Discrete Event System Conceptual Modelling for a Logging Company

Peter Mensah[(a)], Yuri Merkuryev[(b)], Jelena Pecherska[(c)],

Riga Technical University

[(a)]peters.mensahs@rtu.lv, [(b)]jurijs.merkurjevs@rtu.lv, [(c)]jelena.pecerska@rtu.lv

ABSTRACT

As conceptual modelling may be defined as a process of representing a real world model that could change anywhere along the simulation life-cycle study, it is therefore necessary for the conceptual model to be accurate in order for the simulation results not to be misleading. This is important especially because modelling and simulation can be exploited in decision making as any node and link of an organization can be represented by the simulation processes. Hence, an error in a conceptual model would lead to inaccurate simulation results. The fact that, many research papers have not emphasized much on conceptual model has actually motivated the authors to conduct a research in this area. This article therefore portrays how to develop a conceptual model within a discrete event system by exploiting a logging company as an example.

Keywords: discrete event system, conceptual model, framework for conceptual modelling, DEVS

1. INTRODUCTION

When exploiting modelling and simulation especially in decision making, it is vital to have an accurate conceptual model representing the real world model. However, it is not necessary to represent all the components of the real world as this would make the model cumbersome. It is therefore advisable to include only the components that are relevant to the study. Taking the logging companies for example, many of them face uncertainties due to risks that can affect their harvesting and operating processes which might lead to lower productivity and profitability. In order to avoid these uncertainties, it is therefore vital for these companies to be resilient. Once the problem has been identified, for example, *'how resilient the company is to disruptions'*, a conceptual model is then developed by utilizing the relevant components of the real world system. The necessity of obtaining an accurate conceptual model or abstract model of the real world system in order for the simulation results not to be misleading is of great importance. This is because it is transformed into a simulation model where various scenarios can be tested for accurate decision making that would yield to a resilient strategy. The **objectives of the research** is to develop a conceptual model of a discrete event system, using a logging company as an example, in order to obtain accurate simulation results that would yield to better decision making to facilitate a resilient strategy. The **research methodology** is obtained from scientific articles, journals and other relevant publications as well as the authors' professional knowledge and experiences in the field of modelling and simulation and management, which are then utilized in developing a conceptual model for a logging company. The article is divided into five sections. The first section elaborates on discrete event systems and their components with respect to a logging company. Conceptual modelling and its requirements are then discussed in the second section. The third section then highlights the framework for conceptual modelling. In order to understand the DEVS model, DEVS is briefly portrayed in the forth section followed by the DEVS model in the last section.

2. DISCRETE EVENT SYSTEMS

In discrete event systems, the state variable(s) changes at a discrete point in time (Banks, Carson, Nelson and Nicol 2005). Taking figure 1 into consideration, the logging company is an example of a discrete event system in which the state variable, for instance the volume of harvesting trees in the forest, changes only when the trees are cut down into logs and or when the logs are forwarded to the roadside warehouse. Hence, the volume of harvesting trees changes only at a discrete point in time. The necessary components of the logging company considered in this study are given below:

<u>Some Components of the Logging Company:</u>

System
- Logging company (L)

Entities
- Forest, warehouse

Attributes
- Volume of trees, quality of trees, capacity

Activities
- Harvesting, Forwarding, withdrawing

Events
- Arrival of customers' orders; starting and ending time of snowfall, rainfall, road closure

State Variables
- Volume of harvesting trees, forwarding and withdrawing logs

The system addressed is a logging company '*L'* with the forest and warehouse as its entities as they are objects of interests of the system. As the attributes belong to the entities, the forests and warehouses have attributes of volume of trees and capacity respectively. The activities which represent a time period of specific length are mainly harvesting (which is cutting down the trees into logs) and forwarding the logs to the roadside warehouse. Withdrawing the logs from the roadside to be distributed to the customers is another activity with respect to the warehouse entity. The events include arrival of customers' orders, the starting and ending time of heavy snowfall and rainfall as well as road closure which occur instantaneously and might change the state of the system. The state variables are the volume of 'harvesting trees', volume of 'forwarding logs' to the warehouse, and withdrawing the logs from the warehouse to be distributed to various customers, as they can describe the system at anytime relative to the objective of the study. After analysing the component of the system, conceptual modelling can then be developed as discussed in the next chapter.

