Proceedings of the 9th Biennial ASME Conference on Engineering Systems Design and Analysis
ESDA08
July 7-9, 2008, Haifa, Israel

ESDA2008-59077

A SURVEY OF AUTONOMOUS CONTROL ALGORITHMS BY MEANS OF ADAPTED VEHICLE ROUTING PROBLEMS

Bernd Scholz-Reiter
BIBA - Bremer Institut für Produktion und Logistik GmbH

Henning Rekersbrink
BIBA - Bremer Institut für Produktion und Logistik GmbH

Bernd-Ludwig Wenning
ComNets - Working Group Communication Networks

Thomas Makuschewitz
BIBA - Bremer Institut für Produktion und Logistik GmbH

University of Bremen
Bibliothekstraße 1, 28359 Bremen, Germany

ABSTRACT
The German Collaborative Research Centre 637 "Autonomous Cooperating Logistic Processes – A Paradigm Shift and its Limitations", develops, among other things, autonomous routing algorithms for transport networks. The discussed algorithm is designed to match goods and vehicles and to continuously make route decisions within a dynamic transport network. Here, each object makes its own decisions. It is called Distributed Logistics Routing Protocol – DLRP.
Because of obvious similarities to the Vehicle Routing Problem (VRP), one question arises for both practitioners and researchers: How efficient is this protocol compared to traditional, established algorithms or in comparison to the optimal solution? This article tries to answer this question, which appears simple on the first and challenging on the second view.

1. INTRODUCTION
To describe the different approach of autonomous control to transport problems, the developed Distributed Logistics Routing Protocol DLRP is described in the first section. In contrast to traditional algorithms for the VRP problem, which do static optimisation, this approach tries to control an ongoing dynamic transport process.
Therefore the problem definitions of both sides are basically different. The VRP is a static problem, because all customers are known at the beginning. The problem for the DLRP is in contrast a dynamic network formulation: the customers appear continuously and are not known from the beginning. This article describes an adapted VRP scenario, which is used to compare different algorithms.
In order to rate the results in a more objective way, not only results from both sides are presented. To have lower and upper bounds for the overall vehicle distance, we additionally calculated the optimal vehicle ways as best and some kind of random vehicle ways as worst case values.

2. DISTRIBUTED LOGISTICS ROUTING PROTOCOL
The approach to the transportation problem taken for the DLRP is different to the approach for the traditional VRP. The developed protocol is not an optimisation algorithm, but an autonomous control algorithm, which means that it is designed for a continuous changing process. The native scenario for the DLRP is a network, where transport orders arise continuously and a changing fleet of vehicles has to deal with the whole workload of the net.

The routing protocol was inspired by internet routing protocols, which are able to find routes through a net, which is not known exactly and which is changing permanently. The basic concept for one data package which wants to get to its destination is the RouteRequest/RouteReply mechanism. This package sends a RouteRequest to all its neighbour vertices, which for themselves sent it ahead to their neighbours. If one vertex notices, that it is the destination for this RouteRequest, it sends a RouteReply back to the asking package (for a more detailed description, see e.g. [1]).

One part of the DLRP is based on this concept. On the basis of figure 1 the fundamental procedure of the developed protocol can be illustrated: When a package makes a route decision, it first disannounces its possibly announced old route (see figure 1: RouteDisAnnouncement) and announces its actual wished routes to the vertices involved (see figure 1: RouteAnnouncement). An individual vertex has thereby information about when how many packages with which goals will be at its position. Additional information such as restrictions concerning the transport of the goods (e.g. cooling freight) is stored likewise.

If a vehicle needs a route, it sends a RouteRequest to the net – the RouteRequest/RouteReply mechanisms are the same as described above. After receiving several RouteReplies, which are route suggestions with appropriate additional information now, a vehicle decides for a route. This Route is then announced to the involved vertices (see figure 1: RouteAnnouncement). This leads to a continuous cooperative structure. The objects in a transport net do not plan their route at the same time. Packages emerge continuously or achieve their goals, vehicles replan their routes and so on. At each time there is enough information for any route decision.

