A SIMULATION-BASED APPROACH TO THE VEHICLE ROUTING PROBLEM

Stefan Vonolfen[a], Stefan Wagner[b], Andreas Beham[c], Monika Kofler[d], Michael Affenzeller[e], Efrem Lengauer[f], Marike Scheucher[g]

[a][b][c][d][e] Upper Austria University of Applied Sciences, Campus Hagenberg
School of Informatics, Communication and Media
Heuristic and Evolutionary Algorithms Laboratory
Softwarepark 11, A-4232 Hagenberg, Austria

[f][g] Upper Austria University of Applied Sciences, Campus Steyr
School of Management
Logistikum Research
Wehrgrabengasse 1-3, 4400 Steyr, Austria

[a]stefan.vonolfen@heuristiclab.com, [b]stefan.wagner@heuristiclab.com, [c]andreas.beham@heuristiclab.com, [d]monika.kofler@heuristiclab.com, [e]michael.affenzeller@heuristiclab.com, [f]efrem.lengauer@fh-steyr.at, [g]marike.scheucher@fh-steryr.at

ABSTRACT

In this work a simulation-based approach to the vehicle routing problem is presented. The simulation system is used to examine different problem environments and to optimize scenarios based on a generic domain model.

Using the simulation environment, a concrete practical transport logistic problem scenario is modeled and the simulation is coupled to the HeuristicLab optimization framework.

Keywords: vehicle routing problem, optimization, simulation

1. INTRODUCTION

The vehicle routing problem (VRP) is a well known problem in literature (for an overview, see for example Cordeau, Laporte, Savelsbergh and Vigo (2005)) and is used to model practical problem situations in transport logistics. The VRP formulation consists of a fleet of vehicles serving a set of customers with a certain demand from a single depot. There are several derivatives like the capacitated VRP with time windows (CVRPTW), or the multiple depots VRP (MDVRP).

In this work we present a simulation-based approach to vehicle routing problems. This enables the simulation and optimization of different transport logistic scenarios. Furthermore the simulation environment is a step towards online optimization. This can be performed by replacing the simulated values with real-life data.

In our approach, a simulation environment is used to examine diverse problem environments wich can be specified and simulated. The goal is to examine different practical transport logistic scenarios

Properties of the problem environment are for example the number of customers, the customer ordering behavior, the number of vehicles, the number of depots or the delivery strategy.

The simulation of a certain problem environment generates scenarios that can be optimized. For example, during a simulation run, customers make orders which results in a delivery scenario that is formulated and optimized as a VRP.

2. DOMAIN MODEL

The simulation environment is based on a generic domain model for transport logistic scenarios. Basically the domain model stores the master data and represents orders and deliveries.

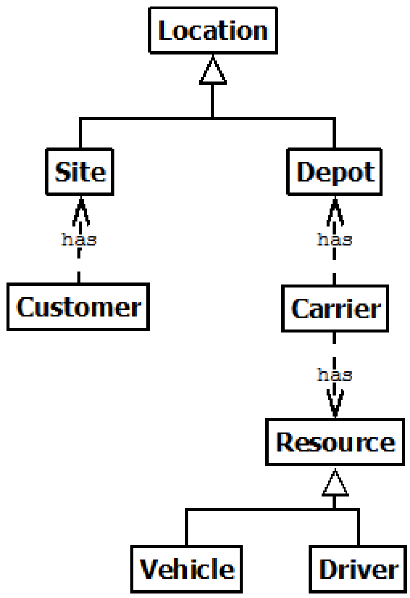

Figure 1: The master data

Figure 1 shows the model for the master data. It consists of location information. Both sites of customers and depots of carriers have a location given in a coordinate system. The carrier has a certain set of resources which consists of drivers and vehicles.

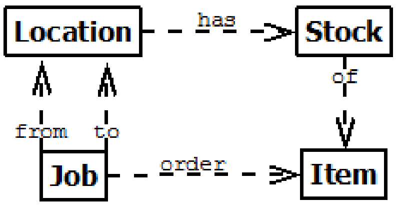

Figure 2: Orders

The representation of orders in the domain model is illustrated in Figure 2. Every location has a given stock of items. A job represents an order to deliver a certain item from a given location to another given location.

