On the Interactive Resolution of Multi-objective Vehicle Routing Problems

Martin Josef Geiger and Wolf Wenger

Lehrstuhl für Industriebetriebslehre, Universität Hohenheim, 70593 Stuttgart, Germany
{mjgeiger,w-wenger}@uni-hohenheim.de
http://www.ibl.uni-hohenheim.de/

Abstract. The article presents a framework for the resolution of rich vehicle routing problems which are difficult to address with standard optimization techniques. We use local search on the basis on variable neighborhood search for the construction of the solutions, but embed the techniques in a flexible framework that allows the consideration of complex side constraints of the problem such as time windows, multiple depots, heterogeneous fleets, and, in particular, multiple optimization criteria. In order to identify a compromise alternative that meets the requirements of the decision maker, an interactive procedure is integrated in the resolution of the problem, allowing the modification of the preference information articulated by the decision maker. The framework is implemented in a computer system. Results of test runs on multiple depot multi-objective vehicle routing problems with time windows are reported.

Keywords: User-guided search, interactive optimization, multi-objective optimization, multi depot vehicle routing problem with time windows, variable neighborhood search.

1 Introduction

The vehicle routing problem (VRP) is one of the classical optimization problems known from operations research with numerous applications in real world logistics. In brief, a given set of customers has to be served with vehicles from a depot such that a particular criterion is optimized. The most comprehensive model therefore consists of a complete graph $G = (V, A)$, where $V = \{v_0, v_1, \ldots, v_n\}$ denotes a set of vertices and $A = \{(v_i, v_j) \mid v_i, v_j \in V, i \neq j\}$ denotes the connecting arcs. The depot is represented by v_0, and m vehicles are stationed at this location to service the customers $v_i, \ldots, v_n$. Each customer v_i demands a nonnegative quantity q_i of goods and service results in a nonnegative service time d_i. Traveling on a connecting arc (v_i, v_j) results in a cost c_{ij} or travel time t_{ij}. The most basic vehicle routing problem aims to identify a solution that serves all customers, not exceeding the maximum capacity of the vehicles Q_k and their maximum travel time T_k while minimizing the total distances/costs of the routes.

S. Obayashi et al. (Eds.): EMO 2007, LNCS 4403, pp. 687–699, 2007.

Various extensions have been proposed to this general problem type. Most of them introduce additional constraints to the problem domain such as time windows, defining for each customer v_i an interval $[e_i, l_i]$ of service. While arrival before e_i results in a waiting time, arrival after l_i is usually considered to be infeasible [1]. In other approaches, the time windows may be violated, leading to a tardy service at some customers. Violations of time windows are either integrated in the overall evaluation of solutions by means of penalty functions [2], or treated as separate objectives in multi-objective approaches [3].

Some problems introduce multiple depots as opposed to only a single depot in the classical case. Along with this sometimes comes the additional decision of open routes, where vehicles do not return to the place they depart from but to some other depot. Also, different types of vehicles may be considered, leading to a heterogeneous fleet in terms of the abilities of the vehicles.

Unfortunately, most problems of this domain are $\mathcal{NP}$-hard. As a result, heuristics and more recently metaheuristics have been developed with increasing success [4,5,6]. In order to improve known results, more and more refined techniques have been proposed that are able to solve, or at least approximate very closely, a large number of established benchmark instances [7]. It has to be mentioned however, that with the increasing specialization of techniques a decrease in generality of the resolution approaches follows. As a result, heuristic optimization frameworks such as HotFrame [8], EasyLocal++ [9] or ParadisEO [10] try to address this issue by providing generic libraries for the resolution of optimization problems.