The algorithm is very complex and has many points, which have a direct effect on the performance (e.g. which packages are loaded into a vehicle at the vertex: all with the same direction, all ones with their destination vertex on the vehicle route or …). Hence, a detailed description is not possible here. For a more detailed description refer to [2], [3] and [4]. The DLRP is enhanced continuously for both dynamic and static scenarios.

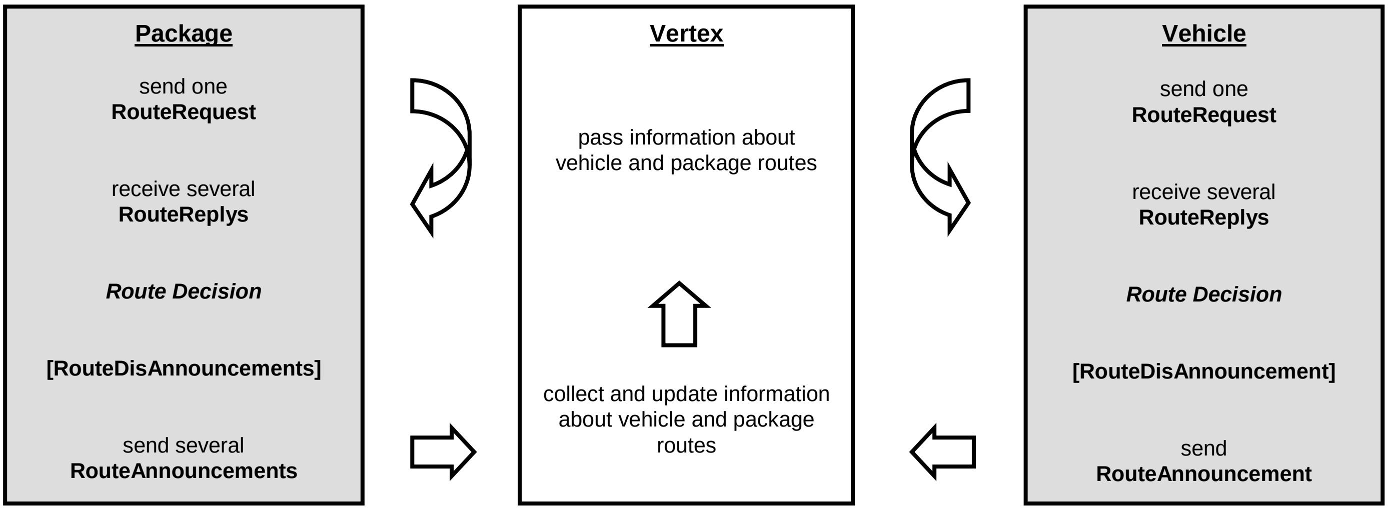

Figure 1. Scheme of the Distributed Logistics Routing Protocol – DLRP (from [3]).

3. ADAPTED VEHICLE ROUTING PROBLEM

The major differences between the traditional vehicle routing problem formulation and the scenarios for the DLRP are:

- the VRP implies a full net, which means that every vertex is connected directly with every other vertex – the DLRP scenario has few edges
- the DLRP is made for a dynamic scenario, orders may appear at every place every time – the VRP orders are known from the beginning
- the VRP enforces closed routes – for the DLRP as a dynamic control algorithm it is very difficult to diagnose route terminating conditions
- the objective function for the VRP is the sum of all vehicle distances – within the DLRP, all objects (all vehicles and all packages) have their own objective function: shortest way for packages and the best utilisation for vehicles

It is not possible for a DLRP implementation to deal with full nets. One can easily imagine what happens if one DLRP-RouteRequest is sent to every neighbour in a full net and again to the next neighbours and so on. In this point, we decided to restrict the net to feasible edges. On the other hand, it is easy for some traditional VRP-algorithms to build them for not-full-nets.

