Wenning, B.-L., Rekersbrink, H., Görg, C.: Scalability Investigations on Communication Traffic in Distributed Routing of Autonomous Logistic Objects. Proceedings of the 9th International Conference on Intelligent Transport System Telecommunications (ITST 09), pp. 8-12.

Scalability Investigations on Communication Traffic in Distributed Routing of Autonomous Logistic Objects

Bernd-Ludwig Wenning
Communication Networks
University of Bremen
Otto-Hahn-Allee, 28359 Bremen, Germany
Email: wenn@comnets.uni-bremen.de

Henning Rekersbrink
Bremer Institut für
Produktion und Logistik GmbH
Bremen, Germany
Email: rek@biba.uni-bremen.de

Carmelita Görg
Communication Networks
University of Bremen
Bremen, Germany
Email: cg@comnets.uni-bremen.de

***Abstract*—In current transport logistics, routing is usually done centrally. A dedicated routing instance solves the optimisation problem of finding the best solution to handle the current set of orders with the set of available vehicles under constraints such as vehicle utilisation, punctuality etc. Because of the increasing complexity of logistic processes, approaches have been suggested recently which change this centralised routing paradigm towards a distributed approach with autonomous logistic entities (vehicles and goods) deciding on their own. To be able to obtain enough knowledge for reasonable decisions, the logistic entities have to communicate with each other. For this interaction, the information exchange concept DLRP (Distributed Logistic Routing Protocol) has been proposed before. The work presented in this paper will focus on the aspect of scalability of communication in a DLRP scenario. Message flooding is identified as potential challenge for the scalability of DLRP, and intelligent flooding restrictions to the communication traffic are applied.**

I. Introduction

In current transport logistics practice, routing is usually handled as a constrained optimisation problem. On a small scale, it may be optimised by a human dispatcher using his experience. For larger scales, computing systems using heuristic methods such as genetic algorithms or tabu search are applied. The optimisation problem formulations are usually Pickup and Delivery Problems (PDP) or Vehicle Routing Problems (VRP), in most cases constrained by pickup or delivery time windows.

When dynamics are introduced into the logistic scenarios, e.g. transport orders that are not known in advance, the optimal solution has to be redetermined as time progresses. These reoptimisations may be done either in regular intervals or on demand and have to consider further constraints as vehicles may already be on the road with goods on them, thus being more limited in their flexibility. As these reoptimisations and the additional constraints therein are a complex challenge for a centralised routing system, the idea of changing paradigms towards a distributed routing in transport logistics is currently under research, for example in the framework of the Collaborative Research Centre 637 at the University of Bremen [1].

As an information exchange framework for distributed routing of autonomous logistic entities, the *Distributed Logistic Routing Protocol* (DLRP) [2], [3] has been proposed as the transfer of ideas from routing in wireless communication networks to routing in transport logistics. In this DLRP, autonomous vehicles and goods make individual route decisions. To make reasonable decisions, these entities need to have knowledge about other entities' decisions, i.e. goods need to know where and when they can be picked up by vehicles and vice versa. Therefore, each of the vehicles and goods has to communicate with other entities in the network to obtain its knowledge and to announce its decisions. This communication traffic is investigated in the work presented here, and it is shown how the evaluation function that is being used for route decision can also be used to limit the communication traffic.

II. The DLRP

The Distributed Logistic Routing Protocol (DLRP) ([2], [3]) is based on the assumption that the vehicles and the goods in a logistic network are equipped with devices capable of computing and communicating. Thereby, they are able to interact and decide autonomously. Its basic concepts are adapted from routing algorithms that are used in wireless ad-hoc communication networks, where routes have to be found in dynamically changing topologies.

In contrast to the classical routing scenarios where heuristic methods are applied to optimisation problems such as the Vehicle Routing Problem (VRP) or the Pickup and Delivery Problem (PDP), the scenarios where the DLRP is applied are restricted on existing connections between locations (vertices) in the logistic network. Scenario topologies are not only defined by a set of vertices, but by a graph connecting those. Figure 1 illustrates this. In reality, the vertices may be logistic distribution centers and the edges the main motorway connections between them.