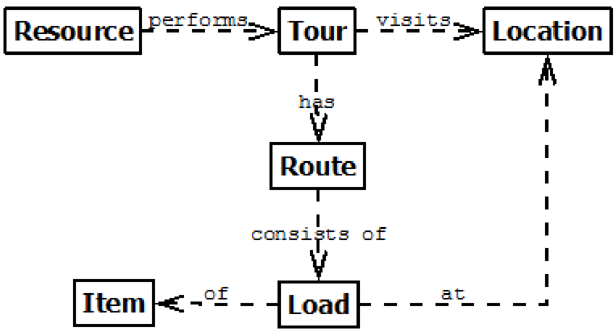

Figure 3: Deliveries

These jobs are then processed by the carrier, as shown in Figure 3. A tour visits certain locations using resources (e.g. vehicles, drivers) in no particular order. Thus, for a given tour multiple alternative routes can exist. A route is a tour where the locations are visited in a particular order. During a route multiple loads of items can be performed at certain locations.

For the classical VRP the customer makes orders (creates jobs) and provides target locations for them. The source location is determined by the depot. The model is generic enough to model other problem types like for example the dial-a-ride problem. In that case, jobs are created by the customers that specify both source and target locations in addition to a time window.

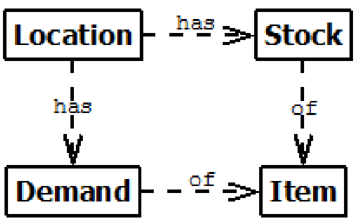

Figure 4: Vendor managed inventory extension

The domain model can also be extended to incorporate vendor managed inventory approaches. In that case a demand is calculated for every location using the current stock situation as illustrated in Figure 4. The jobs are not created by the customer, but by the carrier in that scenario.

Summarizing, because the domain model is generic, different variants of problems in the domain of transport logistics can be modeled.

3. SYSTEM ARCHITECTURE

Figure 5 illustrates the system architecture. The scenario generator creates scenarios which specify the problem environment. All master data and simulation parameters are included in a scenario specification.

The simulation loads this scenario and performs the simulation run. During a run, certain indicators and key figures can be written to a database. These indicators can be used during or after the simulation run to analyze the performance.

During the simulation run, problem instances are created for the simulation that can be optimized. In this case, VRP instances are parameterized by the simulation environment and sent to the optimization component.

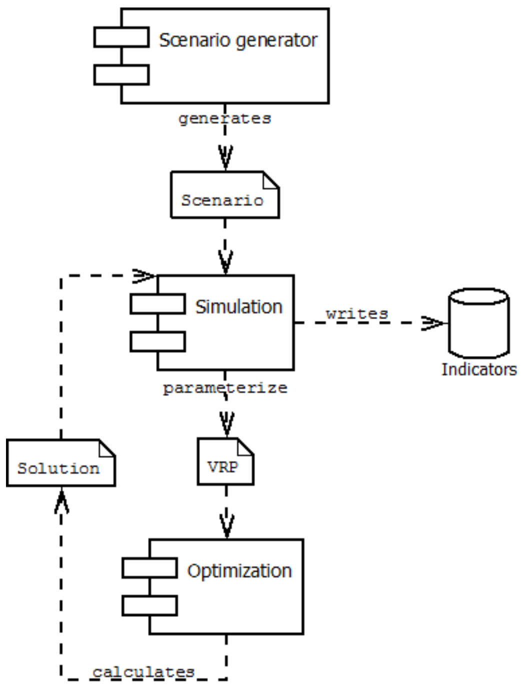

Figure 5: System Architecture

The components are examined individually in the following.

3.1. Scenario generator

As stated earlier, the scenario generator creates the problem environment. The master data is integrated into a single scenario file that can be loaded by the simulation component.

This can be achieved for example by integrating enterprise resource planning (ERP) systems or importing data from file formats like the XLS or CSV format.

Additionally, the scenario generator parameterizes the simulation run. Parameters could for example affect the customer ordering behavior or the delivery strategy.