We decided to take a real world network. In figure 2 you can see the chosen topology, which is the basic autobahn-net of Germany. The topology contains 18 vertices, the biggest cities in Germany, and 35 undirected edges. The scenario edge lengths match the real ones. They are shown in table 2.

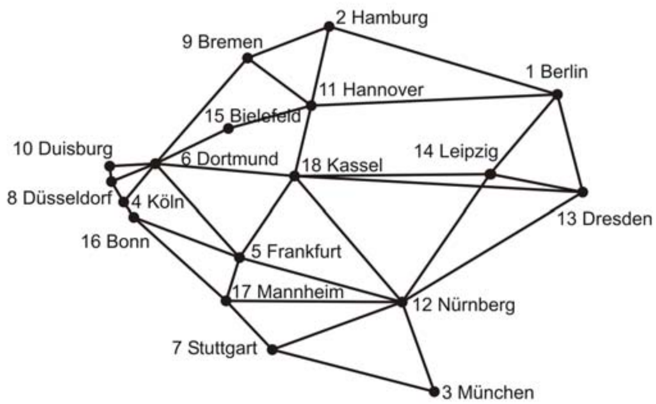

Figure 2. Topology for the adapted VRP (from [5]).

Because of the static nature of all traditional VRP optimisation algorithms, the adapted VRP was also created as a static problem. All orders are known from the beginning. Even though the DLRP was created especially for the control of dynamic environments, it can cope with this static one. For a good performance, the DLRP had to be adopted for this special static case.

Within a simulation of the DLRP, it is almost impossible to find the point when a vehicle has finished its work and could go home. It can always happen, that it loads up another package at the next vertex. Time restrictions for maximum vehicle driving times can be implemented, but forced closed routes for this static scenario would unnecessarily extend the vehicle distances. For the traditional VRP-algorithms, it is not too difficult, to transfer the objective function to non-closed vehicle routes. So the adapted VRP treats the distance of non-closed vehicle routes as objective to minimise.

The fourth point is not a conflict, but it has to be mentioned. One can show that the overall vehicle distance ($\sum d_j$), package distances (p_i) and vehicle utilisation (u_j) are connected:

$$\frac{\sum\limits_{all\ packages\ i} p_i}{\sum\limits_{all\ vehicles\ j} u_j \cdot \frac{d_j}{\sum d_j}} = \frac{vehicle\ capacity}{package\ size} \cdot \sum_{all\ vehicles\ j} d_j \qquad (1)$$

This equation requires uniform vehicles and uniform packages concerning capacity and size. So the DLRP will minimise the requested objective function even though it primarily minimises the package distances and maximises the vehicle utilisation.

The used scenarios were built as distribution scenarios, which means that all vehicles are at Kassel at the beginning and all orders start form Kassel. The different scenarios are shown in table 1. The number of vehicles can be 3, 6 or 9, while the number of packages can vary between 17, 34 or 51. The amount of packages was matched to the topology. All 17 vertices should be costumers. The package destinations for the bigger scenario sets are uniformly distributed, whereas each vertex has one package at least. Therefore the 34 and 51-scenarios have 10 subsets with different package destinations, which are shown in table 3. The vehicle capacity was chosen in that way, that every vehicle is needed, if no vehicle comes back to Kassel.

Table 1. Chosen scenarios and corresponding vehicle capacities.

<table>
<tr><th colspan="2" rowspan="2"></th><th colspan="3">number of vehicles</th></tr>
<tr><th>3</th><th>6</th><th>9</th></tr>
<tr><td rowspan="6">number of packages</td><td rowspan="2">17</td><td>8 pck/veh</td><td>3 pck/veh</td><td>2 pck/veh</td></tr>
<tr><td colspan="3">1 subset</td></tr>
<tr><td rowspan="2">34</td><td>16 pck/veh</td><td>6 pck/veh</td><td>4 pck/veh</td></tr>
<tr><td colspan="3">10 subsets</td></tr>
<tr><td rowspan="2">51</td><td>25 pck/veh</td><td>10 pck/veh</td><td>6 pck/veh</td></tr>
<tr><td colspan="3">10 subsets</td></tr>
</table>

For the scenario, three indicator values can be given:

- the shortest way from Kassel via all 17 vertices is 2235 km long
- the shortest way without any loop is 2245 km long
- the sum of shortest ways for each for the 17 packages is 4965 km

Table 2. Distance matrix for the scenario topology (values in km).

