Proceedings of the 2009 Winter Simulation Conference
M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin and R. G. Ingalls, eds.

AN ANT COLONY OPTIMIZATION APPROACH TO SOLVE COOPERATIVE TRANSPORTATION PLANNING PROBLEMS

Ralf Sprenger
Lars Mönch

Chair of Enterprise-wide Software Systems
Dept. of Mathematics and Computer Science
University of Hagen
Hagen, 58097, GERMANY

ABSTRACT

In this paper, we suggest efficient heuristics to solve a cooperative transportation planning problem that is motivated by a scenario found in the German food industry. After an appropriate decomposition of the entire problem into sub problems, we obtain a set of rich vehicle routing problems (VRP) including due dates for the delivery of the orders, capacity constraints and maximum operating time window constraints for the vehicles, and outsourcing options. Each of these sub problems is solved by a greedy heuristic that takes the distance of the locations of customers and the time window constraints into account. The greedy heuristic is further improved by applying an Ant Colony System (ACS). The suggested heuristics are assessed in a rolling horizon setting using discrete event simulation. The results of some preliminary computational experiments are provided. We show that the ACS based heuristic outperforms the greedy heuristic.

1 INTRODUCTION

Transportation planning problems are important for the German food industry. We consider a real-world scenario where several manufacturers with same customers but complementary food products collaborate by jointly using their vehicle fleets to reduce delivery costs. The different products are delivered to first-class hotels. Therefore, small delivery quantities are typical and high on-time delivery performance is also an important goal. There are different types of deliveries depending on the geographical location of the customers and the transportation capacities of each manufacturer. In the simplest case, each manufacturer uses its own local vehicle to deliver products to customers that are closely located to the manufacturer. In some situations, some of the customers are far away. Vehicles of a specialized shipping company are used to send the transportation orders to an intermediate distribution center (IDC) where own vehicles of the manufacturer are used to deliver the transportation orders to the customers. The third way consists of using the vehicles of a specialized shipping company to send the transportation order directly to the customers.

There are several ways to improve the efficiency of the transportation operations in the present situation. It is possible to use the vehicles of the different manufacturers in each IDC to deliver the transportation orders of all manufacturers to the corresponding customers. A cooperative strategy may consist in sending transportation orders of one manufacturer to another manufacturer and then using the local vehicles of the second manufacturer to deliver the products to the customers of the first manufacturer.

The present authors suggest in the previous paper (Sprenger and Mönch 2008) a simulation framework to assess the performance of algorithms to solve the cooperative transportation planning problem. However, only a very simple heuristic and some computational experiments, mainly to demonstrate the feasibility of the simulation-based approach, are described. In the present paper, we present a more sophisticated heuristic that is based on decomposition and ACS to solve the cooperative transportation planning problem.

The paper is organized as follows. In the next section, we describe the problem and discuss related literature. We suggest a decomposition scheme and heuristics to solve the overall cooperative transportation planning problem in Section 3. Finally, we provide the preliminary results of a simulation based performance assessment of the suggested heuristics within a rolling horizon setting.

2 PROBLEM SETTING

2.1 Problem Description

We consider a scenario where M manufacturers produce different types of foods. Each manufacturer $1 \le m \le M$ has v_m own vehicles. A single vehicle has a maximum capacity Q and a maximum operating time per day. Each vehicle has a home depot. This is either the location of the manufacturer or an IDC. A transportation order $j \in O$ is a quantity of products that is sent to customers to fulfill customer orders. A time window $\left[0, d_j\right]$ is assigned to j where it has to be delivered. Furthermore, each order has a ready time r_j where it is ready of transportation. Transportation orders are assigned to vehicles. Each vehicle has to visit several customers to deliver its transportation orders. This is called a sub tour. The vehicle returns to its home depot after the completion of a sub tour. The collection of all sub tours within the maximum operating time is called a tour. We differentiate three different cases for the organization of the corresponding deliveries:

1. A manufacturer uses its own local vehicle to deliver products to customers that are closely located to the manufacturer. The location of a local vehicle is the location of the manufacturer. This type of delivery is called delivery with own local vehicles.
2. Customers might be far away. In this situation, vehicles of a specialized shipping company are used to send the transportation orders to an IDC where own vehicles of the manufacturer are used. This type of delivery is called delivery with own far away vehicles.
3. Using the vehicle of a specialized shipping company to send the transportation order directly to the customers is called delivery with couriers.