Furthermore, the scenario generator can perform additional tasks like distance matrix generation or data preprocessing.

3.2. Simulation

To simulate the problem environment, an agent-based approach is followed. For the implementation the Repast.NET framework was used which is described by Vos (2005).

The different agents can perform certain actions at each time step of the simulation run. This enables the simulation of emergent behavior and the simulation environment can be modeled close to the actual problem scenario.

Figure 6 outlines the basic classes of the simulation environment and their relationships.

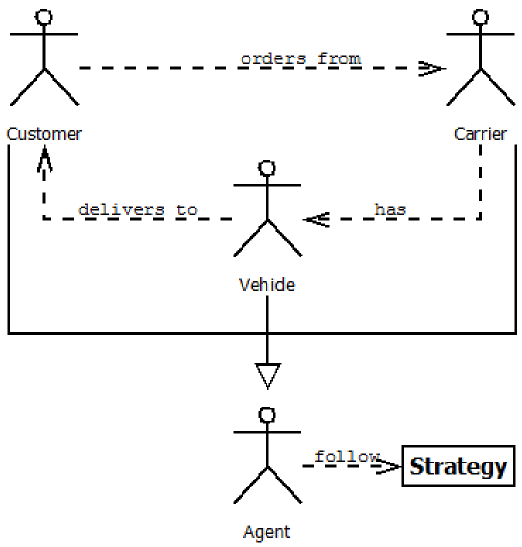

Figure 6: Simulation environment

According to the domain model, three types of agents are identified:

- Customer
- Carrier
- Vehicle

The customer agent simulates the customer behavior, like the customer ordering strategy. A customer can have inventories at different locations and the demand at each location can vary.

The delivery strategy is simulated by the carrier agent. A carrier receives orders from customers and controls a fleet of several vehicles. A carrier can have multiple depots.

The vehicle agent delivers the goods to the customer and receives orders from the carrier agent. It can choose between several alternative routes when delivering the goods to the customers, for example to avoid traffic jams.

All agents follow different strategies of performing certain actions. For example, customers could have diverse ordering strategies that can be parameterized in the scenario. Another example would be different delivery strategies by carriers.

Those strategies can be coupled to the optimization component. In that case, an optimization model like the VRP is parameterized by the agent and sent to the optimization component in order to decide what actions to perform next.

This interaction between the simulation and optimization components is defined by a generic interface. Each agent strategy can use this interface to parameterize different optimization models and use the optimization engine to make certain decisions.

3.3. Optimization

As stated before, the simulation generates scenarios by parameterizing a generic VRP model. This approach is similar to the approach followed by Beham, Kofler, Wagner and Affenzeller (2009).

The optimizer solves the problem and generates solutions for the different scenarios, which are then evaluated by the simulation. For developing the optimization solution, the extensible and generic framework HeuristicLab is used (Wagner, Winkler, Braune, Kronberger, Beham, Affenzeller 2007; Wagner 2009).

A core feature of HeuristicLab is the separation between problems and algorithms. This enables different metaheuristic optimization algorithms to be used and tested in a simulation run. As stated before, different optimization algorithms can be incorporated as strategies into the simulation environment and compared with each other.

In this particular scenario, the encoding proposed by Alba and Dorronsoro (2004) was used as a representation for the VRP, which is illustrated in Figure 8. Each route is separated by the vehicle number. The vehicles numbers start at the city count. For a VRP instance with 10 cities, the first vehicle number is 10.

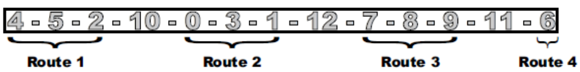

Figure 8: Problem representation (taken from Alba and Dorronsoro (2004))

Every solution is represented as a permutation, so standard permutation operators like the insertion, swap, inversion or translocation manipulation or edge recombination crossover can be used.

In order to generate solutions efficiently, so a simulation run does not take too long, a limit of one Minute was set for the optimization algorithm and the performance of several algorithms was compared. All tests were executed using HeuristicLab 3.3.0 (http://dev.heuristiclab.com) and a Core2Duo E7600 with 4 GB RAM under Windows 7 64 bit.