Vehicles and goods that use the DLRP determine their routes by using a route discovery messaging that is similar to source routing methods in ad-hoc communication networks: They send out a route request to the nearest vertex (the "associated vertex"), which forwards it to the neighbor vertices, who in turn do the same. Each vertex adds local knowledge about the

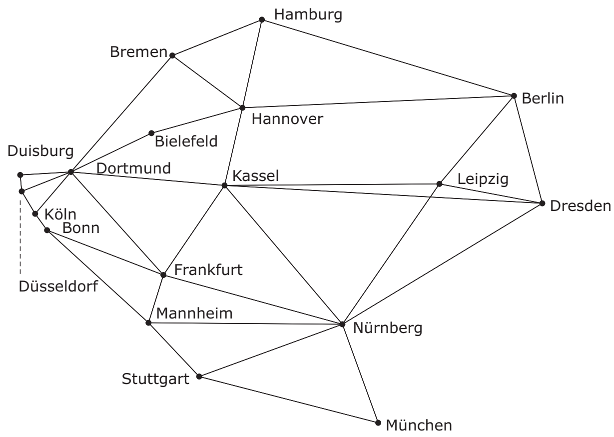

Fig. 1. Germany scenario topology with 18 vertices

current network status and transport demand to the request, so that by the time it reaches the destination vertex, the request has collected information about the complete route that it has travelled. The destination vertex sends a reply to the vehicle or good, which then can make a decision. After having made a decision, the vehicles and goods announce their intended routes to the involved vertices, where they can be used to create the relevant information for route discoveries from other vehicles and goods. Therefore, the vertices can be considered to act as information brokers.

Four main message types are present in DLRP:

- Route Requests, being sent to discover routes
- Route Replies, reply messages returned from the destinations
- Route Announcements, being sent to publish route decisions
- Route Disannouncements, being sent to cancel Route Announcements when a decision is changed

Out of these four message types, the Route Request is the one that potentially has the biggest influence on scalability as several Route Requests are forwarded among the vertices in the network in each route discovery.

Compared to ad-hoc routing protocols in communication networks such as AODV (Ad-hoc On-demand Distance Vector) [4] or DSR (Dynamic Source Routing) [5], multi-criteria routing has a significant difference: The quality of a route is not necessarily correlated with the sequence of arrival of route request messages at a specific location in the network. The sequence of route request arrivals just allows a statement about the communication path on which the route request travelled, but not on the (logistic) route associated with it. Therefore, while in AODV, the first incoming route request is assumed to represent the best route, so that subsequent incoming route requests from the same route discovery can be dropped, this assumption does not hold for routing that is a) based on multiple criteria and b) the routed goods and vehicles travel in a network whose characteristics are different from those of the communication network that transports the routing messages. Consequently, multiple route requests may need to be processed and forwarded by a vertex during a single route discovery. This leads to a potentially high amount of routing messages and causes the message flooding issue that is discussed in section IV.

III. Multi-Criteria Context-based Decision

As the route decision function in DLRP, a *Multi-Criteria Context-based Decision Function* (MCCD) was proposed recently [6]. It combines k routing criteria into one decision utility. This utility has to fulfill the following requirements:

- combination of all criteria into one measure that evaluates the route
- criteria must be weighted according to their importance
- different characteristics of each criterion should be respected, i.e. for some criteria, a non-linear mapping of values to their quality has to be possible
- a single criterion must be able to make a route impossible, e.g. if a route for a piece of goods significantly exceeds budget constraints, it is irrelevant whether all other criteria have "perfect" values.