Manufacturers, IDC, and customers form a distribution network. We are interested in determining feasible transportation plans for all the manufacturers with small transportation costs. The delivery with local vehicles of the manufacturers is the cheapest variant whereas the delivery with couriers is the most expensive one. The delivery with own far away vehicles is somewhere between these two variants.

The efficiency of the transportation operations in the present situation can be improved by using the vehicles of the different manufacturers in each IDC to deliver the transportation orders of all manufacturers to the customers. Sending transportation orders of one manufacturer to another one and then using the local vehicles of the second manufacturer to deliver the products to the customers of the first manufacturer is a second possibility to improve the performance of the overall distribution network. In summary, we consider a VRP with paired pickup and delivery points, time windows, outsourcing options and interim storages.

2.2 Related Research

ACO is a nature inspired approach that can improve construction heuristics for a given combinatorial optimization problem significantly. ACO approaches are quite popular for solving different types of VRP (cf. Bullnheimer, Hartl, and Strauss 1997, Doerner et al. 2006, Favaretto, Moretti, and Pellegrini 2007, Pellegrini, Favaretto, and Moretti 2007, Tan et al. 2005, Gajpal, and Abad 2009 amongst others). The suggest algorithms mainly differ in the used heuristic information that is typically problem specific. Among the ACO type heuristics ACS leads often quickly to good solutions (cf. Gajpal and Abad 2009). Bouhafs, Hajjam, and Koudam (2006) decompose the overall problem into sub problems using simulated annealing and then use ACO to solve each of the sub problems individually.

Cooperative transportation planning strategies are not discussed in the literature expect the papers by Lin (2008) and by (Sprenger and Mönch 2008). The use of rolling horizon schemes for VRP and their performance assessment by discrete event simulation is rarely discussed in the literature. However, it enables us to make more real-world type modeling assumptions like varying transportation times because of traffic jams or vehicle breakdowns and provides generally a more detailed picture of the performance of a heuristic to solve VRP (cf. Simroth and Baumbach 2007 and Sprenger and Mönch 2008).

In (Sprenger and Mönch 2008), we present a simple construction heuristic. Therefore, it is quite naturally to improve this heuristic by ACO type approaches. While Ant Colony Optimization (ACO) approaches are used quite often to solve vehicle routing problems to our best knowledge there is no work described in the literature that uses ACS type approaches for cooperative transportation planning within a rolling horizon scheme.

3 HEURISTIC APPROACHES

3.1 Decomposition Approach

We segment the distribution network into different zones to decompose the overall transportation planning problem as described in (Sprenger and Mönch 2008). A single zone is obtained by assigning each customer to the nearest manufacturer or an intermediate distribution center. A VRP has to be solved for each of the zones.

3.2 Overall Scheme

We describe how we construct tour plans. Therefore, we start with choosing the next available vehicle. Then we select an order to assign it to the vehicle. Only orders are considered that take into account the different vehicle related constraints described in Section 2.1. The heuristics, presented in Sub Section 3.2.1 and 3.2.2, are applied to choose an appropriate next order. The time window constraint of the orders is ensured as follows. We select only those orders j that do not meet d_j when there is no possibility to load the particular order to a different vehicle in order to avoid a late delivery. We denote the set of all orders that are not part of a fully or partially constructed tour and that do not lead to a constraint violation by Ω. Note that we start first with own local vehicles, than with own far away vehicles, and finally with courier vehicles according to the cost structure described in Section 2.1 (see the more detailed description in Section 4.1).