Different trajectory based search strategies were tested, once with a randomly generated initial solution and once generated by the push forward insertion heuristic as proposed by Thangiah (1999). As a neighborhood operator the translocation operator is used, which is described by Michalewicz (1992). The different algorithm configurations are listed in Table 1.

Table 1: Algorithm configurations

LSr (Local search with random initial solution)	
SampleSize	100
LSh (Local search with heuristic initial solution)	
SampleSize	100
SAr (Simulated annealing with random initial solution)	
StartTemperature	4000
EndTemperature	1E-06
InnerIterations	100
SAh (Simulated annealing with heuristic initial solution)	
StartTemperature	100
EndTemperature	1E-06
InnerIterations	100
TAr (Tabu search with random initial solution)	
SampleSize	500
TabuCriterion	Hard
TabuTenure	5
TAh (Tabu search with heuristic initial solution)	
SampleSize	1500
TabuCriterion	Soft
TabuTenure	50

The goal is to generate a feasible solution for the r102 instance from the Solomon benchmark library (http://www.idsia.ch/~luca/macs-vrptw/problems/welcome.htm) within one minute (250000 evaluations). This instance consists of 100 customers and is close to the test instance used in the results section. The quality value was calculated as following:

$$\text{quality} = \text{distance} + \text{vehicles} * 100 \qquad (1)$$

Ten independent test runs were performed and the results are listed in Table 2.

Table 2: Algorithm Runs

Algo	Mean distance	Mean vehicles	Mean quality	Stdev quality
LSr	1786.58	22.40	4026.58	138.59
LSh	1750.71	21,40	**3890,71**	**85,59**
SAr	1780,27	21,70	3950,27	108,51
SAh	1744,94	21,80	3924,94	91,71
TAr	1815,37	22,50	4065,37	96,24
TAh	1772,59	21,30	3902,59	115,60

Basically it can be stated that the heuristic initialization improves the solution quality. The local search algorithm performs best in that configuration and has both the best mean quality and the least variation.

4. RESULTS

To validate the simulation system, a real-world scenario has been implemented and tested. The geocoding and distance matrix generation for the locations was performed using the open route service (http://www.openrouteservice.org) which is described by Neis and Zipf (2008).

The concrete scenario consists of

- One carrier
  - One depot
  - Homogenous fleet of 50 vehicles
- 84 customers
  - 1 site

There are 84 customers with one site. Each of the customer sites has a known demand that is based on real data. The customer ordering behavior is simulated by drawing from a normal distribution.

The atomic simulation step is one day, which means that the VRP model is parameterized and solved for each day.

Figure 8 shows a screen shot of a simulation run. On the left-hand side a map is displayed which shows the customer locations. Customers who have a demand in the current day are marked as red. On the right hand side several indicators are displayed. They can be used for the analysis of a simulation run. Example indicators displayed here are the traveled distance or the vehicle utilization.