While the optimality criterion of minimizing the total traveled distances is the most common, more recent approaches recognize the vehicle routing problem as a multi-objective optimization problem [11, 3, 12, 13, 14]. Important objectives besides the minimization of the total traveled distances are in particular the minimization of the number of vehicles in use, the minimization of the total tardiness of the orders, and the equal balancing of the routes. Following these objectives, it is desired to obtain solutions that provide a high quality of delivery service while minimizing the resulting costs. As many objectives are however of conflicting nature, not a single solution exists that optimizes all relevant criteria simultaneously. Instead, the overall problem lies in identifying the set of Pareto-optimal solutions P and selecting a most-preferred solution $x^* \in P$. In this context, three different general strategies of solving multi-objective optimization problems can be implemented:

1. *A priori* approaches reduce the multi-objective problem to a single-objective surrogate problem by formulating and maximizing a utility function. The advantage of this approach can be seen in its simplicity given the possibility to specify the precise utility function of the decision maker. The concept may however not be used if the decision maker is not able to state his/her preferences in the required way.
2. *A posteriori* approaches first identify the Pareto set P, and then allow the decision maker to select a most-preferred solution $x^* \in P$. The main advantage of this resolution principle is, that the computation of the optimal

solutions can be done offline without the immediate participation of the decision maker. A large number of elements of the Pareto set are on the other hand discarded later during the decision making procedure.
3. *Interactive* approaches allow the gradual articulation of preferences by the decision maker and compute a sequence of solutions based on his/her individual statements. Several advantages result from this concept. First, the computational effort is smaller in comparison to the identification of the entire Pareto set. Second, the gradual articulation of preferences allows the decision maker to reflect the chosen settings in the light of the obtained results and therefore adapt and react to the optimization procedure. A disadvantage of interactive multi-objective optimization procedures is however the need of the presence of a decision maker and the availability of an interactive software to present the results. Also, comparably little time for computations is allowed as the system should be able to react in (almost) real-time to inputs of the decision maker.

While it has been stressed already quite early, that combining computer programs with interactive planning procedures may be a beneficial way of tackling complex routing problems [15, 16, 17, 18], research in interactively solving multi-objective metaheuristics is a rather newly emerging field of research [19]. Given the increasing computing abilities of modern computers however, approaches can become increasingly interesting as they allow the resolution of complex problems under the consideration of interactive, individual guidance towards interesting solutions.

The article is organized as follows. In the following Section 2, a framework for interactive multi-objective vehicle routing is presented that aims to address two critical issues:

1. The necessary generality of resolution approaches when trying to solve a range of problems of different characteristics.
2. The integration of multiple objectives and the consideration of individually articulated preferences of the decision maker during the resolution procedure of the problem.

An implementation of the framework for multi-objective vehicle routing problems is presented in Section 3. The system is used to solve instances of multi-objective vehicle routing problems. Conclusions are presented in Section 4.

2 A Framework for Interactive Multi-objective Vehicle Routing

Independent from the precise characteristics of the particular VRP, two types of decisions have to be made when solving the problem.

1. Assignment of customers to vehicles (clustering).
2. Construction of a route for a given set of customers (sequencing).

It is well-known that both types of decisions influence each other to a considerable extent. While the clustering of customers to vehicles is an important input for the sequencing, the sequencing itself is of relevance when adding customers to routes as constraints of maximum distances have to be respected. The two types of decisions can be made either sequential (cluster first-route second vs. route first-cluster second) or in parallel.

Therefore, the framework presented here proposes the use of a set of elements to handle this issue with upmost generality. Figure 1 gives an overview about the elements used.