3.2.1 Greedy Heuristic

To choose the next order we use a greedy heuristic that takes the distance of the locations of customers and the time window constraints into account. We use the following index

$$I_{ij}(t) := \frac{1}{t_{ij}} e^{-\frac{\max(d_j - t_{ij} - t, 0)}{\kappa \bar{t}}}, \tag{1}$$

where i denotes the current order on the vehicle, j is a candidate order, t_{ij} is the time that is required to drive from the customer related to order i to the customer related to order j. The notation t is used for the current time, and $\bar{t}$ is the average time that is needed to drive from a first customer to a second one. The order j with the largest index (1) is selected. The first term of the index takes the distance between the customers of order i and j into account whereas the second part is used to assure that orders that are close to their due date are preferred. κ is a scaling parameter for blending the two parts within the index. Its appropriate setting is crucial for the performance of the heuristic. A grid search approach is taken to determine the κ value that leads to best tour plans. Therefore, we consider different $\kappa \in (0,1000]$ from a grid with step size 0.1.

Note that the form of the index (1) is influenced by the Apparent Tardiness Cost (ATC) dispatching rule in manufacturing. It is well-known that the application of ATC type rules leads to good on-time delivery performance (cf. Pinedo 2002). The notation GH is used for abbreviation.

3.2.2 ACS Type Heuristic

The main idea of ACS consists in a set of artificial ants. Each ant starts with an empty assignment of the orders to vehicles and constructs sub tours for the single vehicles by adding iteratively one of the remaining orders to one of the partial sub tours found so far. The search for good solutions of our VRP is coupled between the different ants by artificial pheromone trails. The ants communicate indirectly by modifying the pheromone trails after the construction of a new solution for the VRP. The solutions found by the ants can be improved by local search techniques. The overall scheme of an ACO algorithm is given as follows (Dorigo and Socha 2007):

1. Set parameters and initialize pheromone trails.
2. Construct a new solution of the VRP by creating a new ant. This ant takes the pheromone trail information into account during its construction of a solution.
3. Improve this solution by local search.
4. Update pheromone trails by using the solution obtained by the ants from Step 2 and 3.
5. When the termination criterion is reached then stop, otherwise go to Step 2.

In the remainder of this section, we describe Step 2 and Step 4 in more detail. We denote by η_{ij} the heuristic desirability of delivering order *j* immediately after order *i*. The η_{ij} values are typically derived by using appropriate, problem specific construction rules. In our experiments, we use the index (1) in order to provide the heuristic information within the ACS approach.

We denote by $\tau_{ij}(t)$ the pheromone intensity that is associated with the selection of order *j* immediately after order *i*. The parameter *t* is used to denote the current iteration of the ACO scheme.

A single iteration of the ACS is described. We assume that order *i* is the last selected order and we want to choose the successor order *j*. We create a sample of a $U[0,1]$ distributed random variable. Denote the obtained value by q_0. When $q_0 \le q$ then the order $j \in \Omega$ is selected that maximizes the value of $\tau_{ij}\eta_{ij}$. Here, $q \in (0,1)$ is a given parameter of the ACS scheme. When $q_0 > q$ then job $j \in \Omega$ is selected according to the following discrete distribution with the probabilities p_{ij} given by

$$p_{ij} := \begin{cases} \dfrac{\tau_{ij}\eta_{ij}}{\sum\limits_{h\in\Omega}\tau_{ih}\eta_{ih}} & \text{if } j \in \Omega \\ 0 & \text{otherwise} \end{cases} . \tag{2}$$