These constraints led to a multiplicative utility U_j for route option j of the form

$$U_j = \prod_{i=1}^{k} (f_{s,i}(c_{i,j}))^{w_i}. \tag{1}$$

In this utility, each criterion $c_{i,j}$ is scaled to a common value range $[0 \quad 1]$ with the help of the scaling function $f_{s,i}$. This is done because different criteria usually have different value ranges. Without any normalisation or scaling, some criteria may dominate over others because of their value range. To be able to adjust the criterion importance for the decision, the scaled values are weighted with w_i. Then, all scaled and weighted criteria are combined by multiplication to form a utility U_j that represents the route's quality. A multiplication was chosen because it enables eliminating a route option if one of the criteria is not acceptable, thus facilitating the fourth of the aforementioned requirements. For each criterion that should be able to make the complete route option unacceptable, the scaling function has to map the unacceptable values to 0, and the multiplication then causes the complete utility to become 0, thereby rejecting the route option.

A. Applied context criteria

As described in [6], the route decisions of vehicles and goods are based on three criteria each. For vehicle routing, these criteria are:

- The expected revenue for the vehicle, which is based on the goods' offers and the transport costs. The revenue values can be positive or negative (the latter is the case if the transport costs are higher than the price the goods offer). Negative revenues, however, are mapped to 0 by the scaling function, as it is not useful for the vehicle to travel on this route.
- The ecological impact. Efficient utilisation of a vehicle's cargo space reduces the pollution per tkm. Only the

carbon dioxide output is considered here, as this can be easily calculated if the vehicles' fuel consumption is known. Low carbon dioxide output is preferred, while high output should be avoided.
- The reliability. Based on historic data collected during previous transports on a route, it can be estimated whether the expected revenue can really be achieved.

For goods routing, the three criteria are:

- The route costs. These costs depend on the offers the goods make towards the vehicles, storage costs, transshipment costs and delay fines. The goods' offers are supposed to depend on the available budget and on the urgency.
- The risk of damage. Each transshipment operation implies a risk that the goods may be damaged. Additionally, there is a damage risk related to the transport itself.
- The risk of being delayed. This risk can be deduced from knowledge about how long it takes in average to travel on a specific route. This knowledge is based on feedback from previous transports. Based on the historic travel time statistics and the time that is still left for an in-time delivery, a probability of being delayed is calculated.

IV. The message flooding issue

Whenever a DLRP-enabled vehicle or cargo is determining a route, it sends the route request message to its "associated vertex", which is the vertex which it currently approaches or where it currently is. The vertex adds its local knowledge to the request end forwards it to the neighbor vertices. These do the same, until the destination is reached. This means the route request is flooded through the network. As there are usually much more goods than vehicles in a logistic scenario, and therefore the main part of the route discovery traffic is generated by the goods, the following considerations are limited to the goods' route discoveries. Goods' routes can be considered loop-free, while vehicle routes can contain loops as long as there is some transport demand being fulfilled on the loops.

If this flooding is unrestricted, it can easily amount to a very high traffic volume in the network, especially if the network has many vertices and/or each vertex has many outgoing connections to its neighbors. An estimate for the amount of traffic is developed in the following paragraphs of this paper.

The worst case is if there are direct connections between all vertices in the network. In this case, a network of N vertices contains is exactly one path with a length of 1. For a length of 2, there are $N-2$ paths (there is exactly one path for each possible intermediate vertex), for a length of 3, there are $(N-2)(N-3)$ paths (one for each possible pair of intermediate vertices) and so on. The general term for the number of paths with a length of i is $(N-2)(N-3)\cdots(N-i) = \frac{(N-2)!}{(N-i-1)!}$.