		Berlin 1	Hamburg 2	München 3	Köln 4	Frankfurt 5	Dortmund 6	Stuttgart 7	Düsseldorf 8	Bremen 9	Duisburg 10	Hannover 11	Nürnberg 12	Dresden 13	Leipzig 14	Bielefeld 15	Bonn 16	Mannheim 17	Kassel 18
Berlin	**1**		280									260		175	160				
Hamburg	**2**	280								105		150							
München	**3**							230					175						
Köln	**4**						100		50								25		
Frankfurt	**5**						235						200				180	100	190
Dortmund	**6**				100	235			60	215	75					80			160
Stuttgart	**7**			230									210					135	
Düsseldorf	**8**				50		60				30								
Bremen	**9**		105				215					105							
Duisburg	**10**						75		30										
Hannover	**11**	260	150							105						115			165
Nürnberg	**12**			175		200		210						305	255			245	310
Dresden	**13**	175											305		120				390
Leipzig	**14**	160											255	120					300
Bielefeld	**15**						80					115							
Bonn	**16**				25	180												220	
Mannheim	**17**					100		135					245				220		
Kassel	**18**					190	160					165	310	390	300				

Table 3. Package destinations for the different scenario subsets.

		Berlin 1	Hamburg 2	München 3	Köln 4	Frankfurt 5	Dortmund 6	Stuttgart 7	Düsseldorf 8	Bremen 9	Duisburg 10	Hannover 11	Nürnberg 12	Dresden 13	Leipzig 14	Bielefeld 15	Bonn 16	Mannheim 17	Kassel 18
34 packages	No 1	3	2	2	1	1	1	2	3	1	2	4	2	2	2	2	2	2	0
	No 2	2	1	4	2	3	2	2	1	2	1	2	2	1	1	3	3	2	0
	No 3	3	2	1	2	2	2	2	3	2	1	2	1	2	2	3	1	3	0
	No 4	2	2	2	2	1	2	1	4	4	2	1	1	2	1	1	4	2	0
	No 5	2	1	2	1	1	2	4	2	2	3	3	1	1	1	3	2	3	0
	No 6	2	4	1	1	1	1	4	1	1	3	1	1	1	7	1	3	1	0
	No 7	4	1	1	2	2	4	3	1	3	2	2	1	2	1	2	1	2	0
	No 8	1	1	4	2	3	1	2	2	2	2	1	4	2	2	3	1	1	0
	No 9	2	1	4	1	1	2	1	1	2	3	1	3	4	1	4	2	1	0
	No 10	2	3	1	1	2	1	1	2	2	2	2	2	3	2	4	2	2	0
51 packages	No 1	3	4	4	2	2	3	2	7	1	3	5	2	3	3	3	2	2	0
	No 2	3	1	4	3	5	4	2	2	2	1	5	2	1	3	3	7	3	0
	No 3	3	4	3	3	2	3	2	4	3	4	2	2	4	2	5	1	4	0
	No 4	3	4	3	4	2	4	2	6	4	2	3	1	4	2	1	4	2	0
	No 5	2	2	3	1	1	3	7	4	2	4	5	3	2	2	5	2	3	0
	No 6	3	8	1	1	1	2	5	3	1	4	1	2	2	8	1	5	3	0
	No 7	5	1	1	4	4	4	4	2	3	4	4	4	3	1	2	2	3	0
	No 8	3	2	6	2	3	2	3	2	3	5	2	4	4	2	4	1	3	0
	No 9	3	3	4	2	1	2	1	3	3	6	4	3	5	1	6	3	1	0
	No 10	5	4	4	2	2	2	2	2	4	3	3	2	4	2	5	3	2	0

4. RANDOM-LIKE SOLUTION

In order to give something like an upper bound for reasonable solutions to the scenarios, a random-like heuristic was implemented.