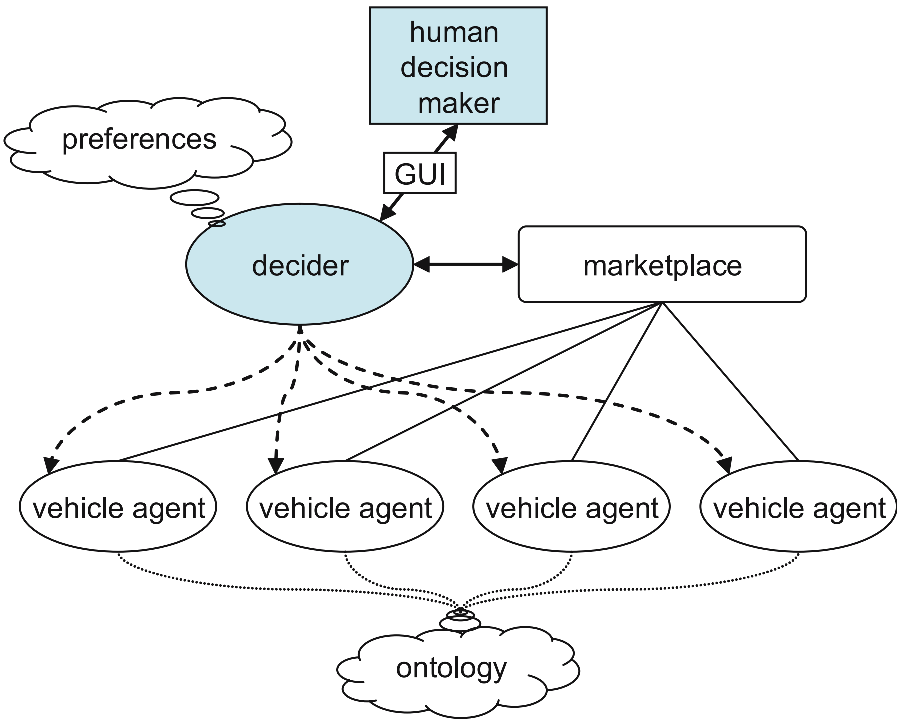

Fig. 1. Illustration of the framework for interactive multi-objective vehicle routing

- The *marketplace* represents the element where orders are offered for transportation. This elements is particularly necessary to allow an exchange of information gathered during the execution of the optimization procedure.
- *Vehicle agents* place bids for orders on the marketplace. These bids take into consideration the current routes of the vehicles and the potential change when integrating an additional order. Integrating additional orders into existing routes leads to an increase in terms of traveled routes and/or time window violations. This information is reported back to the marketplace.

- An *ontology* describes the precise properties of the vehicles such as their capacity, availability, current location, etc. This easily allows the consideration of different types of vehicles. It also helps to model open routes, where vehicle do not necessarily return to the depot where they depart from.
- A *decider* communicates with the human decision maker via a graphical user interface (GUI) and stores his/her individual preferences. In comparison to generic graphical user interfaces for multi objective optimization such as GUIMOO [20] we chose an approach that also visualizes the actual solution on a map, not only the evaluation of the currently considered solution.

  The decider also assigns orders to vehicles, taking into consideration the bids placed for the specific orders.

A solution is constructed by placing the orders on the marketplace, collecting bids from the vehicle agents, and assigning orders to vehicles while constantly updating the bids. Route construction by the vehicle agents is done in parallel using local search heuristics so that a route can be identified that maximizes the preferences of the decision maker. Reviewing possible ways of solving the clustering/sequencing problems, the presented approach follows the concept of combining both decisions in parallel.

In the proposed framework, the decision maker is allowed to change his/her preferences during the construction of the solution. If this happens, the *decider* updates the stored preference information and in consequence, the vehicles resequence their orders such that the updated preference information is met.

3 Implementation and Experimental Investigation

3.1 Configuration of the System

The framework has been implemented in a computer system. In the experiments that have been carried out, two objective functions are considered, the total traveled distances $DIST$ and the total tardiness $TARDY$ caused by vehicles arriving after the upper bound l_i of the time window. It should be noticed however, that neither the concept presented in Section 2 nor the actual implementation are restricted to two objective functions only. A sensible choice however had to be made in order to investigate the system in a quantitative way in a controllable experimental setting.

The preferences of the decision maker are represented introducing a weighted sum of both objective functions. Using the relative importance of the distances w_{DIST}, the overall utility UTILITY of a particular solution can be computed as given in Expression (1).

$$UTILITY = w_{DIST}\ DIST + (1 - w_{DIST})\ TARDY \qquad (1)$$

The vehicle agents are able to modify the sequence of their orders using four different local search neighborhoods.

- Inverting the sequence of the orders between positions p_1 and p_2, $p_1 \neq p_2$. While this may be beneficial with respect to the distances, it may pose a problem for the time windows as usually orders are served in the sequence of their time windows.
- Exchanging the positions p_1 and p_2, $p_1 \neq p_2$ of two orders.
- Moving an order from position p_1 and reinserting it at position p_2, $p_1 < p_2$ (forward shift).
- Moving an order from position p_1 and reinserting it at position p_2, $p_1 > p_2$ (backward shift).