3. CONCEPTUAL MODELLING

According to Robinson (2008a), 'conceptual modelling is a non-software specific description of the computer simulation model describing the objectives, inputs, outputs, content, assumptions and simplifications of the model'. Simplifying the definition, conceptual modelling is a process of abstracting a real world model which could change anywhere along the simulation life-cycle study. Moreover, Becker and Parker (2011), highlighted that the hypothetical complete description of the original system is formed by the conceptual model. Hence, when forming the conceptual model, having adequate knowledge about the objectives, inputs and outputs of the real world model is important. It is also vital to consider various assumptions and simplifications in decision making related to the content of the model. Assumptions and simplifications are also quite distinctive concept in conceptual modelling (Robinson 2008a), as they are made due to uncertainties and simplification respectively. From figure 1 below,

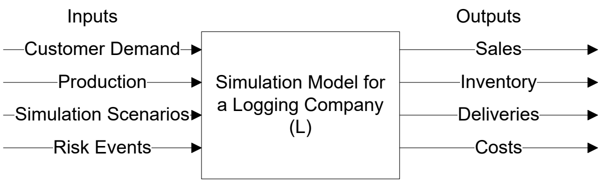

Figure 1: Conceptual Model of Logging Company 'L'

the conceptual model of a logging company *'L'* in which the simulation model receives inputs from the left side of the diagram and delivers output on the right is portrayed. The inputs, which represent experimental data and events, consist of demand, production, simulation scenarios and risk events. On the right side of the diagram, the outputs -performance measure estimates- consist of sales, inventory, deliveries and costs. The performance measures could be expressed in monetary units as risks in the supply chain should be measured, valued and managed by costs. As the risk events the system faces such as heavy rainfall, heavy snowfall, road closure, wood quality etc might have an impact on the system's output, the key question is therefore; '*how resilient is the system if affected by these risks and what is the impact on the system's performance*? This is where the requirements of a conceptual model play a great role as discussed below.

Requirements of a Conceptual Model:
The requirements of a good conceptual model include validity, credibility, feasibility and usefulness (Robinson 2008a).
Validity:
Nance (1994), points out that it is important for the model to be correct and can be easily tested. Hence, the model should be able to produce sufficient and accurate results for the purpose: For example, understanding the impact of the risks on the volume of logs delivered.
Credibility:
Unlike validity, credibility is more from the perspective of the client. It is therefore important for the modeller not only to include the important components and their relationships in the model, but to also be able to convince the clients about the accuracy. The model should also be easy to understand (Brooks and Tobias 1996a) by the clients who have to be capable of interpreting the results and believing in their accuracies.
Usefulness:
This is when the model is sufficiently easy to use, flexible, visual and quick to run. Nance (1994), further highlights that the model should be adaptable, reusable and maintainable. Consequently, the model could be used again for the same or different researches.
Feasibility:
Pritsker (1986), highlighted that feasibility should be timely; whilst Brooks and Tobias (1996), further elaborate on time and cost to build and run a model, as well as analysing the results. Hence, the modeller should be able to build the model within the available data and time constraints. The aforementioned will help develop a simple model and the results will be easy to interpret. The framework for conceptual modelling is discussed next as it helps to support the formation of the conceptual modelling of the logging company 'L'.

4. FRAMEWORK FOR CONCEPTUAL MODELLING

Taking the framework for conceptual modelling into consideration provides guidance by utilizing a set of steps and tools as given below. Figure 2 portrays a framework for conceptual modelling (Robison 2008), consisting of five activities which are as follows:

- Understanding the problem situation.
- Determining the modelling and general project objectives.
- Identifying the model outputs (responses).
- Identify the model inputs (experimental factors).
- Determining the model content (scope and level of detail), identifying any assumptions and simplifications.

ISBN 978-88-97999-77-5; Affenzeller, Bruzzone, Longo, Merkuryev and Piera Eds

The first step which is 'understand the problem situation' and this is clearly defined with respect to the conceptual model in figure 1.On the other hand, if the problem situation is not fully understood, it may lead to difficulties and therefore assumptions have to be made in this case to get a better understanding of the problem situation. After which, it is vital to determine the modelling and general project objectives given in the second step.