In the first step, all packages are assigned randomly to the vehicles. A uniform distribution is used for this step, with the restriction that every vehicle has to carry at least one package. The second step is to find an optimal way for one vehicle and its load of packages. To save computing time, only loop-free routes are considered. This restriction makes the route non optimal in some cases, but it is assumed that the optimal solution is not too far (see above, the difference between the shortest way with and without loops is 10 km or 0.4 %).

This algorithm was calculated 10,000 times for each subset. The resulting mean values are shown in the next table.

Table 4. Vehicle distances calculated by the random-like algorithm.

		number of vehicles		
		3	6	9
number of packages	17	3,927	5,118	5,474
	34	5,051	7,594	9,072
	51	5,583	8,940	11,225

The described algorithm has some analogies with the real world transport market: the different forwarder companies receive their orders randomly and each company tries to optimise its vehicle routes on its own. An overall optimum cannot be expected from a procedure like this. Note that the average utilization which was reached in the biggest scenario 51-9, about 60 %, would be a very good value for real world forwarder companies. The vision of the DLRP is to implement this protocol independently from different companies. In this vision, an overall optimisation can happen without taking any decision possibilities from the single forwarders [3].

5. OPTIMAL SOLUTION

On the other side, a lower boundary for the overall vehicle distance should be given. To calculate optimal solutions for the given instances, the specified problem was formulated as a MIP (Mixed-Integer Problem). Because of the quite small size for the instances, the MIP could be solved by CPLEX within feasible time. In the following table the resulting optimal values for the overall vehicle distances can be seen. Note that in this solution, routes with loops are allowed and each vehicle must carry at least one package.

Scholz-Reiter, B.; Rekersbrink, H.; Wenning, B.-L.; Makuschewitz, T.: A SURVEY OF AUTONOMOUS CONTROL ALGORITHMS BY MEANS OF ADAPTED VEHICLE ROUTING PROBLEMS. In: Proceedings of the 9th Biennial ASME Conference on Engineering Systems Design and Analysis ESDA 08. ASME, Haifa, Israel, 2008

Table 5. Optimal vehicle distances.

		number of vehicles		
		3	**6**	**9**
number of packages	**17**	2,245	2,565	3,095
	34	2,245	2,868	3,310
	51	2,245	2,608	3,362

6. TABU SEARCH SOLUTION

As a representative of established solution techniques [6] for vehicle routing problems, a tabu search algorithm was applied to the scenarios.

This algorithm is similar to the random-like solution technique. One solution set for the tabu search is one assignment set. This set assigns all packages to the vehicles. After the assignment, an optimal route for each vehicle is calculated (again with loop-free routes only like the random-like algorithm).

For the neighbourhood generation, the λ -interchange generation mechanism by Osmand [7] was implemented. The λ was set to 1 and it was insured that no vehicle is empty.

Note that tabu search is very close to the optimal solution in the small scenarios and moves away with bigger scenarios.

Table 6. Vehicle distances calculated by the tabu search algorithm.

		number of vehicles		
		3	**6**	**9**
number of packages	**17**	2,410	2,610	3,095
	34	3,335	3,689	3,775
	51	3,988	4,637	5,026

7. DLRP SOULTION

The DLRP solution was calculated with the DLRP roughly described above. To increase the performance, the protocol was adapted to this special static situation. Additionally the decision functions for the different objects were simplified and harmonized: packages only choose their routes by the route length, vehicles only by the estimated utilisation. This means that the packages do not adjust to the vehicle routes, in contrast to the dynamic version. Only vehicles choose their route dependent on the package route situation.

The results are shown in table 7. Routes with loops are allowed here and each vehicle does not need to carry a package, but they do so.