Summing up the paths for all possible path lengths from 1 to $N-1$, the number of possible loop-free paths between two vertices is

$$n_{paths,full} = \sum_{i=1}^{N-1} \frac{(N-2)!}{(N-i-1)!}. \tag{2}$$

Due to the factorial term, the amount of paths grows immensely with growing network size. For example, a fully connected mesh of 10 vertices has already 109601 possible loop-free paths between any pair of vertices. Assuming a route discovery process where route request messages are sent over the network and use (discover) all of the loop-free paths, route requests are propagating on each link of each path. But due to the branching at each vertex, the number of route requests fortunately is not the sum of all path lengths, as multiple paths share common subpaths. In fact, a path with a length of $i > 2$ shares all but 2 links with a path of a length of $i-1$ *in a full mesh*. Paths with a length of $i \leq 2$ do not have shorter paths with which they share links. Considering this, the total amount of route request transmissions accumulates to

$$n_{RREQs,full} = \sum_{i=1}^{N-1} \left[\min(i,2) \frac{(N-2)!}{(N-i-1)!} \right]. \tag{3}$$

In the 10 vertices example, these are as much as 219201 route request messages being transmitted during one route discovery.

Fortunately, networks are usually not fully connected meshs, but each of the vertices just has direct connections to a subset of the other vertices. Assuming an average vertex degree of K in a network of N vertices (i.e. in average, each of the N vertices has K outgoing links to neighbor vertices) and no further knowledge about the network's topology, there is a probability

$$P_{link} = \frac{K}{N-1} \tag{4}$$

for the existence of a direct link between a specific pair of vertices. As all links of a path have to exist if the path should exist, the probability for the existence of a specific path of length i is $P_{path} = P_{link}^i$.Taking this into account, an estimate for the number of loop-free paths between two vertices can be expressed as

$$n_{paths} = \sum_{i=1}^{N-1} \left[\frac{(N-2)!}{(N-i-1)!} \left(\frac{K}{N-1} \right)^i \right]. \tag{5}$$

For a small network with $N = 10$ and $K = 5$, this formula already gives an estimated average of 1229.56 paths between two vertices. This is two orders of magnitude less than the fully connected mesh presented before, but it is still a high number, considering the relatively small network size. Without any restrictions, route requests would be propagated along all these paths in a route discovery. As this becomes worse for larger networks, an efficient route request flooding restriction is needed.

V. Flooding restriction concept

The previous section has clearly shown that the flooding of route requests can cause a significant communication traffic volume, so that a flooding restriction is required. In section II, it was explained why the usual flood restriction in ad-hoc communication networks, which is done by only forwarding the first incoming route request, cannot be applied here. Therefore, to restrict the flooding of routing messages from autonomous logistic entities, a new restriction concept is proposed. This restriction concept consists of two components.

A. Hop limitation

An efficient way to limit the propagation depth of a route request is the use of a hard limit for the hop count. Each route request that has travelled a certain number of hops without reaching its destination is dropped. This is an approach which is also well-known in communication networks. To be able to apply this hop limitation, some knowledge about the dimensions of the network, e.g. the network diameter, has to be available, as the limit should be chosen such that it is still possible to reach any vertex in the network, but the route requests do not propagate unnecessarily far. Here, only the paths which are not longer than the hop count limit are valid. In a network with N vertices and an average vertex degree of K, the amount of loop-free paths between two vertices, restricted to those with a length of $l_{max} < N-1$ or less, is

$$n_{paths,l_{max}} = \sum_{i=1}^{l_{max}} \left[\frac{(N-2)!}{(N-i-1)!} \left(\frac{K}{N-1} \right)^i \right]. \quad (6)$$

B. Intermediate route evaluation

The second flood limitation component is the use of intermediate evaluations of the MCCD utility and its individual criteria to restrict the flooding.

The vehicle or cargo that initiates the route discovery adds forwarding limits for the individual criteria and for the MCCD utility result to the route request before the request is sent. When a vertex then receives the route request, it does not only add its local information to the request, but after that, it calculates the MCCD utility. Then it compares the result to the limits specified in the route request, and only if the limits are not violated, the route request can be forwarded. So the forwarding depends on the MCCD utility and the set of forwarding limits. Generally, the probability that a path is valid with respect to the limits depends on the length of the path, and it is denoted as $P_{val,i}$ for a path of length i here. With this, and without the hop limitation discussed before, the estimated number of valid paths between two vertices becomes

$$n_{paths,MCCD} = \sum_{i=1}^{N-1} \left[\frac{(N-2)!}{(N-i-1)!} \left(\frac{K}{N-1} \right)^i P_{val,i} \right]. \quad (7)$$

