusion Simulation Relative to Movements of a Population that Acts on the Territory: Bio-Dynamic Comments and Evaluations", Proc. of WMC99, San Francisco, January

[5] Bocca E., Pierfederici, B.E. (2007) “Intelligent agents for moving and operating Computer Generated Forces”, Proc.of SCSC, San Diego July

[6] Bocca E., Massei M. et al. (2010) "Italian Requirements on Simulation and CGF for Operation Planning", Simulation Team Technical Report, Genoa

[7] Bocca E., Massei M. et al. (2011) "CAPRICORN Subsystem Specification" , Simulation Team Technical Report, Genoa

[8] Bruzzone A.G., Massei M. et al. (2011) "Objective System Specification for CAPRICORN Project", Simulation Team Technical Report, Genoa

[9] Bruzzone A.G., Massei M. et al. (2011) "Review of Methodologies and technologies used in PIOVRA to be reused and/or adapted for CAPRICORN", Simulation Team Technical Report, Genoa

[10] Bruzzone A.G., Massei M. et al. (2011) "State of the Art for HBM in a CIMIC and PSYOPS", Simulation Team Technical Report, Genoa
[11] Bruzzone A.G. Tremori A., Massei M. (2011) "Adding Smart to the Mix", Modeling Simulation & Training: The International Defense Training Journal, 3, 25-27, 2011
[12] Bruzzone A.G., Tarone F. (2011) "Innovative Metrics And VV&A for Interoperable Simulation in NEC, Urban Disorders with Innovative C2", MISS DIPTEM Technical Report, Genoa
[13] Bruzzone A., (2010) “Human Behaviour Modelling as a Challenge for Future Simulation R&D: Methodologies and Case Studies”, Plenary Speech at Eurosim 2010, Prague
[14] Bruzzone A.G., Massei M. (2010) "Intelligent Agents for Modelling Country Reconstruction Operation", Proceedings of AfricaMS 2010, Gaborone, Botswana, September 6-8
[15] Bruzzone A., Madeo F., Tarone F., (2010) “Modelling Country Reconstruction based on Civil Military Cooperation”, Proc. of I3M2010, Fez, Oct.
[16] Bruzzone A.G., Massei M., Tremori A., Bocca E., Madeo F. (2010) "Advanced Models for Simulation of CIMIC Operations: Opportunities & Critical Issues Provided by IA", DIFESA2010, Rome, Italy
[17] Bruzzone A.G., Reverberi A., Cianci R., Bocca E., Fumagalli M.S, Ambra R. (2009) "Modeling Human Modifier Diffusion in Social Networks", Proceedings of I/ITSEC2009, Orlando, Nov-Dec
[18] Bruzzone A.G., Frydman C., Cantice G., Massei M., Poggi S., Turi M. (2009) "Development of Advanced Models for CIMIC for Supporting Operational Planners", Proc. of I/ITSEC2009, Orlando, November 30-December 4
[19] Bruzzone A.G., Frydman C., Tremori A. (2009) "CAPRICORN: CIMIC And Planning Research In Complex Operational Realistic Network" MISS DIPTEM Technical Report, Genoa
[20] Bruzzone A.G. (2008) "Intelligent Agents for Computer Generated Forces", Invited Speech at Gesi User Workshop, Wien, Italy, October 16-17
[21] Bruzzone A.G., Scavotti A., Massei M., Tremori A. (2008) "Metamodelling for Analyzing Scenarios of Urban Crisis and Area Stabilization by Applying Intelligent Agents", Proceedings of EMSS2008, September 17-19, Campora San Giovanni (CS),Italy
[22] Bruzzone A.G., Massei M. (2007) "Polyfunctional Intelligent Operational Virtual Reality Agent: PIOVRA Final Report", EDA Technical Report
[23] Bruzzone A., Massei M., Caussanel J. (2006) “Survey current CGF Reporting Current Situation” PIOVRA EDA Technical Report, Genoa, Italy
[24] Bruzzone A. G et al. (2004) "Poly-Functional Intelligent Agents For Computer Generated Forces", Proceedings of the 2004 Winter Simulation Conference Washington D.C., December
[25] Caussanel J., Frydman C., Giambiasi N., Mosca R. (2007) “State of art and future trend on CGF” Proceedings of EUROSIW2007, Santa Margherita, Italy, June
[26] Chauvancy F. (2011) “Réflexions militaires sur les opérations d’influence dans le cadre de la stabilization”, Séminaire « Compréhension des paramètres d’environnement dans les opérations de stabilisation », Université des Sciences de Marseille, 5 janvier 2011
[27] DIA-3.10, Doctrine interarmées, Centre Interarmées de Concepts, de Doctrines et d’Expérimentation, (2006) “Les opérations d’information” N°570/DEF/EMA/EMP.1/ NP May 29
[28] Fletcher M., (2006) “A Cognitive Agent-based Approach to Varying Behaviours in Computer Generated Forces Systems to Model Scenarios like Coalitions”, Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and its Applications,
[29] Frydman C., Massei M. et al. (2010) "French Requirements on Simulation And CGF for Operation Planning", Simulation Team Technical Report, Marseille
[30] Frydman C., Massei M. et al. (2011) "Uses-Cases Definition", Simulation Team Technical Report, Marseille
[31] Frydman C., Massei M. et al. (2011) "1st Report Experimentation Definition", Simulation Team Technical Report, Marseille
[32] Frydman C., Massei M. et al. (2011) "CAPRICORN System Specification", Simulation Team Technical Report, Genoa
[33] Frydman C., Massei M. et al. (2011) "CAPRICORN Architecture Specification", Simulation Team Technical Report, Marseille
[34] Frydman C., Massei M. et al. (2011) "Definition of Metrics and Validation, Verification and Plans", Simulation Team Technical Report, Marseille
[35] Galula D., (1964) “Counterinsurgency Warfare”, Praeger Security International
[36] Galula D., (2006) “Pacification in Algeria 1956-1958”, Rand Corporation,.
[37] Haugh, B. and Lichtblau, D., (2001) “An Information Technology Support Strategy for