In each step of the local search procedure, a neighborhood is randomly picked from the set of neighborhoods and a move is computed and accepted given an improvement. We select each neighborhood with equal probability of $\frac{1}{4}$.

Bids for orders on the marketplace are generated by the vehicle agents, taking into consideration all possible insertion points in the current route. The sum of the weighted increase in distance DIST and tardiness TARDY gives the prize for the order. This price reflects the individual preferences articulated by the decision maker using the w_{DIST} parameter which expresses the tradeoff between distances and time window violations.

The decider assigns orders to vehicles such that the maximum regret when *not* assigning the order to a particular vehicle, and therefore having to assign it to some other vehicle, is minimized. It also analyzes the progress of the improvement procedures. Given no improvement for a certain number of iterations, the decider forces the vehicle agents to place back orders on the market such that they may be reallocated. In the current setting, the vehicle agents are allowed to compute 1000 neighboring solution without any further improvements before they are contacted by the decider to place back one order on the marketplace. The order to be placed back is the one of the current route that, when removing it from the route, leads to the biggest improvement with respect to the overall evaluation of the route.

3.2 Experiments

The optimization framework has been tested on ten benchmark instances taken from [21]. The instances range from 48 to 288 customers that have to be served from 4 to 6 depots, each of which possesses 2 to 7 vehicles vehicles. The precise description of the instances is given in [21] and therefore not repeated here. Download of the problem files is e. g. possible from `http://neo.lcc.uma.es/radi-aeb/WebVRP/`.

We simulated a decision maker changing the relative importance w_{DIST} during the optimization procedure. First, a decision maker starting with a $w_{DIST} = 1$ and successively decreasing it to 0, second a decision maker starting with a $w_{DIST} = 0$ and increasing it to 1, and third a decision maker starting with a $w_{DIST} = 0.5$, increasing it to 1 and decreasing it again to 0. Between adjusting the values of w_{DIST} in steps of 0.1, enough time for computations has been given to the system to allow a convergence to (at least) a local optimum. The system then has to follow the updated preference information, resequencing and reassigning the customers using the implemented local search metaheuristics.

The linear, additive model is one of the possibilities to describe a utility function, established in the literature. It is well-known that not any utility function follows the described approach, but we nevertheless introduce it as it has several advantages, one being the decision makers familiarity with graphical user interfaces where slider bars are used to modify the weight settings. Given the interaction of the decision maker with the system by means of a slider bar, it may equally be possible that the decision maker changes the weight settings in rather large steps instead of the small linear steps of size 0.1. In this case, we expect that the experimental setup does not lead to different results. Given appropriate computer hardware, the system should simply change the solutions quickly from one solution to another without necessarily showing intermediate alternatives.

Figure 2 to 6 plot the results obtained during the test runs.

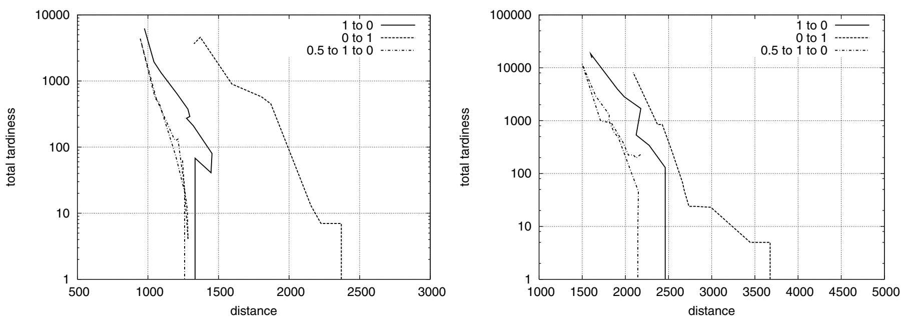

Fig. 2. Results of the test runs on instance 1a and 2a

It can be seen, that the results are significantly different depending on the initial chosen value of w_{DIST}. For initial values of $w_{DIST} = 0.5$, the framework follows more closely the Pareto front compared to other initial parameter settings.