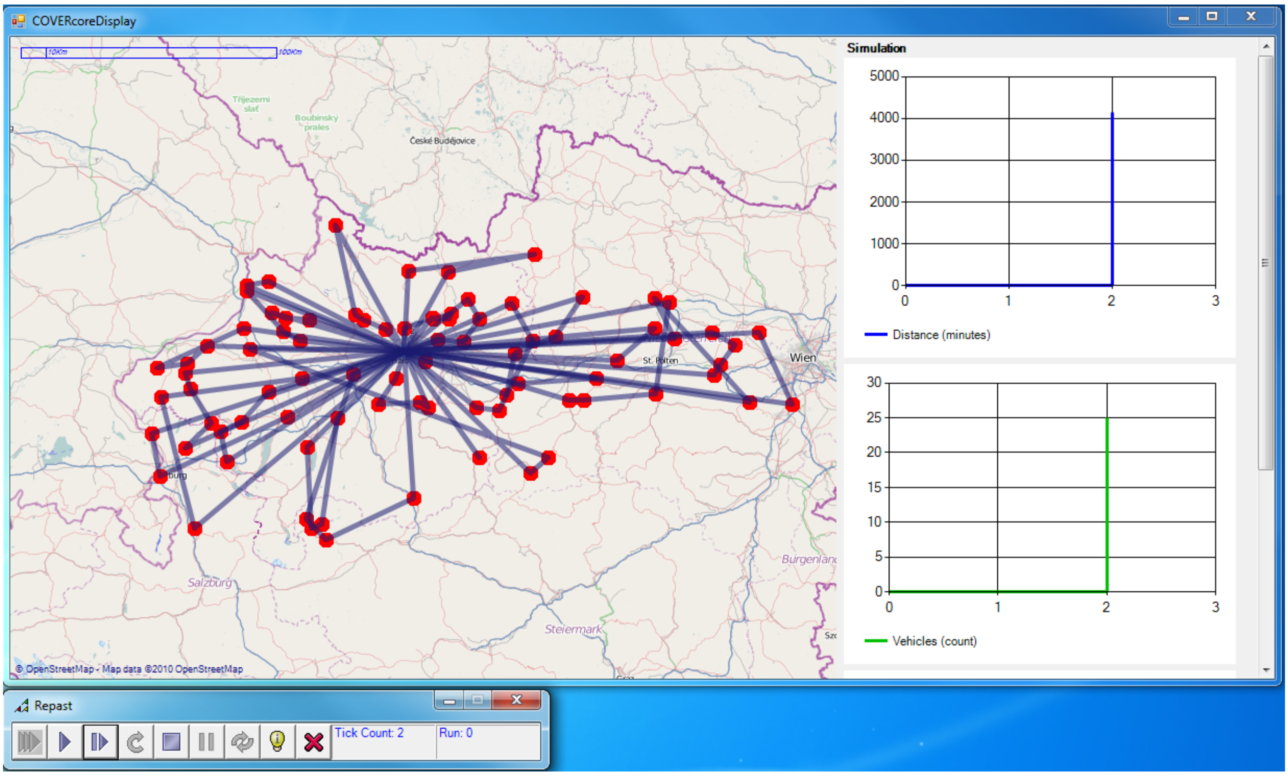

Figure 8: A simulation run

5. CONCLUSION

Concluding, a particular real-world scenario has been implemented in the simulation environment. The scenario was optimized with combination of an efficient heuristic and a metaheuristic optimizer using a permutation encoding.

In the future, different algorithms and encodings can be integrated and tested in the environment. Using the simulation system their performance for a particular scenario can be evaluated.

Additionally, different scenarios can be examined. Examples would be to examine different customer ordering behaviors, or to simulate seasonal fluctuations and evaluate how certain algorithms adapt to it.

A concrete example that will be examined is a mixed vendor-managed inventory approach to a real-world problem environment. The goal is to evaluate, how additional degrees of freedom in conjunction with fixed orders can affect the efficiency of delivery strategies.

Possible questions answered are for example:

- How much is the benefit of additional degrees of freedom in comparison to fixed orders according to costs, amount delivered and time delivered?

- In what scenarios are vendor managed inventory approaches beneficial and evaluation of mixed scenarios where some goods are ordered by the customer and some are managed by the vendor?

To sum things up, a flexible and extensible simulation system has been created that can incorporate diverse problem scenarios in the field of transport logistics as well as different algorithmic concepts to solve them and can be used to examine and compare them.

ACKNOWLEDGMENTS

The work described in this paper was done within the *Regio 13* program sponsored by the *European Regional Development Fund* and by Upper Austrian public funds.

REFERENCES

Alba, E.; Dorronsoro, B. 2004. *Solving the Vehicle Routing Problem by using Cellular Genetic Algorithms*. In: Barnhard, C.; Laporte, G. ed. *Handbooks in Operations Management & Management Science: Transportation*. Massachusetts Institute of Technology, USA. Chapter 6.

Beham A., Kofler M., Wagner S., Affenzeller M., 2009. Coupling Simulation with HeuristicLab to Solve Facility Layout Problems. *Proceedings of the 2009 Winter Simulation Conference*, pp. 2205-2217. December 13-16, Austin (Texas, USA).