PSYOP Impact Analysis," IDA Paper P-3587, Institute for Defense Analyses, February
[38] Haugh, B. and Lichtblau, D., (2000) "PSYOP Impact Analysis White Paper," IDA Paper P-3060, Institute for Defense Analyses, August
[39] Hue B., EMA/CPCO2J9, (2007) "What Do CIMIC Activities Bring to Stabilization1 Operations", France, Doctrine General Military Review #12, page 29
[40] Kallmeier V., Henderson S., McGuinness B., Tuson P., Harper R., Price S. Storr J. (2001) "Towards Better Knowledge: A Fusion of Information, Technology, and Human Aspects of Command and Control", Journal of Battlefield Technology, Volume 4 Number 1.
[41] Lichtblau, D., et al., (2004) "Influencing Ontology," ex-tended abstract in Behavior Representation in Modeling and Simulation Conference
[42] Rietjens S.J.H., M. Bollen, (2008) "Managing Civil-Military Cooperation", Military Strategy and Operational Art
[43] Nacer A., Taylor A., Parkinson G. (2007) "Comparative Analysis of Computer Generated Forces" Artificial Intelligence, Ottawa
[44] Rehse P., (2004) "CIMIC : Concepts, Definitions and Practice", Heft 136, Hamburg, June
[45] Thagard, P., (2000) Coherence in Thought and Action, Cambridge, MA: MIT Press
[46] Tremori A., Bocca E., Tarone F., Longo F., Poggi S. (2009)"Early Testing Procedures For Supporting Validation Of Intelligent Agents For Simulating Human Behavior In Urban Riots", Proceedings of MAS2009, Tenerife, September

Simulating Urban Environment for assessing impact of Alternative Command & Control Netcentric Maturity Models within Asymmetric Scenarios

Agostino Bruzzone
MISS DIPTEM University of Genoa
Email agostino,@itim.unige.it
URL www.itim.unige.it

Enrico Bocca
MAST srl
Via Molinero 1, 17100 Savona, Italy
Email enrico.briano@mastsrl.eu
URL www.mastsrl.eu

Federico Tarone
Simulation Team
Email federico.tarone@simulationteam.com
URL www.simulationteam.com

ABSTRACT

This paper focus on the development of a simulation framework for assessing the effectiveness and efficiency of different alternative Command, Control, Communications & Computers (C4) solutions in an urban environment affected by asymmetric warfare.
The authors presents their approach in developing Intelligent Agents Computer Generated Forces (IA-CGF) non conventional framework (NCF) for project CGF C4 IT; in this context it is critical to measure the effectiveness of different C2 (Command and Control) Maturity Models involving local and coalition forces, police and other resources in an oversea urban framework; in fact this it is one of main goal for the CGF C4 IT project that it is devoted to investigate alternative C2 models for guarantee agility in complex scenarios. The CGF C4 IT Federation is an High Level Architecture simulator, designed by the authors, and currently devoted to support Italian Army Simulation Capabilities; this represents an innovative and effective solution for investigating by experimental analysis the C2 Agility concepts within complex framework with special attention to Human Behavior Models.

Keywords: CGF, IA, C2, Agility, M&S, HBM

INTRODUCTION

Since C2 (Command and Control) in a Network-Centric Environment introduces challenges with many entities interacting, complex phenomena, significant influence of many stochastic and human factors, it is usually necessary to develop advanced simulation models to face these aspects. In fact, currently, it is very critical to develop Modeling & Simulation (M&S) assets able to support experimentation in this framework to evaluate new and alternative approaches. In fact the authors propose an approach developed to model of C4 (Command, Control, Communications & Computers) within complex scenarios; this paper focus in particular on asymmetric warfare in overseas urban environments; the authors propose the use of their IA-CGF (Intelligent Agents Computer Generated Forces) Libraries, Units and NCF (Non Conventional Frameworks) as building blocks to create a federation able to study these phenomena in order to support different aspects such as training, simulation based acquisition and experimentation & testing; in fact the use of IA-CGF allows to run multiple scenarios quickly driven by intelligent agents, in this way it becomes possible to investigate alternative solutions and parameters influences as well as to estimate critical target functions such as reliability, efficiency and effectiveness over credible complex scenarios.