We start with making a few runs with different values for the scaling parameter κ. Then we use the value for κ that leads to tour plans with the smallest costs. We initialize the pheromone intensities τ_{ij} by choosing the reciprocal value of the smallest cost for tour plans found by using GH. A local update of the pheromone intensities is performed by the expression

$$\tau_{ij} := (1-\rho)\tau_{ij} + \rho\tau_0 , \tag{3}$$

after each insertion of a new order *j*. τ_0 is the initial pheromone value. $\rho \in (0,1]$ is a parameter of ACS that affects the balance between exploration and evaporation. When the ant has constructed the tour plan, a local search heuristic is applied to improve the solution obtained by the ant. We use a 3-opt type heuristic. When all ants within one iteration have computed a tour plan, a global update of the pheromone values is performed. We apply the update equation

$$\tau_{ij}(t+1) := (1-\alpha)\tau_{ij}(t) + \alpha/\cos t^* , \tag{4}$$

for orders *i* and *j* that are selected in a consecutive manner in the global best solution found by the ants. $\cos t^*$ is the associated cost. We also may allow the iteration best ant to deposit additional pheromone. In our experiments, we use a mixed strategy by using the iteration best ant for global update after five consecutive iterations. The quantity $\alpha \in (0,1]$ is a parameter of the ACS scheme.

3.3 Comparison of the Two Approaches

In a first step, we apply GH and ACS to the orders and vehicles within one single zone to assess the performance of the ACS approach. A tour plan for the full day is computed. We choose $\alpha = \rho = 0.9$ and $q = 0.95$ based on extensive computational experiments. Ten independent replications are performed and average values for the performance measures are calculated to obtain stochastically significant results. The results for 50 and 100 orders, a vehicle capacity of 10 or an unlimited number of orders and different due date settings are shown in Table 1. We use 50 ants for a single iteration. At maximum 200 consecutive iterations are allowed.

The iteration where the best solution is found for the first time increases with a larger number of orders. Consequently, the ACS has to run longer to find acceptable solutions that outperform the GH results. The results in Table 1 demonstrate that it makes sense to spend this additional time for computing because ACS outperforms GH in all the tested scenarios. GH requires on average 30 seconds per instance while ACS requires additional 60 seconds for computing tour plans for a total of 100 orders. In a next step, we have to research the impact of ACS and GH in a rolling horizon setting.

Table 1: Results for GH and ACS for one Single Zone

Orders	Capacity	Due Dates	Cost GH [km]	Cost ACS [km]	Improvement	Iteration with best solution
50	10	tight (~ 24% late)	4250	3521	16.9%	55
50	10	moderate (~ 14% late)	3677	3326	8.9%	87
50	10	wide (~ 8% late)	3288	2769	15.7%	106
50	∞	tight	3990	3586	10.0%	71
50	∞	moderate	3595	3069	14.1%	86
50	∞	. wide	3195	2881	9.6%	86
100	10	tight	6255	5492	12.0%	116
100	10	moderate	5794	5527	4.5%	119
100	10	wide	5216	4667	10.3%	143
100	∞	tight	6086	5731	5.6%	106
100	∞	moderate	5544	5172	6.5%	162
100	∞	wide	4640	4258	8.6%	153

4 SIMULATION-BASED PERFORMANCE ASSESSMENT OF THE HEURISTICS

We are interested in assessing the performance of the suggested heuristics in a real world scenario including some stochastic effects that may influence the execution of the computed tour plans.

4.1 Scenarios and Simulation Framework

We consider two manufacturers similar to the situation described by Sprenger and Mönch (2008). Each of them uses two IDC that are delivered by vehicles from shipping companies. The planning scenario is based on a real-world data set that is collected in the German food industry. The considered customer network consists of 550 different customers. We simulate 100 days in all the scenarios.