To illustrate this behavior more closely, we are going to discuss the results for instance '1a' more closely and verbally. The first decision maker starts with $DIST = 975, TARDY = 6246$ and moves to $DIST = 1412, TARDY = 0$ while the second starts with $DIST = 2953, TARDY = 0$ and moves to $DIST = 1326$, $TARDY = 3654$. Clearly, the first strategy outperforms the second. While an initial value of $w_{DIST} = 0$ allows the identification of a solution with zero tardiness, it tends to construct routes that, when decreasing the relative importance

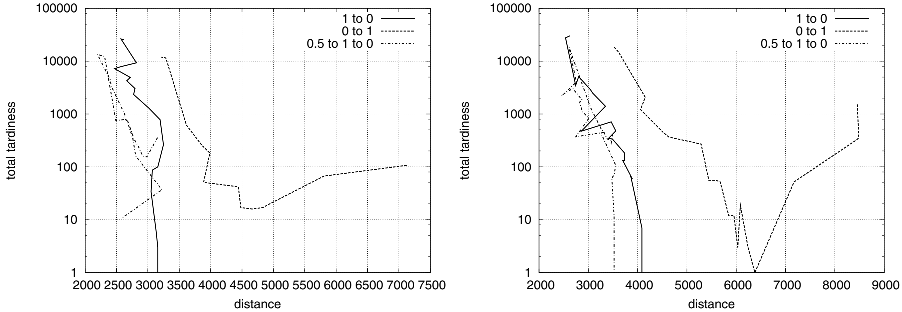

Fig. 3. Results of the test runs on instance 3a and 4a

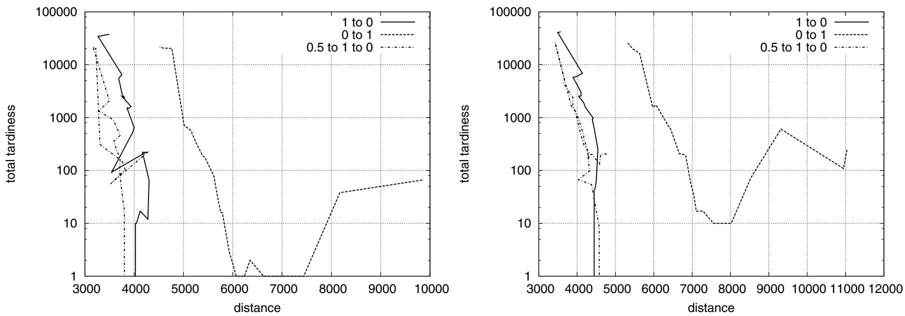

Fig. 4. Results of the test runs on instance 5a and 6a

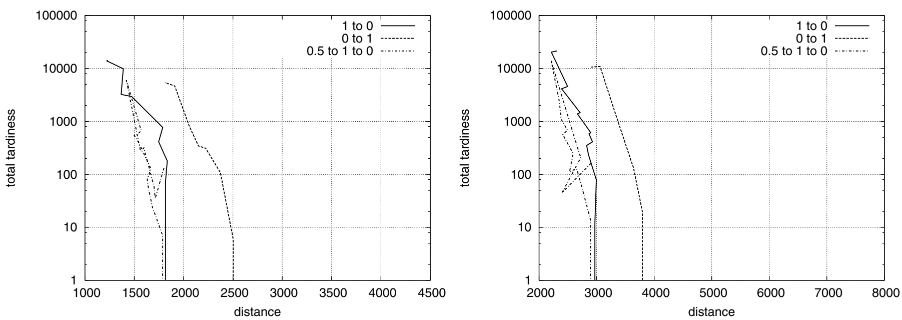

Fig. 5. Results of the test runs on instance 7a and 8a

of the tardiness, turn out to be hard to adapt. In comparison to the strategy starting with a $w_{DIST} = 1$, the clustering of customers turns out the be prohibitive for a later improvement.