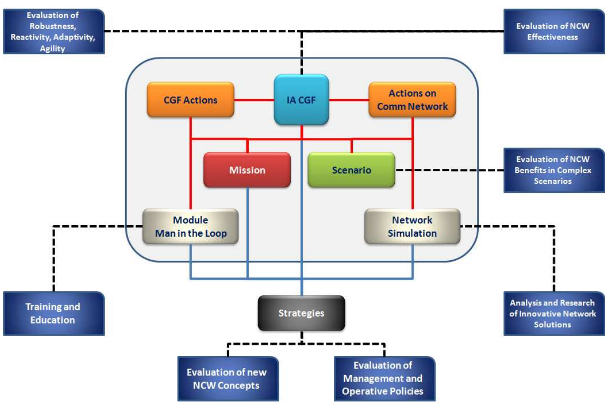
IA & C4 IN NETCENTRIC URBAN ASYMMETRIC ENVIRONMENTS

In fact, the aim of this research is to develop an interoperable simulator based on CGF (Computer Generated Forces) to create complex scenarios devoted to support evaluation and training on different C4 solutions such that ones proposed by N2C2M2 (NATO NEC C2 Maturity Model). Therefore, usually, development of innovative C4 solutions have a strong technological focus and don't pay too much on people and organizations, therefore on urban asymmetric scenarios the presence of HBM (Human Behavior Modifiers) at individual and organizational level have a very strong influence on the operations; it is evident that in addition to the technology network it is necessary to include effective models of human factors impact on the C4; this becomes even more important in NEC, in fact in this context mature models need to consider each single entity in relation to the entire information flow;

in addition, C2 is effective only if information flows result to be usable, reliable, consistent and influential for the elements of the network; in this sense a mature model must take into account the HBM and Social Networks in addition to the different approaches and methodologies designed to identify inaccurate or non-influential pieces of information. Thus, different architectures need to be tested, reproducing “intelligent” information processing. Considering these aspects, interoperable CGF designed to direct entities and to operate C2 in NEC need to demonstrate some intelligent capability in order to support the simulation of complex processes related to information sharing, processing and decision making; the authors propose to introduce intelligent agents able to apply various criteria and to define tasks, activities as well as to identify the most appropriate C2 policy duringt he dynamic evolution of the context. The authors defined how intelligent CGF have to be and how to use them to create and develop scenarios for verifying the real performances of C4 solutions in term of robustness and considering in-network functions (i.e. in-network data fusion, aggregation and routing).

TESTING NETCENTRIC ENVIRONMENTS vs. HUMAN FACTORS

A very important aspect in this context is related to the effect of human factors affecting elements in the organizations involved in C2 (i.e. stress, fear). It is evident that a C2 Mature Model should consider the HBM (Human Behavior Modifiers) in order to evaluate the man in the loop, the humans mistakes and to evaluate how people skills affect the overall complex system. In this paper it is proposed an innovative methodology based on the use of interoperable models integrated in an M&S environment aiming at testing the N2C2M2. The key to obtaining successful results is to create a Networking M&S environment that interoperates with intelligent CGF; the proposed CGF are the IA-CGF by simulation team that have been developed for experimentation and training on complex scenarios, taking care of technological, functional, operational & procedural aspects considering human factors.



In fact the Netcentric Evolution in Defense requires to develop Scenarios where to test effectiveness of Alternative Solutions in term of Infrastructures, Architecture and Management; Italian MoD launched in 2010 a new project on this subject titled CGF C4 IT, coordinated by Simulation Team MISS DIPTEM and involving directly and indirectly several entities such as: CAE, Mast, Selex Elsag, University of Genoa, University of Catania. CGF C4 IT goal is to develop and test a federation involving Intelligent CGFs (Computer Generated Forces) able to act as new technology enabler and to support coordinate actions on the battlefield in order to test NEC (Network Enabled Capability) solutions from technological and functional point of view; the intelligent CGF are expected to dynamically react to the Net Centric Resource Availability by using intelligent agents and with the benefits of previous researches carried out by Simulation Tema Genoa; the Italian MoD (Ministry of Defense) involved CESIVA (Center for Simulation and Validation of Italian Army) as opportunity to test the initiative within a military structure. In order to test different solutions the authors developed the CGF C4 IT federation that allows to test a complex scenario related to urban asymmetric warfare in an oversea mission environment.
The authors defined the elements to be included in the CGF C4 IT Scenario that include the following elements:

- Friend, Foe, Neutral and unknown Entities and Unitis on the Area
- Actions on the scenario affected by different Net Centric Frameworks and HBM
- Reactions to the dynamic evolution of NCW (Net Centric Warefare) and Communication Network Situation Evolution
- Actions devoted to restore, damage or compromise Communication Network for NCW

In addition CGF C4 IT projects allows to investigate different integration and interoperability solutions for Simulators with special attention to HLA (High Level Architecture) in term of effectiveness; in fact the IA-CGF used as agents are originally designed as HLA federates.
The CGF C4 IT project is based on several parts:

1. Feasibility Analysis related to the development of an environment where Intelligent Agent Computer Generated Forces (IA-CGF) operate within a Scenario affected by Human Factors, HBM, Communication systems and Network Centric Warfare (NCW) policies
2. Creation of a CGF C4 IT Demonstrator by developing an HLA Federation
3. Ad Hoc Metrics Development to tailor VV&A (Verification, Validation and Accreditation) on Simulation Involving CGF and C2 (Command & Control) in innovative Net-centric Environments

4. Development of Innovative Procedures and Best Practices in NEC for evaluation of different policies, strategies and doctrines by using CGF C4 IT Demonstrator
5. Introduction in the simulation of factors affected by C2 Software, quantity and type of communications to/from Command Centers and related Information Dissemination Policies (IDP)
6. Development of a Module interacting with IA-CGF for generating alarms to the Command Posts related to network problems and possible solutions
7. Investigating the possibility to Demonstrate the capability to stimulate the C2 by alternative methods (i.e. Gateway, Demonstrator PNRM 37/05 BIT Fase 2 and/or SIACCON2 Database)

STATE OF ART IN INTELLIGENT CGF MODELING HUMANS

Current military missions need simulation models able to capture and foresee the behavior of humans acting in social units, ranging from small groups, cultural and ethnic groups, and entire societies. Individual, Organizational, and Societal (IOS) models are needed to understand adversary and non-adversary behavior and to forecast the effects of alternative courses of action on that behavior.

A number of research efforts on human behavior modeling aimed at understanding complex behaviors are under way to analyze the effects that diplomatic, information, military, and economic (DIME) actions will have across the full range of the political, military, economic, social, information, and infrastructure context (PMESII). An important part of today's missions focus on courses of action (COAs) for influencing the attitudes and behaviors of noncombatants.

Predicting the response to integrated DIME COAs requires the analysis of the effects that a set of DIME actions will have across the full range of PMESII variables. A successfully analysis lays in the development of behavioral models which allow the simulation of the full range of PMESII variables and how they are affected by specific DIME actions. Hybrid models combining different paradigms at different levels of modeling offer considerable promise for the simulation of large-scale societal behavior. Then, different classes of models are used in the field of human behavior simulation and three levels of modeling are generally considered:

- Macro level, macroscopic models abstract away from individual behavior in favor of a broader more holistic view of behavior.
- Meso level, meso models are often referred to as meso-level models. Typically the models represent interactions and influences among individuals in groups and cover both individual and group phenomena with their interrelations.
- Micro level, microscopic models emphasize the description of the behavior of individuals rather than the description of global characteristics.

At the macro level, we consider interactions between macro-level variables, such as unemployment, crime, education, poverty, and resources. Formal macro modeling approaches principally based on dynamic system paradigm enable one to identify feedbacks and to see global effects without getting bogged down in details. At the other extreme, one can model the cognitive or affective processes of individual actors or at least their individual decisions and actions. These more micro modeling approaches include cognitive models from psychology, expert systems models, and rational choice models, which include game theory and decision theory. Over time, social scientists have come to appreciate the importance of the level in between—the meso level.

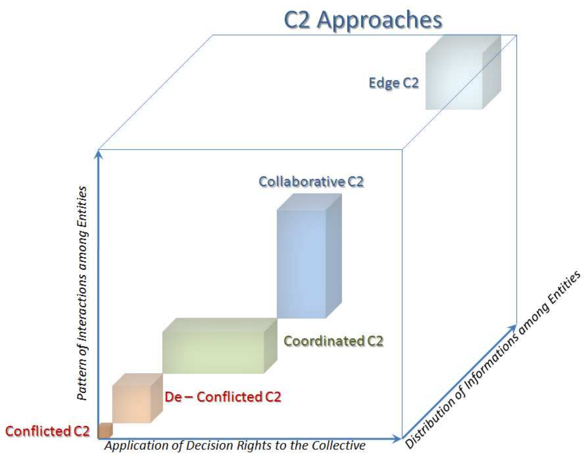
These three levels of models used different modeling paradigms such as: concept maps, social network models, influence diagrams, discrete event models, differential equations, causal models, Bayesian networks, Petri nets, event-based simulation, and agent based simulation. The need for a variety of modeling paradigms also stems from the fact that the different domains of knowledge needed to represent human behavior cannot be done by only one paradigm.

At the same time in 2007 it was demonstrated the result of a EDA Project, coordinated by Simulation Team MISS DIPTEM, related to Intelligent Agents reproducing civil disorders; the demonstration was based on applying PIOVRA agents into a civil environments by interoperating with a constructive theater simulator creating a multi level scenario, where agents was dealing with town evolution combining paramilitary, military, population and health care units and the constructive framework was simulating strategic planning ; in these case the use of G-DEVS/HLA Models was devoted to reproduce the agents structure, while AI (artificial intelligence) techniques was used for soft computing the emotional and rational behavior and perception of the units .

As follow up of this project there was presented RATS (Riots, Agitators, Terrorist Simulators) where warlords and civil disorders models was further extended. Currently the authors are developing new CAPRICORN agents devoted to consider not only the parameters related to psychological and social aspects but even to direct these perception among the different players (i.e. attribute the fear of some specific event to some of the parties playing the game).