The transportation orders used in the simulation experiments are generated in the following way. Incoming orders are concentrated in the morning. Therefore, we use for r_j the expression

$$r_j \sim U\left[24k, 24k+6\right], k = 0,\ldots,100, \tag{5}$$

where the value six controls how early in the morning the transportation orders are ready. Due dates are chosen in a similar way. We use the following expression for d_j

$$d_j - r_j \sim U\left[shift, shift + \beta u\right], \tag{6}$$

where the quantity *shift* denotes the minimum due date. The quantity u is an appropriate upper bound for the due dates. The parameter $\beta \in (0,1]$ controls the tightness of the due dates. Here, we use the values $u = 2$ days and vary the values of β in our experiments. The quantity *shift* has been set to one day to avoid orders that are not deliverable due to the static transportation between the zones. In addition we assure that the d_j are being within the driving time windows of the vehicles.

In addition, we want to show the improvements when the two manufacturers act in a cooperative manner and share their distribution network similar to the scenarios investigated in (Sprenger and Mönch 2008). In the cooperative case, manufacturer 1 has ten vehicles at its location, but manufacturer 2 only two vehicles. Four and two vehicles are assigned to the IDC respectively. We release equally distributed between 120 and 180 orders per day into the distribution network. The orders are equally distributed among the two manufacturers. Based on some preliminary simulation results, this number of vehicles turns out to be the minimum number of vehicles that is necessary to deliver most of the orders at almost all days for a workload of 150 orders per day. Consequently, 19 vehicles are used in the cooperative case.

In the non cooperative case, we have to change the number of vehicles to 21 due to the changed size of the zones. In the cooperative case, there are four zones with vehicles that can be used for transportation by both manufacturers. In the non cooperative scenario, each of the manufacturers can use only three zones, i.e., the zone that contains the location of the manufacturer and the zones of the two IDC. We also modified the location of the vehicles. Ten vehicles are assigned to the location of manufacturer 1, two and four to the two IDC respectively. Manufacturer 2 runs three vehicles at its location and seven and two at the corresponding IDC.

The maximum operating time for vehicles is ten hours per day. The vehicles start in the morning at 8 am and drive until 6 pm. Incoming orders during the day will not be taken into account. The new tour plans for the next day are computed at 7 pm. This is done in the following way.

1. Ready orders at a location of a manufacturer are assigned to one of the zones. After this assignment, we compute plans for external transportation between the locations of the manufacturers and IDC. We assume that the shipping company can deliver the orders to their target zone over night. Consequently, we know which orders are available next morning in each of the zones.
2. A tour plan is computed for each of the zones using the two suggested heuristics. We enforce that each order is part of the tour plan and that the expected delivery time of the orders is smaller than its due date by penalizing each not delivered order with a very high cost value. In addition, orders that are part of the tour plan, but their real delivery time is too late are penalized with a lower costs value. This assures that the large costs for direct transportation or losing a customer by violating the delivery date are avoided without regard to the additional km that must be driven to delver the order by own vehicles.
3. Orders that are not part of the tour plan are assigned to external suppliers that transport them directly to the customer over night.
4. Orders with expected delivery dates that do not meet the due date are iteratively removed from the tour plan. We start with the order with the strongest due date violation, assign this order to a courier, and update the expected delivery times for the remaining orders in the tour plan. When the plan still contains some orders that are late, we repeat the removal procedure until the tour plan contains only orders that meet the due dates. This step is optional. Repairing the tour plan after completing the computation reduces the number of not delivered orders when using ACS. This step is not applied when GH is used because the results are better if an order that will be late is not taken into account for insertion while computing tour plans.
5. Orders that are assigned for transportation by couriers are removed from the IDC transportation list that has been computed in Step 1.
6. The transportation plan for IDC transportation computed in Step 1 and transportation by couriers computed in Step 5 are immediately executed. At 8 am in the morning the tour plans for the vehicles at the locations of the manufacturers and IDC are recomputed and executed. Recomputation of the tour plans is necessary because some vehicles may not available in the morning due to breakdowns.

We set the maximum number of orders per vehicle to 15, whereas the number of orders per external supplier for IDC transportation and also for the transportation by couriers is unlimited. We use 50 ants for each of the 100 iterations. The remaining parameters of GH and ACS are selected as described in Sub Section 3.2.1 and 3.2.2.