Cordeau, J.-F.; Laporte, G.; Savelsbergh, M.W.P.; Vigo, D. 2005. Vehicle Routing. In: *Evolutionary Computation in Combinatorial Optimization (EvoCOP), Lecture Notes in Comput. Sci. vol. 3004*, Springer, pp. 11–20.

Neis, P., Zipf, A. 2008. *Zur Kopplung von OpenSource, OpenLS und OptenStreetMaps in OpenRouteService.org*. AGIT 2008. Symposium für angewandte Geoinformatik. Salzburg. Austria.

Michalewicz, Z. 1992. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer.

Thangiah, S.R. 1999. *A Hybrid Genetic Algorithms, Simulated Annealing and Tabu Search Heuristic for Vehicle Routing Problems with Time Windows*. In: *Practical Handbook of Genetic Algorithms, Volume II: Complex Structures.* Pages 347-381.

Vos, J.R. 2005. Repast .NET: The Repast Framework Implemented in the .NET Framework. *Proceedings of the 2005 NAACSOS Conference*. June 26-28, Notre Dame (Indiana, USA).

Wagner, S.; Winkler, S.; Braune, R.; Kronberger, G.; Beham, A. 2007. Benefits of plugin-bases heuristic optimization software systems. *Computer Aided Systems Theory - EUROCAST Conference*, pp. 747-754.

Wagner, S. 2009. *Heuristic optimization software systems – Modeling of heuristic optimization algorithms in the HeuristicLab software environment*. Thesis (PhD). Johannes Kepler Univiersity, Linz, Austria.

AUTHORS BIOGRAPHY

STEFAN VONOLFEN studied Software Engineering at the Upper Austrian University of Applied Sciences, Campus Hagenberg and received his MSc in engineering in 2010. Since January 2010 he works at the Research Center Hagenberg of the Upper Austrian University of Applied Sciences. His research interests include transport logistics optimization and simulation-based optimization.

STEFAN WAGNER received his MSc in computer science in 2004 and his PhD in engineering sciences in 2009, both from Johannes Kepler University (JKU) Linz, Austria; he is professor at the Upper Austrian University of Applied Sciences (Campus Hagenberg). Dr. Wagner's research interests include evolutionary computation and heuristic optimization, theory and application of genetic algorithms, machine learning and software development.

ANDREAS BEHAM received his MSc in computer science in 2007 from Johannes Kepler University (JKU) Linz, Austria. His research interests include heuristic optimization methods and simulation-based as well as combinatorial optimization. Currently he is a research associate at the Research Center Hagenberg of the Upper Austria University of Applied Sciences (Campus Hagenberg).

MONIKA KOFLER studied Medical Software Engineering at the Upper Austrian University of Applied Sciences, Campus Hagenberg, Austria, from which she received her diploma's degree in 2006. She is currently employed as a research associate at the Research Center Hagenberg and pursues her PhD in engineering sciences at the Johannes Kepler University Linz, Austria.

MICHAEL AFFENZELLER has published several papers and journal articles dealing with theoretical aspects of evolutionary computation and genetic algorithms. In 2001 he received his PhD in engineering sciences from JKU Linz, Austria. Dr. Affenzeller is professor at the Upper Austria University of Applied Sciences, Campus Hagenberg, and head of the Josef Ressel Center *Heureka!* at Hagenberg.

EFREM LENGAUER received his PhD in economics in 2002 from JKU Linz, Austria. He is Professor of Logistics Management at the Upper Austrian University of Applied Sciences, Campus Steyr. Dr. Lengauer has published several papers dealing with transportation management and distribution network design. His research interests include heuristic optimization and simulation of logistical problems.

MARIKE SCHEUCHER received her MSc in business sciences in 2003 and her PdD in logistics in 2007, both from Johannes Kepler Univerity (JKU) Linz, Austria. She is research associate at the Logistikum.research of the Upper Austria University of Applied Sciences (Campus Steyr). Her research interests include distribution logistics, especially transport logistics.

The Web-pages of the authors as well as further information about HeuristicLab and related scientific work can be found at http://heal.heuristiclab.com/ and information about the Logistikum can be found at http://www.logistikum.at.