NET CENTRIC COMMAND AND CONTROL MATURITY MODELS

The concept of Net-Centric was established in military sector and introduced in the early '90. This concept is used to describe an operational paradigm that exploits information and technological infrastructure to increase speed of command, resulting faster and more agile in carrying out operation and a sharing of knowledge. During recent years it was critical to consider how different C2 solutions are able to reproduce different maturity levels (i.e. conflicted, de-conflicted, coordinated, collaborative and edge).

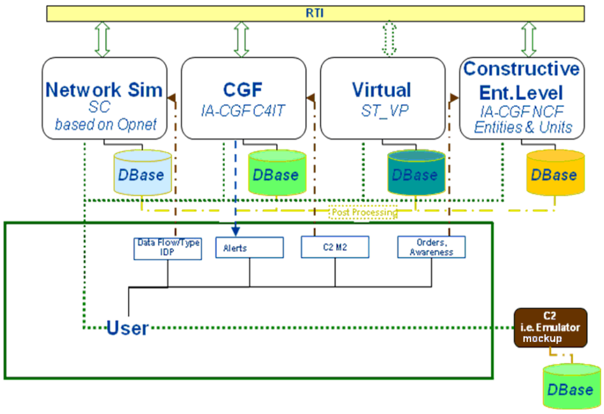


Nowadays, the critical issue on this matter is to develop experiments to support investigation about characteristics of C2 solutions such as robustness, resilience, agility. A major concept related to NecC2M2 (Netcentric Command and Control Maturity Model) is represented by the idea that in the same scenario over time, it could make sense to have different C2 maturity levels evolving based on the needs. Another important aspect is to test critical conditions or events that requires to adapt the C2 maturity level.

– Flat Model: IA-CGF Entities simulate units on the field (patrolling an area or checking critical points) and manages threats. Operative units are able to exchange messages in order to communicate and to transmit information about a potential threat or a particular event. For instance if a unit detects a threat (i.e. a suspicious vehicle or a potential explosion threat), it sends a message to the correspondent headquarters and/or to other units in the scenario.
– Hierarchical Model: In this case units transmit information in an hierarchical way directly to the relative chief or headquarter. If a unit detects a threat (i.e. a suspicious vehicle or a potential explosion threat), it sends a message to the higher level node, that sends the information to the involved operative units.

CGF C4 IT FEDERATION

The architecture for CGF C4 IT demonstrator is based on HLA (High Level Architecture) Architecture in order to create keep it flexible & expandable. Following the federation configuration:

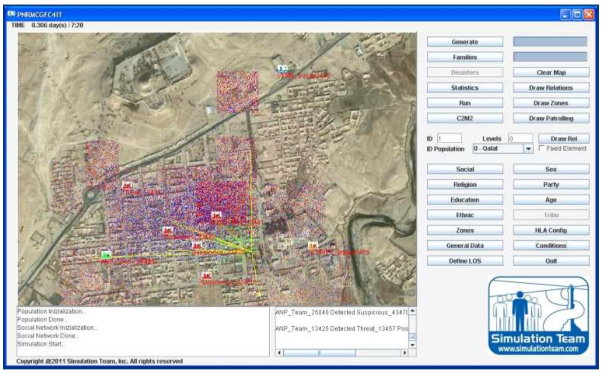


The IA-CGF C4 IT Federation includes:

– IA-CGF C4 IT federate that is in charge of the simulation of the town including agents for each individual, social networks and human factors (developed by Simulation Team)
– IA-CGF C4 IT Entities & Units including the intelligent agents that are driving all the units and entities on the scenario (developed by Simulation Team)
– ST_VP Virtual Simulation Environment for demonstrating virtual interoperable capability in reproducing a urban NCW (developed by Simulation Team)
– SC Simulation of the Communications Networks (developed by Selex Elsag)

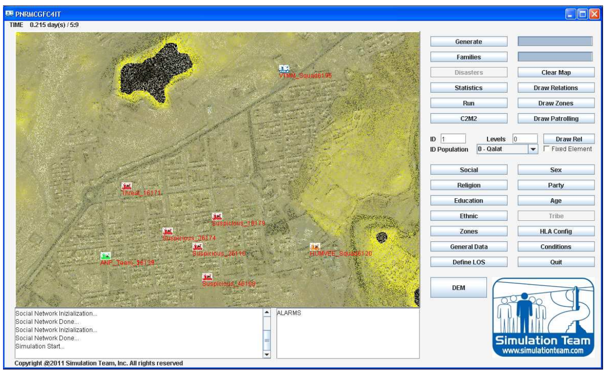
A real C2 integration was originally expected, therefore these external components evolution was delayed and it is currently in evaluation to just use some mockup or emulator on interconnection to demonstrate the concept; database structure is specific for each component considering the R&D Nature of the project, while obviously in case of further development it could become possible to automatically align the initial setup of the terrain DBases.

As anticipated the IA-CGF C4 IT simulator is the federate that is in charge of population management and human behavior modeling; the federate is based on IA-CGF Libraries and it represents a specific IA-CGF NCF ad hoc for the mission environment under analysis.