We use the simulation framework described in (Sprenger and Mönch 2008) that mimics the behavior of the real-world distribution network. We include breakdowns of vehicles with a given probability at the beginning of the day, deviations of the deterministic driving times, and also some deviations from the deterministic load and unload times of orders modeled by appropriate uniform distributions.

The two heuristics compute tour plans according to deterministic driving times. When the operating time window of a vehicle is exceeded by some stochastic effects, then the vehicle drives to its depot and unloads all of its not delivered orders. These orders have to be inserted into a later computed tour plan. In order to obtain stochastically significant results, we repeat all simulation runs three times and take average values of the performance measures.

4.2 Results of Computational Experiments

The results for $\beta = 0.5$ and $\beta = 1.0$ are shown in Tables 2 and Table 3. The ACS approach turns out to be about 10-15% better with respect to the number of driven km. This result is caused by a higher utilization of the vehicles of the manufacturers. When the due dates are tight, i.e. $\beta = 0.5$, the number of orders that do not meet the due dates is increased due to the fact that stochastic effects have a larger impact in this particular case.

The difference between the cooperative and the non cooperative case is still huge. We obtain about 40% more driven km for the vehicles of the manufacturers and courier km but half external supplier km due to the lower number of zones.

Table 2: Results for Tight Due Date Obtained by $\beta = 0.5$

		GH (cooperative)	ACS (cooperative)	ACS (non cooperative)
vehicle	km total	533,330	652,966	882,579
	average orders on vehicle	12	11	9
	km per order	42	49	72
	sub tours per day	10	12	13
	km per sub tour	562	547	721
	orders transported	12,586	13,463	12,291
external supplier (IDC)	km total	248,500	248,500	147,300
	average orders on vehicle	14	15	19
	km per order	30	28	19
	orders transported	8,356	8,949	7,705
external supplier (courier)	km total	808,244	562,943	1,099,374
	average orders on vehicle	1.07	1.06	1.06
	km per order	373	366	433
	orders transported	2,164	1,540	2,541
order	due date violation	3.32%	3.08%	3.95%

5 CONCLUSIONS AND FUTURE RESEARCH

In this paper, we presented algorithms for a cooperative transportation planning problem. We suggested a decomposition technique to assign orders to a given set of vehicles. Each of these sub problems is solved by a greedy heuristic that takes into account the distance of the customers and the due dates for the orders. This heuristic is improved by ACS. We conducted some preliminary simulation experiments that demonstrated that the ACS type outperforms the greedy heuristic.

Table 3: Results for Wide Due Dates Obtained by $\beta = 1.0$

		GH (cooperative)	ACS (cooperative)	ACS (non cooperative)
vehicle	km total	621,952	719,448	935,886
	average orders on vehicle	13	11	10
	km per order	45	51	72
	sub tours per day	11	13	13
	km per sub tour	610	595	774
	orders transported	13,688	14,102	12,932
external supplier (IDC)	km total	248,500	248,500	147,300
	average orders on vehicle	16	15	22
	km per order	27	27	17
	orders transported	9,318	9,106	8,886
external supplier (courier)	km total	384,040	182,750	650,748
	average orders on vehicle	1.08	1.07	1.05
	km per order	321	329	397
	orders transported	1,196	556	1,640
order	due date violation	0.39%	0.46%	3.16%

There are some directions for future research. So far, we use a static decomposition method to assign orders to different zones in the distribution network. In order to make the decomposition more adaptive, we are working on the extension of the multi agent framework suggested by Mönch and Stehli (2006). The transportation planning for a zone can be represented by a decision-making agent. This agent is supported by a staff agent that computes solutions for the planning problem using suggested ACS algorithm. Orders will be exchanged between the zones to make the distribution to zones more dynamic and adaptive. Furthermore, we expect that we can use ideas from Montemanni et al. (2002) for the dynamic inclusion of new orders by using already existing pheromone values from the previous instance.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support from the German Federal Ministry of Economics (BMWi) within the project iCoTrans (grant 19G7022A).