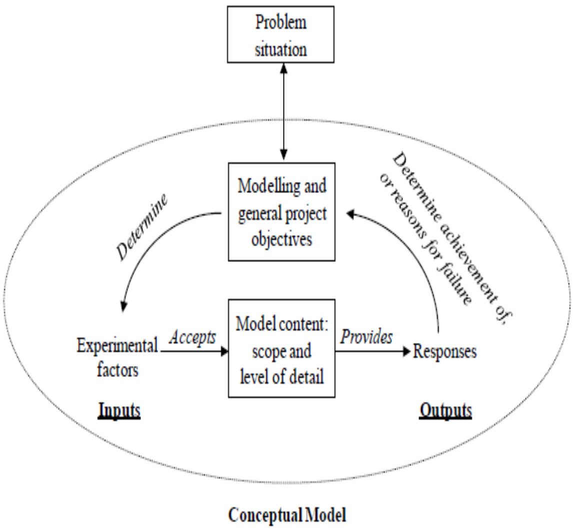

Figure 2: A Framework for Conceptual Modelling (Robinson 2008b)

Flexibility, run-speed, visual-display and the reuse of the model and its components are considered within the general project objectives. These objectives help to derive the conceptual model by defining the inputs (experimental factors) and the outputs (responses) of the model which are the third and forth steps respectively. The input data can be experimented in order to meet with the model's objectives with the output determining if the objectives have been met and this is also illustrated in figure 1. The output could be represented statistically or graphically. The last step is defining the content of the model in terms of its scope and level of detail. Throughout this process, it is also necessary to identify assumption and simplification. Subsequent to understanding the problem situation, and obtaining the requirements and framework of the conceptual model, data is then collected whereby a simulation model is developed for research analysis by testing various scenarios. From another perspective, a DEVS model is considered in developing a conceptual model as discussed in the next paragraph.

5. DEVS

Before exploiting the DEVS model, it is necessary to grasp a better understanding of DEVS which is one of the discrete event methodologies introduced by B. Zeigler in 1976. It is important for discrete event models as well as discrete time and differential equations because of its computational capabilities for implementing behaviours (Zeigler 1976). Moreover, it works with an infinite number of states that is useful for numerical integration. The early form of DEVS is known as Classic DEVS, but after 15 years, a modified version was introduced, namely Parallel DEVS. Classic DEVS is considered in this case since it is relevant to developing the model for the logging company system portrayed in figure 1. In addition, DEVS (discrete event system specification) is a structure '*M' in which; M=* $\{X, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta\}$

Where;

X- is the set of input values

S- is a set of states

Y- is the set of output values

δ_{int}*,: S→S is the internal transition function*

δ_{ext}*: Q x X→S is the external transition function*

The total state set: $Q = \{(s,e) | s \epsilon S, 0 \leq e \leq ta(s)$ *where 'e' is the elapse time since last transition.*

λ: S→Y is the output function

ta: $S \rightarrow R^{+}_{0,\infty}$ *is the set of positive reals with zero and infinity.*

According to Castro and Kofman (2000), 'DEVS is a system theoretic-based representation of the systems whose input/output behaviour can be described by sequences of events. Thus, the state variable(s) changes at a discrete point in time. From a practical point of view, a logging company is an example of a discrete system in which the state variables, volume of harvesting, forwarding and withdrawing logs, changes only when the trees are cut down into logs and or when the logs are forwarded to the roadside warehouse, and withdrawn from the warehouse to be distributed to customers. Hence, the volume of trees harvested changes only at a discrete point in time. In figure 1, the system gets its input from the environment, which is then transformed and sent back into the environment as output.