When comparing the third strategy of starting with a $w_{DIST} = 0.5$, it becomes obvious that this outperforms both other ways of interacting with the system. Here, the solutions start with $DIST = 1245$, $TARDY = 63$, go to $DIST = 946$,

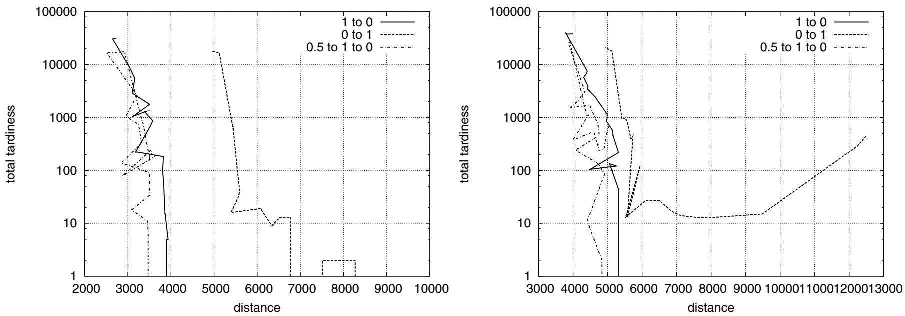

Fig. 6. Results of the test runs on instance 9a and 10a

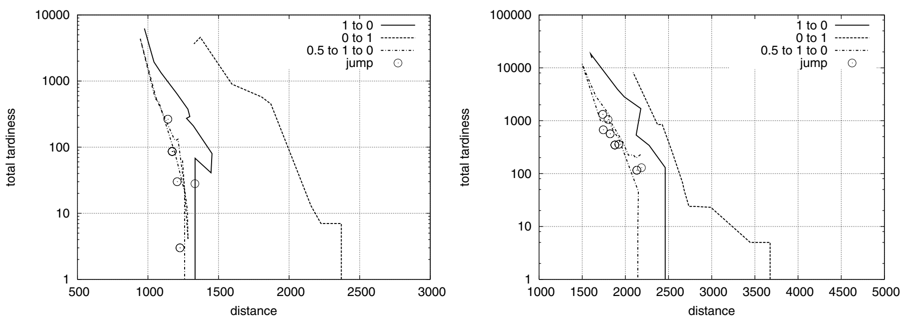

Fig. 7. Results of the test runs on instance 1a and 2a, compared to discrete jumps around $w_{DIST} = 0.7$

$TARDY = 4342$, and finally to $DIST = 1335$, $TARDY = 0$. Apparently, starting with a compromise solution is beneficial even for both extreme values of $DIST$ and $TARDY$.

To further investigate cases where the decision maker changes the weight setting in bigger discrete steps, we simulated a decision maker finally approaching a weight value of $w_{DIST} = 0.7$, starting from values around 0.7 with decreasing distance. The precise sequence of weights in the experiment is $w_{DIST} = \{0.4, 0.9, 0.6, 0.8, 0.65, 0.75, 0.7\}$, and the distances to the final, desired weight setting of the decision maker are $-0.3, +0.2, -0.1, +0.1, -0.05, +0.05, 0$. The setting of the experiment is based on the assumption that starting from an initial weight setting, the decision maker approaches the actually desired one by an alternating process of over- and underestimating the true value of w_{DIST}. The distances to the desired w_{DIST} decrease in this process as the decision maker reflects upon the chosen weight combination and the solution presented by the system.

Figure 7 plots the obtained results for the instances 1a and 2a. We chose to omit the plots for the other instances as the following interpretation and the resulting conclusions are identical.

It can be seen, that the discrete changes of the weight values lead to results that closely follow the curve obtained for the third decision maker. As suspected above, the results are comparable to the strategy of changing w_{DIST} in steps of 0.1.

4 Summary and Conclusions

A framework for the interactive resolution of multi-objective vehicle routing problems has been presented. The concept has been implemented in a computer system. Results on a benchmark instance have been reported, compared, and analyzed.

First investigations indicate that the concept may successfully solve vehicle routing problems under multiple objectives and complex side constraints. In this context, an interaction with the system is provided by a graphical user interface. The relative importance of the objective functions can be modified by means of a slider bar, resulting in different solutions which are computed in real time by the system, therefore providing an immediate feedback to the user. Figure 8 shows two extreme solutions that have been interactively obtained by the system.