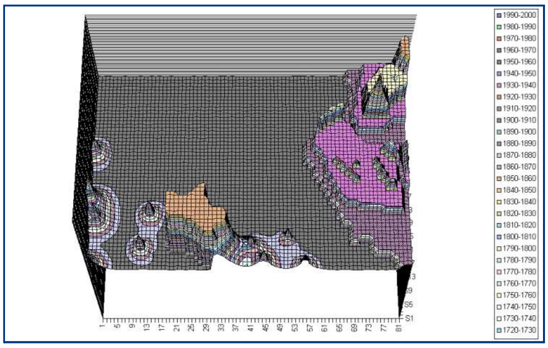
In fact based on the location chosen (i.e. in this case a 12'000 inhabitants town located in Afghanistan) and the reference local data inserted (i.e. different tribes and groups, statistical distribution in term of sex, ethnic, social level, education, religion, etc) IA-CGF C4 IT generates statistically the population for the specific scenario and based on user hypotheses; an agent for each individual of the population is generated and the model, after creating population, sets, based on similitude algorithms and parameters, the social connections among people (i.e. father, mother, son, daughter, friends). Referring to these connections the psychological parameters of each person evolve taking in consideration their specific social network and based on this fact every single individual influence other people as it is supposed to be happen in reality.
When the simulation starts people began to become alive (i.e. go to work, meet friends etc.) and they evolve based on events taking in consideration regular life, critical events, military presence and operations; even the units on the area (i.e. terrorist cells, snipers, squad, teams, platoons, etc.) are driven by IA-CGF Entities & Units. In fact IA-CGF Entities and Units federate drive the military operations and threat events:

- COA (Coalition)
- ANA (Afghan National Army)
- ANP (Afghan National Police)
- Demonstrations and Riots
- Asymmetric Threats (i.e. Snipers)



For instance, the Sniper threat acts in order to maximize the impact of its actions; as intelligent agent, it is capable to attract military force by deploying and activating an IDE (Improvised Explosive Device) and the decide to attack selected targets or to retire based on its perception of the scenario that consider HBM and estimation on the measure of success probability; obviously similar process is applied by military units, and their activity depends on NecC2M2, military coordination and the coordination depends on communication. The communication is managed by SC (SC Simulatore delle Comunicazione CGF C4 IT) based on OPNET Technology.
This federate allows to know if the message/ information are properly distributed in communication networks: it simulates communication and networking component performances that in many scenarios (i.e. in the mobile and tactical scenarios) are often unpredictable due to:

– Effects of the land orography
– Low bandwidth available
– Presence of noise (environment or intentional jamming)
– Communication and networking devices probability failure
– Low availability of communication infrastructures



Integrated in the federation there is also a complex alarms manage system directly connected with the federation.
Alarm Module generates alarms to the headquarters based on messages and data from IA-CGF Entities & Units about network behavior and conditions, and specific solutions to be performed in such cases:

– Loose a communication node
– Inconsistent and incoherent formation
– Not allowed or broken Communication

The CGF C4 IT Simulator have been already successfully passed the integration test and preliminary demonstration running in a joint configuration involving IA-CGF C4IT, IA-CGF Entities & Units, and SC

AFGHAN COMPLEX SCENARIO & METRICS

The scenario proposed in this paper is bases on an Urban Area where units Communication Squad, Patrolling Units, Local Army, Police, Population, Terrorists, Blue Forces operates as a network of Intelligence, Surveillance and Reconnaissance (ISR) assets. The simulation cover several aspects including Communications, C2M2, Operative Actions, Terrorist

Actions, False Alarms, Cooperation among friendly forces etc.

The IA-CGF are able to deal autonomously with their task, therefore the user is enable to inject events or to assign specific orders; an example of COA (course of action) to be investigated is proposed in the following as intelligent driven simulation scenario evolution:

- A Communications Group is deployed within an urban area in order to provide communication support to Friendly Forces
- Coalition Units provide Patrolling the Urban Area (i.e. 1 or 2 teams)
- On the Field also operate local army units and local police forces
- Town population evolve based on their life cycles on the area
- In the urban area there are threats and also suspicious activities
- Based on ROE (Rules of Engagements) and C2M2 (Command and Control Maturity Models) the units communicate and operate together in order to proceed in the detection, identification, classification and engagement of Threats
- Communication Units operates in order to guarantee communications
- Local Police and Local Army control the Territory and interact
- Threats operate in order to implement their plans
- Town population react based on their perceptions and scenario awareness.

All these entities and units as well as the town population are driven by IA-CGF; the authors defined the metrics for this scenario in order to support the VV&A of the CGF C4 IT Federation as well as to evaluate the effectiveness and efficiency of different C2M2 approaches within this NCW framework and even considering Urban disorders. Current the scenario involve Patrolling Units, Command Posts (PC) from Coalition and ANA as well and Police Forces (ANP) that will share information considering different C2M2 (i.e. Conflicted and De-conflicted).