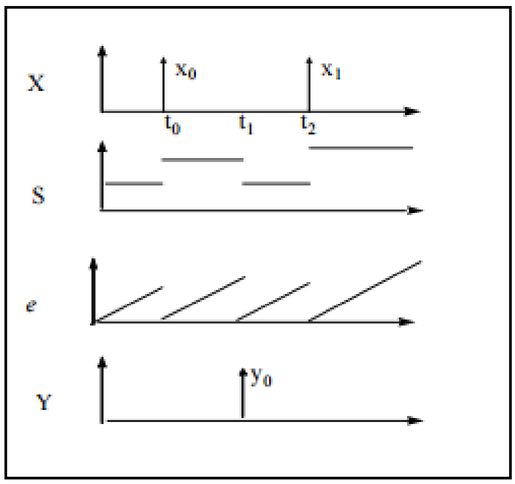

Figure 3: DEVS Trajectories

However, the risk events might affect the process of transformation which could cause an impact on the output. The behaviour of DEVS is illustrated in figure 3 where the input trajectory '*X* 'is a series of events occurring at t_0 and $t_2$with time t_1 representing an internal event. The state 'S' changes with respect to the input trajectory with the upper lines reacting to external events and the lower ones with the internal events. The elapsed time trajectory ‘e’ shows the flow of time that resets to zero at every event. The output trajectory ‘Y’ shows the output events generated by the output function ‘just before applying the internal transition function at internal events’ (Zeigler 1976). When taking the logging company into consideration, the demand of the volume of trees to be harvested is realized at the time $t_{0.}$ This then changes the state *‘S’* and the elapse time is reset to zero. At the time $t_{1,}$ which is an internal event, the trees are harvested and forwarded to the roadside resulting in a change of the system *‘S’* and yielding to an output Y_0 in the form of pile of logs on the roadside ready to be distributed to the customers. The DEVS model is discussed next.

6. DEVS MODELS

DEVS is divided into two classes of models, namely Atomic models and Coupled models. The Atomic models are exploited in basic formalism and the Coupled models are expressed using the coupled model specification. After careful examination, a pipeSimple Classic DEVS (Coupled) is selected to be exploited in developing a model with respect to the logging company illustrated in figure 1. The pipeline coupled model is elaborated in figure 4.

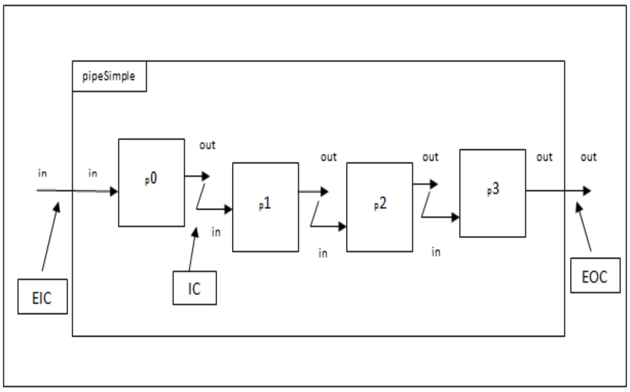

Figure 4: Pipeline Coupled Model for a Logging Company System ‘L’

With reference to figure 4, four processors are connected in series to form a pipeline in order to construct a coupled model for the logging company system.
Where:

- p0- representing the input, for example the customers demand etc indicated in figure 1.
- p1- harvesting of trees into logs
- p2- forwarding the logs to the roadside
- p4- the output, for example, distribution to the customers as shown in figure 1.

The output port of the first processor(p0) is coupled with the input port of the second processor(p1) and the rest of the processors are connected in a similar way known as internal coupling(IC). The input port of the first processor ‘in’ is connected to the input port of the pipeline know as external input coupling(EIC). Similarly, the output port of the 4th processor is linked with the external output port(EOC).

Hence, the DEVS formalism is meant to build models from components. DEVS specification with ports includes the external interface(input and output ports and values), the components (that is DEVS models) and the coupling relations.
From, $N = (X, Y, D, \{M_d | d \in D\}, EIC, EOC, IC, select)$
X is a set of input ports and values
Y is the set of output ports and values
D is the set of component names
M_d *is a DEVS*
The pipeline coupled DEVS specification is given as:
$N = (X, Y, D, \{M_d | d \in D\}, EIC, EOC, IC).$
where;
InPorts = {“in”},
$X_{in} = V$ *(an arbitrary)*
X = {(“in”,v)|vϵV}
OutPorts = {“in”}
$Y_{out} = V$
Y = {(“out”, v)|vϵV}
D ={processor0,processor1, processor2,processor3}
$M_{processor3} = M_{processor2} = M_{processor1} = M_{processor0=processor}$
EIC = {((N, “in’), (processor0, “in”)}
EOC = {((processor3, “out”), (N, “out”)}
IC = {((processor0, “out”), (processor1, “in”)),
((processor1, “out”), (processor2, “in”)),
((processor2, “out”), (processor3, “in”))}
Select (D ’) = the processor in D ’ with the highest index.
The DEVS model can then be transformed into a simulation model by using DEVS tools such as DEVS-C++, DEVSim++ etc, for obtaining simulation results.