As a result of the experiments, it becomes clear that for the investigated case, a compromise value of $w_{DIST} = 0.5$ should be chosen for the computation of a first solution before starting an interaction with the system. The so constructed alternative can be modified towards the minimization of the traveled distances as well as towards the minimization of the total tardiness.

Besides this theoretically gained insight, the contribution of the framework can also be seen in describing a general concept for the resolution of complex vehicle routing problems. As practical problems often vary in terms of their characteristics, this may turn out to be beneficial when problems with different side constraints have to be addressed using a single optimization procedure. An additional use can be found for dynamic vehicle routing problems. The market mechanism provides a platform for the matching of offers to vehicles without the immediate need of accepting them, yet still obtaining feasible solutions and gathering a prize for acceptance of offers which may be reported back to the customer.

Future developments are manifold. First, other ways of representing preferences than a weighted sum approach may be beneficial to investigate. While the comparable easy interaction with the GUI by means of a slider bar enables the user to directly change the relative importance of the objective functions, it prohibits the definition of more complex preference information, e. g. involving aspiration levels.

Second, different and improved ways of implementing the market mechanism have to be investigated. First results indicate that the quality of the solutions is biased with respect to the initial setting of the relative importance of the optimality criteria. It appears as if more complex reallocations of orders between vehicles are needed to address this issue.

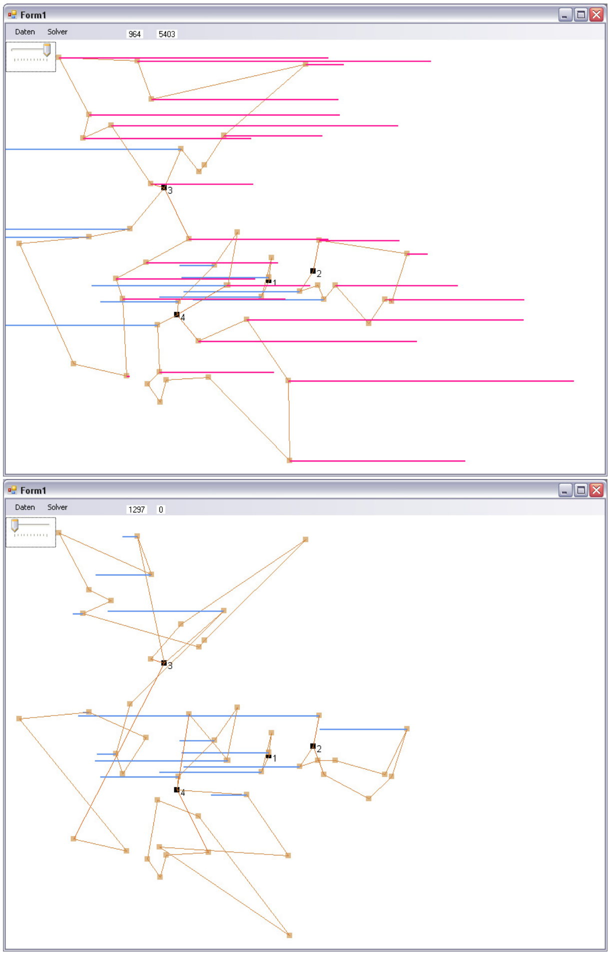

Fig. 8. Two screenshots of the graphical user interface. On the top, a short solution with high tardiness, on the bottom, a solution with low tardiness but long traveling distances.