REFERENCES

Bouhafs, L., A. Hajjam, and A. Koudam. 2006. A combination of simulated annealing and ant colony system for the capacitated location-routing problem. In *Proceedings International Conference on Knowledge-Based Intelligent Information and Engineering Systems* (LNCS 4251), 409–416.

Bullnheimer, B., R. F. Hartl, and C. Strauss. 1997. Applying the ant system to the vehicle routing problem. In *Proceedings 2nd International Conference on Metaheuristics*.

Doerner, K. F., R. F. Hartl, S. Benker, and M. Lucka. 2006. Ant colony system for a VRP with multiple time windows and multiple visits. *Parallel Processing Letters* 16(3):351–369.

Dorigo, M., and K. Socha. 2007. *Ant colony optimization*. Handbook on Approximation Algorithms and Metaheuristics (Chapter 26). New York: Chapman & Hall.

Favaretto, D., E. Moretti, and P. Pellegrini. 2007. Ant colony system for a vrp with multiple time windows and multiple visits. *Journal of Interdisciplinary Mathematics* 10:263–284.

Gajpal, Y., and P. Abad. 2009. An ant colony system (ACS) for vehicle routing problem with simultaneous delivery and pickup. To appear in *Computers and Operations Research*.

Lin, C. K. Y. 2008. A cooperative strategy for a vehicle routing problem with pickup and delivery time windows. *Computers and Industrial Engineering* 55: 766-782.

Mönch, L. and M. Stehli. 2006. ManufAg: a Multi-Agent-System Framework for Production Control of Complex Manufacturing Systems. *Information Systems and e-Business Management* 4(2):159–185.

Montemanni, R., L. M. Gambardella, A. E. Rizzola, and A. V. Donati. 2003. A new algorithm for a dynamic vehicle routing problem based on ant colony system. In *Second International Workshop on Freight Transportation and Logistics*, 27-30.

Pellegrini, P., D. Favaretto, and E. Moretti. 2007. Multiple ant colony optimization for a rich vehicle routing problem: a case study. In *Proceedings International Conference on Knowledge-Based Intelligent Information and Engineering Systems* (LNCS 4693), 627–634.

Pinedo, M. 2002. *Scheduling: theory, algorithms, and systems*. Prentice Hall, Second Edition.

Simroth, A., and B. Baumbach, 2007. Eine Planungs- und Experimentierumgebung für die Tourenplanung unter Unsicherheit. *Technical Report*, Fraunhofer Institut für Verkehrs- und Infrastruktursysteme Dresden.

Sprenger, R., and L. Mönch. 2008. A simulation framework for assessing the performance of cooperative transportation planning algorithms. In *Proceedings of the 2008 Winter Simulation Conference*, 2769–2776.

Tan, X., X. Luo, W. N. Chen, and J. Zhang. 2005. Ant colony system for optimizing vehicle routing problem with time windows. In *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation*, 209–214.

AUTHOR BIOGRAPHIES

RALF SPRENGER is a Ph.D. student in the Department of Mathematics and Computer Science at the University of Hagen, Germany. He received a master's degree in computer science at Dresden University of Technology in 2007. His current research interests are transport optimization and production control of semiconductor wafer fabrication facilities. His email address is <Ralf.Sprenger@fernuni-hagen.de>.

LARS MÖNCH is a Professor in the Department of Mathematics and Computer Science at the University of Hagen, Germany. He received a master's degree in applied mathematics and a Ph.D. in the same subject from the University of Göttingen, Germany. His current research interests are in simulation-based production control of semiconductor wafer fabrication facilities, applied optimization and artificial intelligence applications in manufacturing and logistics. He is a member of GI (German Chapter of the ACM), GOR (German Operations Research Society), SCS, INFORMS, and IIE. His email address is <Lars.Moench@fernuni-hagen.de>.