C. Combination of flood limitation components

Both flood restriction components are used subsequently to efficiently restrict the amount of communication traffic that is sent in a route discovery: First, it is checked whether the hop limit is reached, and if not, the criteria and the utility are compared against the thresholds. Only after successfully passing both stages, a route request is forwarded. The resulting number of valid paths using both restriction components is now

$$n_{paths,MCCD,l_{max}} = \sum_{i=1}^{l_{max}} \left[\frac{(N-2)!}{(N-i-1)!} \left(\frac{K}{N-1} \right)^i P_{val,i} \right]. \quad (8)$$

It has to be noted that the limits have to be chosen such that it is still possible to find a valid route. If no route was found, the route discovery has to be restarted with modified limits that allow more route requests to be forwarded.

VI. Simulative evaluation

Simulation results will be given which show that with the proposed flooding restriction method, the communication traffic in DLRP can be significantly reduced without causing degradations in the logistic performance.

A. Simulation scenario

The routing traffic limitation was evaluated by simulation of a scenario based on the topology depicted in figure 1. This topology represents a map with 18 German cities and motorway connections between these cities and has already been used in several previous publications [7], [3], [6].

Within the scenario, goods have to be transported between the vertices, and each of the vertices is source of some of the goods and destination of others. The goods are generated during simulation runtime and are supposed to be delivered within 25 hours of model time after their generation. 25000 goods are generated in total, and 12 vehicles, each with a capacity of 12 goods, are present in the scenario to fulfill the transport demand.

The goods are initiating a route discovery triggered by the following events:

- when the goods enter the scenario
- when they approach a new vertex, to evaluate whether their intended route is still good
- when they have not been picked up for a certain time, to evaluate whether better route options have emerged

B. Simulation results

To verify the flood limitation approach, two cases were investigated in the simulation: the hop limitation of goods requests, and the variation of the limit on the goods decision criterion “costs”.

Table I shows the effect of hop limitation for goods route requests. The results are based on 10 simulation runs with different random seeds for the generation of goods. The mean delay is the average time difference between the due time and the delivery time of goods. A negative delay means the goods were delivered within their given time window, a positive delay means the deliveries are too late. The capacity utilisation is the percentage of vehicle capacity that is occupied in average. The delivered goods are the goods that reached their destination within the observed time frame, which ends when the last of the 25000 goods was generated. The number of goods route requests is the overall number counted in the scenario within the observed time frame. It can be seen that while the hop limitation reduces the route requests, it does not have a negative impact on the logistic measures as long as it is not lower than 4. With a hop limit of 3, some goods cannot find routes to their destinations any more because they are more than 3 hops away. Therefore, the number of delivered goods is lower there, and only short routes are served, so that the mean delay for the fulfilled transport decreases.

TABLE I
DIFFERENT HOP LIMITS FOR GOODS ROUTE REQUESTS

Hop limit	Mean delay	Capacity utilisation	Delivered goods	Total goods RREQs
3	-6.3706	0.6214	21843.9	34999737
4	-3.4942	0.7534	24492.5	69821309
5	-3.2270	0.7592	24445.9	87145728
6	-2.8380	0.7612	24433.0	102386787
7	-2.8129	0.7632	24469.6	107767587

As a representation of the limitation based on the decision criteria, a variation of a limit on the goods criterion “costs” was investigated. It was assumed that the goods have a budget from which they pay the vehicles for being transported, and from which they also pay transshipment and storage costs.