Table 7. Vehicle distances calculated by the DLRP.

		number of vehicles		
		3	**6**	**9**
number of packages	**17**	4,530	2,875	4,400
	34	5,184	5,928	7,022
	51	4,918	4,807	6,310

8. CONCLUSIONS

For a better overview, all results are shown as line charts in the following figures 3 to 5. One can recognize that the tabu search heuristic leads on the average to better solutions for these scenarios than the DLRP. But compared to the optimum and the random-like solution it can be detected, that tabu search is getting worse with bigger scenarios. This is a normal behavior for tabu search algorithms and it's due to the growing solution space. In contrast, it can be found that the DLRP is getting better with growing scenarios – compared to the optimum and the worst case scenario. For the big scenarios with 51 packages, the two methods can be described as nearly equal. Larger scenarios were not implemented this time because of the exponentially growing computation time.

Further on some indications speak for the adequacy of the DLRP. On the one hand, it is possible to reach a very good result for the shortest way for one vehicle mentioned above (2,780 / 2,235 km). On the other hand, bigger scenarios (204 packages, 9 vehicles, capacity 6) show a good utilisation of 60% and adequate relative package ways of 1,6 (see description below). In addition, the optimisation of the DLRP is still on work. It is a very complex algorithm (at least 15 main parameters) and there are always points to enhance.

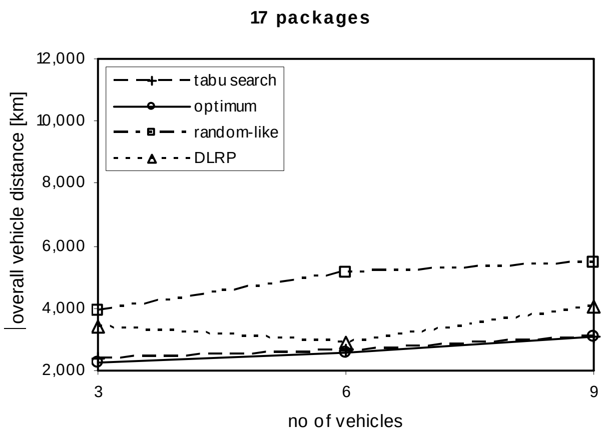

Figure 3. Results for 17 packages.

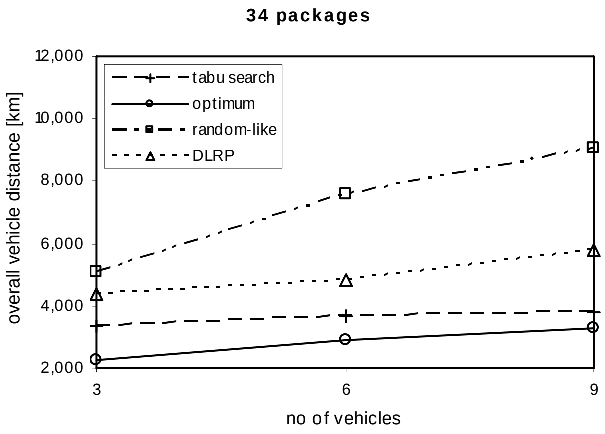

Figure 4. Results for 34 packages.

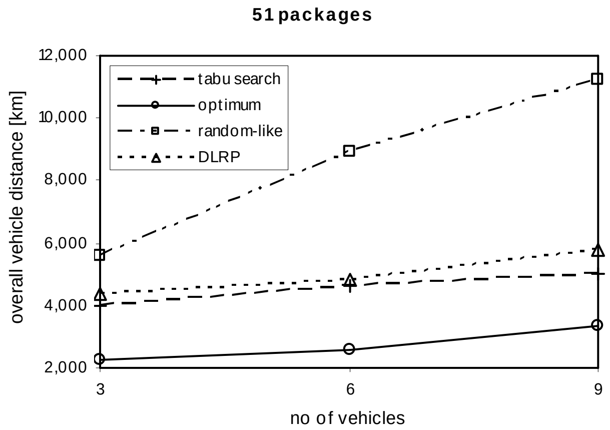

Figure 5. Results for 51 packages.