7. CONCLUSION

This article has discussed how to develop a conceptual model in a discrete event system by exploiting a logging company as an example. The discrete event system and its components were considered from a practical point of view with respect to the logging company. This was followed by exploiting two conceptual models namely the Robinson’s model and the DEVS model. The requirements of Robinson’s model namely validity, credibility, feasibility and usefulness were discussed followed by the framework for building the model. The DEVS structure and its behaviour were described in theory which were they applied in practice onto the logging company. This was followed by developing a DEVS model in which a pipeSimple Classic DEVS (Coupled) was selected and its specification given as: $N = (X, Y, D, \{M_d | d \in D\}, EIC, EOC, IC),$ where M_d is a DEVS. The DEVS model seems to be more complicating when compared to the Robinson’s model which is simpler and very practical.
Future research
Translating both conceptual models namely Robinson’s and DEVS into simulation models by exploiting

SIMUL8 and DEVSim++ respectively in order to study the system of the logging company, analyze the impact of the risks on the system and develop a resilient strategy will be considered next. Moreover, application of the Six Sigma processes within the SIMUL8 software in order to develop a resilient strategy will also be considered.

REFERENCES

Banks J., Carson J.S., Nelson B.L., Nicol D.M., 2005. Discrete Event Simulation. 4th ed.Upper Saddle River,NJ 07458: Pearson

Brooks R.J., Tobias A.M., 1996. Choosing the Best Model: Level of Detail,

Complexity and Model Performance. Mathematical and Computer Modeling, 24 (4), pp. 1- 14.

Castro R., Kofman E., 2000. A Formal Framework for Stochastic Discrete Event System Specification Modeling and Simulation. Available online: http://usuarios.fceia.unr.edu.ar/~Kofman/files/sim_stdevs.pdf

Nance R.E., 1994. The Conical Methodology and the Evolution of Simulation Model

Development. Annals of Operations Research, 53, pp. 1-45.

Becker K., Parker J. R., 2011. Guide to Computer Simulations and Games. Hoboken, NJ: John Wiley & Sons.

Pritsker A.A.B., 1986. Model Evolution: A Rotary Table Case History. Proceedings of the 1986 Winter Simulation Conference. IEEE, Piscataway, NJ, pp. 703-707.

Robinson S., 2008a. Conceptual modelling for simulation Part I: definition and requirements, J. Oper. Res. Soc. 59 (3), pp. 278–290.

Robinson S., 2008b. Conceptual modelling for simulation Part II: a framework for conceptual modelling, J. Oper. Res. Soc. 59 (3) pp. 291–304.

Zeigler B.P., 1976. Theory of Modeling and Simulation. Wiley, New York.

AUTHORS BIOGRAPHY

PETER MENSAH has an MBA in Business Management, MSc in Computer Science, and is also a Cambridge CELTA qualified English language teacher. He is currently a lecturer in Business Management and Innovation at Riga Technical University in Riga, as well as an English Language tutor at Riga Business School, Riga, Latvia. He has also worked as a PSE tutor, and guest lecturer in areas of Project Management, Systems Thinking and Teaching and Learning from a Cultural Perspective' at Coventry University for the past five years. He is currently a PhD Candidate at Riga Technical University, Department of Modelling and Simulation, and his research area is in 'Using Simulation to Develop a Resilient Supply Chain Strategy' in which he has written several scientific publications.

YURI MERKURYEV is a Professor and Head of the Department of Modelling and Simulation at Riga Technical University. He obtained the Dr.sc.ing. degree in System Identification in 1982, and Dr.habil.sc.ing. degree in Systems Simulation in 1997, both from Riga Technical University. His professional interests include modelling and simulation of complex systems, methodology of discrete-event simulation, supply chain simulation and management, as well as education in the areas of simulation and logistics management. Professor Merkuryev is Full Member of the Latvian Academy of Sciences, Fellow of the European Academy for Industrial Management, president of Latvian Simulation Society, board member of the Federation of European Simulation Societies (EUROSIM), senior member of the Society for Modelling and Simulation International (SCS), senior member of the Institute of Electrical and Electronics Engineers (IEEE), and Chartered Fellow of British Computer Society, FBCS CITP. He is an associate editor of "Simulation: Transactions of The Society for Modelling and Simulation International", and editorial board member of "International Journal of Simulation and Process Modelling". He has authored more than 330 scientific publications, including 7 books and 6 textbooks.

JELENA PECHERSKA is a Doctor of Information Technology in the field of system analysis, modelling and development. She has been working for Riga Technical University since 1979. At the moment, she is a Docent at the Department of Modelling and Simulation of Riga Technical University. Professional interests include methodology of discrete-event simulation, combined simulation, supply chain modelling, practical applications of discrete-event simulation and discrete-event simulation in education. She is a member of the Latvian Simulation Society.