In the current experimentation the threats are representing snipers within the Urban Context, urban disorders as well as IDE used to generate critical conditions by insurgents. Even in this simple scenario it is pretty complex to create an effective experimental analysis considering all the possible alternatives; in fact it is recommended the combined use of performance Metrics and Design of Experiments (DOE) in order to evaluate the effectiveness of different alternative C2 M2 (Command and Control Maturity Models) by the CGF C4 IT Demonstrator; it is evident that a main aspect in applying experimental techniques, in such a complex problems, it is to obtain the maximum quantity of information and knowledge about the scenario behavior with the minimum number of simulations runs; so in this case it is proposed to use:

- Means square pure error: in order to quantify the experimental error as measure of the risks and uncertainty on the simulation results in term of confidence bands
- Sensitivity analysis: by this approach it is possible to put in evidence what are main factors and their interactions that mainly influence variation of model outputs
- Metamodelling: approximate the real system behavior with metamodels able to support optimization on some controlled variable

Obviously for this kind of evaluation it is necessary to define the specific Measure of Effectiveness (MOE); for the proposed case study the MOE could result in something like: "to determine the capability of Forces to manage the threats in the proposed scenario through different policies, organization and infrastructures in terms of exchanging data, sensors, information and management reports and in terms of handling responsibilities, actions and tasks". In this case, for the sake of simplicity, the authors propose a set of Measure of Performances (MOP) that support the estimation of the above mentioned MOE and correlate the available forces with their effectiveness in term of time response and success issues.

By this approach the MOE and MOP address C2M2 respect Detection, Identification, Engagement & Agility capabilities in term of:

- Effectiveness
- Lead Time
- Success Probability
- Casualties
- Efficiency
- Private and PC Workloads
- Alarm Avoidance

Metrics are defined in order to measure operative effectiveness and communication network efficiency. MOP (Measurement of Performance) examples:

- Time to detect
- Time to engage
- Success rate
- Casualties
- Bandwidth Occupation
- Info processing workload
- Latency to delivery info

In fact the combined MOP for the CGF C4 IT experimentation are proposed by following relations.

$$MOP_d = k_{Act}\left(\frac{FEL_b}{FEL_{ref}} \cdot \frac{t_d}{t_{Act}}\right)^{-1}$$

$$MOP_i = k_{Id}\left(\frac{FEL_b}{FEL_{ref}} \cdot \frac{t_i}{t_{Act}}\right)^{-1}$$

$$MOP_e = k_{Eng}\left(\frac{FEL_b}{FEL_{ref}} \cdot \frac{t_e}{t_{Act}}\right)^{-1}$$

$$MOP_c = k_{Clr}\left(\frac{FEL_b}{FEL_{ref}} \cdot \frac{t_c}{t_{Act} + \Delta t_{Clr}}\right)^{-1}$$

$$MOP_{All} = MOP_d + MOP_i + MOP_e + MOP_c$$

$MOP_{d,i,e,c}$	Performance Index on the Target Function (TF): detection (d), identification (i), engagement (e), clearance (c)
$t_{d,i,e,c}$	time of achievement of the TF, in case of failure $\rightarrow\infty$
t_{Act}	expected threat activation time
Δt_{Act}	reference delta time expected to engage and clear the threat
$k_{d,i,e,c}$	Coefficient for Target Function
FEL_b	Blue Force Equivalent Level
FEL_{ref}	Reference Force Equivalent Level
MOP_{all}	Overall Performance Index

The overall performance index allows to compare quickly different C2M2 alternative respect a specific mission environment; the authors are currently developing experimental analysis on a specific Afghan town representing an interesting case study.

CONCLUSION

The use of innovative IA-CGF, introducing new characteristics in Simulation Scenarios, enables advanced applications for testing in complex scenarios ideas, alternatives, solutions. In fact as proposed in this paper the Intelligent Agents driving CGFs provide a Competitive Advantage for using Simulation in Exercise/Planning & Operation Support respect existing tools and packages.
Obviously it is critical to define specific application frameworks, as proposed in R&D (Research and Development) Projects, in order to support the necessary activities for validating and verifying the effectiveness of different methodologies in creating specific interoperable solutions; for instance CGF C4 IT project provided an opportunity to integrate human factors, asymmetric urban warfare, communication networks and N2C2M2 in a complex interoperable simulation. The success of this initiative is strongly based on the experience acquired in using IA (intelligent agents) as well as on the availability of IA-CGF Libraries, Units and NCF. The authors are considering to further extend these models to cover new mission environments.

ACKNOWLEDGMENTS

The authors acknowledge the Italian MoD for the support provided to this research within CGF C4 IT Project, PNRM 33/08 Contratto N. 9754.