Because the DLRP is originally a control method, its main advantages point themselves in dynamic and close to reality scenarios: self-adaption to changing situations, possible manual interventions at runtime, implementation of uncertain knowledge and complex and context driven decision functions [3].

For bigger scenarios, the classification and evaluation of algorithms for the VRP gets more and more difficult. For these scenarios, it is not possible to calculate optimal solutions, so a lower bound is missing. On the other hand, it is not possible to compare two different scenarios, because the optimal way lengths can be very different. On the basis of equation (1), we suggest an alternative comparison approach. The vehicle utilisation and the package distances are, with some restrictions, directly connected to the overall vehicle distance. The vehicles utilisation has a natural upper bound: it cannot be greater than one. The package distances have a natural lower bound, which is easy to calculate: their individual shortest way to their destination. Therefore we can define a relative package distance: driven package distance by shortest possible distance. The lower bound is one here.

Now we can illustrate these two values in one plane, shown in figure 6. The lower left corner represents one extreme: each package has one vehicle of capacity 1 and drives its shortest way. The horizontal line at utilisation = 0.5 is another extreme: when one vehicle with capacity for all packages brings out all packages, the utilisation goes to 0.5 and the relative package distance increases infinitely.

In this presentation form, on can see that an optimal solution must be somewhere in the lower left corner: a relative package distance of nearly 1 and a high utilisation of 80 – 90 % (see optimum of scenario 51-9)

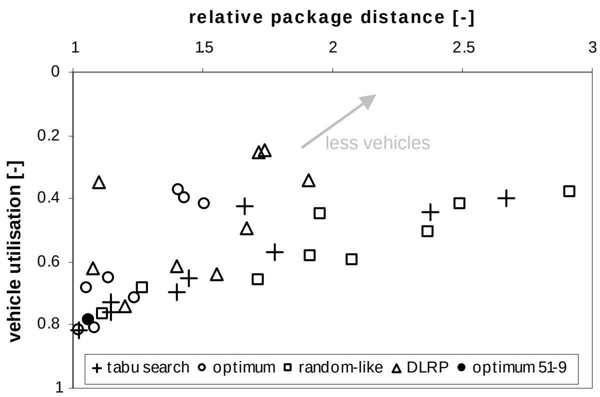

Figure 6. Alternative approach to the comparison of VRP solutions.

9. REFERENCES

[1] Perkins, C. E., 2001, "Ad hoc networking", Addison-Wesley, Boston

[2] Scholz-Reiter, B., Rekersbrink, H., Freitag, M., 2006, "Kooperierende Routingprotokolle zur Selbststeuerung von Transportprozessen", Industrie Management, 22/3, pp. 7-10

[3] Scholz-Reiter, B.; Rekersbrink, H.; Freitag, M., 2006, "Internet routing protocols as an autonomous control approach for transport networks", Proc. of the 5th CIRP international seminar on intelligent computation in manufacturing engineering, pp. 341-345

[4] Wenning, B.-L.; Rekersbrink, H.; Timm-Giel, A.; Görg, C.; Scholz-Reiter, B., 2007, "Autonomous control by means of distributed routing", Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow, Springer, Berlin, pp. 325-336

[5] Wenning, B.-L.; Rekersbrink, H.; Becker, M.; Timm-Giel, A.; Görg, C.; Scholz-Reiter, B., 2007, "Dynamic Transport Reference Scenarios", Understanding Autonomous Cooperation & Control in Logistics – The Impact on Management, Information and Communication and Material Flow, Springer, Berlin, pp. 337-350

[6] Laporte, G., 1992, "The Vehicle Routing Problem: An overview of exact and approximative algorithms", European Journal of Operational Research, 59, pp.345-358

[7] Osman, I. H., 1993, "Metastrategy simulated annealing and tabu search for the vehicle routing problem", Annals of Operations Research, 41, pp. 421-451