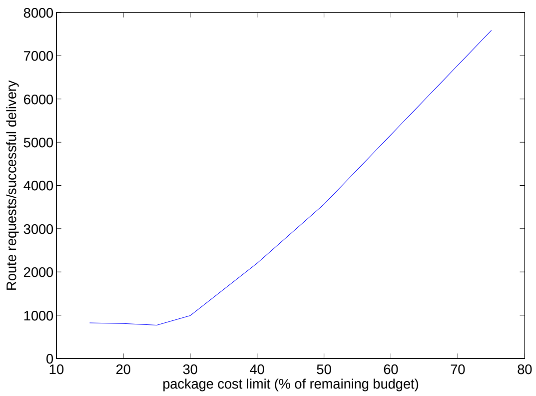

Fig. 2. Route request forwardings per successfully delivered good

Figure 2 shows that stricter limits on the costs can significantly reduce the generated route discovery traffic. There is a minimum visible around a value of 25% of the remaining budget. Lower limits then have a slightly adverse effect because they lead to more re-discovery attempts when no route was found due to the strict limits.

From figure 3, which shows the cumulative distribution functions for delivery delays under different cost limits, it can be seen that the delivery delays can even be slightly reduced. The reason is that with stricter limits, it is less likely that goods have only received replies of low-quality routes when they decide after having received a certain number of replies.

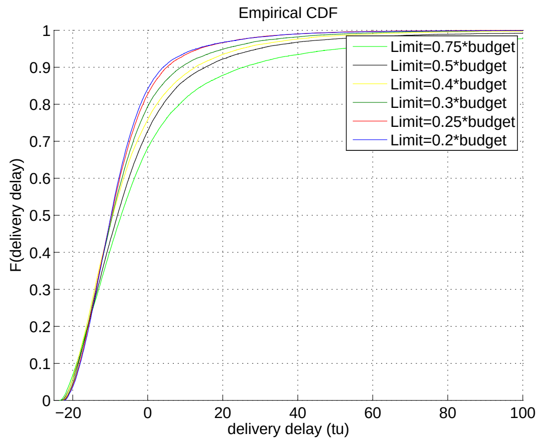

Fig. 3. Delivery delay cdfs with varying forwarding limit for goods costs

VII. CONCLUSION

In this paper, the route request flooding issue as an important issue for the scalability of distributed routing of autonomous logistic entities is investigated. It was shown that without countermeasures, the flooding can be a serious problem for scalability. To handle this problem, a two-stage method of route request forwarding limitations is presented. Simulation results have shown that this method can significantly reduce the communication traffic that is caused by the route discoveries for the DLRP-enabled autonomous logistic entities.

Future research will extend these investigations to higher topology scales in order to further improve the scalability of the routing concept.

ACKNOWLEDGMENT

This research was supported by the German Research Foundation (DFG) as part of the Collaborative Research Centre 637 “Autonomous Cooperating Logistic Processes”.

REFERENCES

[1] B. Scholz-Reiter and K. Windt and M. Freitag, *Autonomous Logistic Processes: New demands and first approaches*, in: Proceedings of the 37th CIRP international seminar on manfacturing systems, 2004, pp. 357-362.

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

[3] B.-L. Wenning and H. Rekersbrink and A. Timm-Giel and C. Görg and B. Scholz-Reiter, *Autonomous control by means of distributed routing*, in: Understanding Autonomous Cooperation and Control in Logistics - the Impact on Management, Information and Communication and Material Flow, Springer, 2007, pp. 335-345.

[4] C. Perkins and E. Belding-Royer and S. Das, *Ad hoc On-Demand Distance Vector (AODV) Routing*, IETF RFC 3561, 2003.

[5] D. Johnson and D. Maltz, *Dynamic Source Routing in Ad Hoc Wireless Networks*, in: Mobile Computing, 1996, pp. 153-181.

[6] B.-L. Wenning and H. Rekersbrink and A. Timm-Giel and C. Görg, *Weighted multiplicative decision function for distributed routing in transport logistics*, in: Proceedings of the 2nd International Conference on Dynamics in Logistics (LDIC 2009), 2009, to appear.

[7] B.-L. Wenning and C. Görg and K. Peters, *Ereignisdiskrete Modellierung von Selbststeuerung in Transportnetzen*, in: Industrie Management, volume 21, issue 5, 2005, pp. 53-56.