References

1. Solomon, M.M., Desrosiers, J.: Time window constrained routing and scheduling problems. Transportation Science **22**(1) (February 1988) 1–13
2. Éric Taillard, Badeau, P., Gendreau, M., Guertin, F., Potvin, J.Y.: A tabu search heuristic for the vehicle routing problem with soft time windows. Transportation Science **31**(2) (May 1997) 170–186
3. Geiger, M.J.: Genetic algorithms for multiple objective vehicle routing. [22] 349–353
4. Potvin, J.Y., Kervahut, T., Garcia, B.L., Rousseau, J.M.: The vehicle routing problem with time windows part i: Tabu search. INFORMS Journal on Computing **8**(2) (Spring 1996) 158–164

5. Potvin, J.Y., Bengio, S.: The vehicle routing problem with time windows part ii: Genetic search. INFORMS Journal on Computing **8**(2) (Spring 1996) 165–172
6. Gendreau, M., Bräysy, O.: Metaheuristic approaches for the vehicle routing problem with time windows: A survey. In Ibaraki, T., Yagiura, M., Nonobe, K., eds.: Proceedings of the Fifth Metaheuristics International Conference - MIC2003, Kyoto, Japan (August 2003) 1–10
7. Beasley, J.E.: OR-library: Distributing test problems by electronic mail. Journal of the Operational Research Society **41**(11) (1990) 1069–1072
8. Fink, A., Voß, S.: HotFrame: A heuristic optimization framework. In Voß, S., Woodruff, D.L., eds.: Optimization Software Class Libraries. Kluwer Academic Publishers, Boston, Dordrecht, London (2002) 81–154
9. Di Gaspero, L., Schaerf, A.: Easylocal++: an object-oriented framework for the flexible design of local-search algorithms. Software: Practice & Experience **33**(8) (July 2003) 733–765
10. Cahon, S., Melab, N., Talbi, E.G.: ParadisEO: A framework for the reusable design of parallel and distributed metaheuristics
11. Rahoual, M., Kitoun, B., Mabed, M.H., Bachelet, V., Benameur, F.: Multicriteria genetic algorithms for the vehicle routing problem with time windows. [22] 527–532
12. Jozefowiez, N., Semet, F., Talbi, E.G.: Parallel and hybrid models for multi-objective optimization: Application to the vehicle routing problem. In et al., J.M.G., ed.: Parallel Problem Solving from Nature VII. Volume 2439 of Lecture Notes in Computer Science., Berlin, Heidelberg, New York, Springer Verlag (2002) 271–280
13. Murata, T., Itai, R.: Multi-objective vehicle routing problems using two-fold EMO algorithms to enhance solution similarity on non-dominated solutions. In Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E., eds.: Evolutionary Multi-Criterion Optimization, Third International Conference, EMO 2005. Volume 3410 of Lecture Notes in Computer Science., Berlin, Heidelberg, New York, Springer Verlag (2005) 885–896
14. Jozefowiez, N., Semet, F., Talbi, E.G.: Enhancements of NSGA II and its application to the vehicle routing problem with route balancing. In Talbi, E.G., Liardet, P., Collet, P., Lutton, E., Schoenauer, M., eds.: Artificial Evolution. Volume 3871 of Lecture Notes in Computer Science., Berlin, Heidelberg, New York, Springer Verlag (2006) 131–142
15. Stacey, P.J.: Practical vehicle routeing using computer programs. Journal of the Operational Research Society **34**(10) (1983) 975–981
16. Waters, C.D.J.: Interactice vehicle routeing. Journal of the Operational Research Society **35**(9) (1984) 821–826
17. Park, Y.B., Koelling, C.P.: A solution of vehicle routing problems in a multiple objective environment. Engineering Costs and Production Economics (10) (1986) 121–132
18. Park, Y.B., Koelling, C.P.: An interactive computerized algorithm for multicriteria vehicle routing problems. Computers & Industrial Engineering **16**(4) (1989) 477–490
19. Phelps, S.P., Köksalan, M.: An interactive evolutionary metaheuristic for multi-objective combinatorial optimization. Management Science **49** (2003) 1726–1738

20. Cahon, S., Simarik, T., Vironda, T.: A graphical user interface for multi objective optimization guimoo http://guimoo.gforge.inria.fr/
21. Cordeau, J.F., Laporte, G., Mercier, A.: A unified tabu search heuristic for vehicle routing problems with time windows. Journal of the Operational Research Society (52) (2001) 928–936
22. de Sousa, J.P., ed.: Proceedings of the Metaheuristics International Conference MIC 2001. In de Sousa, J.P., ed.: Proceedings of the Metaheuristics International Conference MIC 2001, Porto, Portugal (July 2001)