REFERENCES

1. Amico Vince, Guha R., Bruzzone A.G. (2000) "Critical Issues in Simulation", Proceedings of SCSC, Vancouver, July
2. Avalle L, A.G. Bruzzone, F. Copello, A. Guerci, P.Bartoletti (1999) "Epidemic Diffusion Simulation Relative to Movements of a Population that Acts on the Territory: Bio-Dynamic Comments and Evaluations", Proc. of WMC99, San Francisco, January
3. Alberts D. S., (2002), "Information Age Transformation", revision June 2002, CCRP Publication Series
4. Alberts David S., Gartska J.J., Stein F.P. (2000) "Net Centric Warfare", CCRP, Washington
5. Alberts David S., Hayes Richard E., (2006) "Understanding Command and Control", Washington: CCRP, fig. 11, p. 35
6. Alberts D.S., Hayes R.E. (2003) "Power to the Edge", CCPR Publications, Washington D.C.
7. Alberts, D.S. (2007) “Agility, Focus, and Convergence: The Future of Command and Control.” The Int. C2 Journal, Vol. 1, No. 1, 1-30
8. Alberts, D.S. (2008) “NATO NEC C2 Maturity Model Overview,” Draft for Peer Review, SAS-065 Study Group, 2008.
9. Bocca E., Pierfederici, B.E. (2007) “Intelligent agents for moving and operating Computer Generated Forces” Proceedings of SCSC, San Diego July
10. Bruzzone A.G. Tremori A., Massei M. (2011) "Adding Smart to the Mix", Modeling Simulation & Training: The International Defense Training Journal, 3, 25-27, 2011
11. Bruzzone A.G., Tarone F. (2011) "Innovative Metrics And VV&A for Interoperable Simulation in NEC, Urban Disorders with Innovative C2", MISS DIPTEM Technical Report, Genoa
12. Bruzzone A.G., Massei M. (2010) "Intelligent Agents for Modelling Country Reconstruction Operation", Proceedings of Africa MS2010, Gaborone, Botswana, September 6-8
13. Bruzzone A.G., Cantice G., Morabito G., Mursia A., Sebastiani M., Tremori A. (2009) "CGF for NATO NEC C2 Maturity Model (N2C2M2) Evaluation", Proceedings of I/ITSEC2009, Orlando, November 30-December 4
14. Bruzzone A.G., Frydman C., Cantice G., Massei M., Poggi S., Turi M. (2009) "Development of Advanced Models for CIMIC for Supporting Operational Planners", Proc. of I/ITSEC2009, Orlando, November 30-December 4
15. Bruzzone A.G., Frydman C., Tremori A. (2009) "CAPRICORN: CIMIC And Planning Research In Complex Operational Realistic Network" MISS DIPTEM Technical Report, Genoa
16. Bruzzone A.G., Scavotti A., Massei M., Tremori A. (2008) "Metamodelling for Analyzing Scenarios of Urban Crisis and Area Stabilization by Applying Intelligent Agents", Proceedings of EMSS2008, September 17-19, Campora San Giovanni (CS),Italy
17. Bruzzone A.G., Massei M. (2007) "Polyfunctional Intelligent Operational Virtual Reality Agent: PIOVRA Final Report", EDA Technical Report, Genoa
18. Bruzzone A., Massei M., Caussanel J. (2006) “Survey current CGF Reporting Current Situation” PIOVRA EDA Technical Report, Genoa, Italy
19. Bruzzone A.G., Bocca E., Rocca A. (2006) "Algorithms devoted to Reproduce Human Modifiers in Polyfunctional Agents", Proc. of SCSC2006, Calgary, Canada, July 30-August

20. Caussanel J., Frydman C., Giambiasi N., Mosca R. (2007) “State of art and future trend on CGF” Proceedings of EUROSIW2007, Santa Margherita, Italy, June
21. Cinardo G. (2007) “Air Force Research Laboratory, Expert Common Immersive Theater Environment – Research and Development (XCITER&D) User’s Manual Version 1.0”, Mesa, Arizona, USA, April
22. Fletcher M., (2006) “A Cognitive Agent-based Approach to Varying Behaviours in Computer Generated Forces Systems to Model Scenarios like Coalitions”, Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and its Applications,
23. Galula D., (1964) “Counterinsurgency Warfare”, Praeger Security International
24. Galula D., (2006) “Pacification in Algeria 1956-1958”, Rand Corporation,.
25. Haugh, B. and Lichtblau, D., (2001) “An Information Tech-nology Support Strategy for PSYOP Impact Analysis,” IDA Paper P-3587, Institute for Defense Analyses, February
26. Haugh, B. and Lichtblau, D., (2000) “PSYOP Impact Analysis White Paper,” IDA Paper P-3060, Institute for Defense Analyses, August
27. Hue B., EMA/CPCO2J9, (2007) “What Do CIMIC Activities Bring to Stabilization1 Operations”, France, Doctrine General Military Review #12, page 29
28. Kallmeier V., Henderson S., McGuinness B., Tuson P., Harper R., Price S. Storr J. (2001) "Towards Better Knowledge: A Fusion of Information, Technology, and Human Aspects of Command and Control", Journal of Battlefield Technology, Volume 4 Number 1.
29. Lichtblau, D., et al., (2004) “Influencing Ontology,” ex-tended abstract in Behavior Representation in Modeling and Simulation Conference
30. Mursia A., Ganga L., Leonardi A., Morabito G., Rametta C., Wihl L. (2011) "A Simulation Study Of C4i Communications Network Under Cyber Attacks", Proc.of I3M2011, Rome, September
31. Rietjens S.J.H., M. Bollen, (2008) “Managing Civil-Military Cooperation”, Military Strategy and Operational Art
32. Nacer A., Taylor A., Parkinson G. (2007) Comparative Analysis of Computer Generated Forces’ Artificial Intelligence, Ottawa
33. Thagard, P., (2000) Coherence in Thought and Action, Cambridge, MA: MIT Press

AUTHORS' INDEX