Path Heuristics using ACO for Inter-Domain Routing in Mobile Ad Hoc and Sensor Networks

Falko Dressler[1], Roman Koch[1], and Mario Gerla[2]

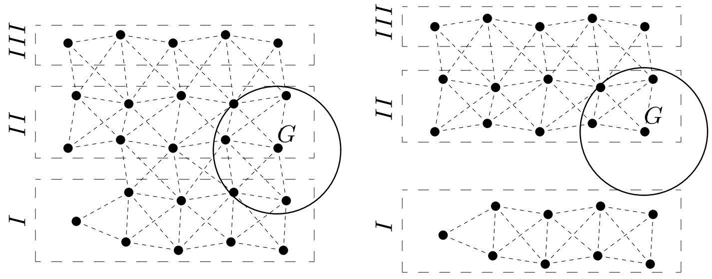
[1] Dept. of Computer Science, University of Erlangen, Germany
`dressler@cs.fau.de`
[2] Dept. of Computer Science, University of California, Los Angeles
`gerla@cs.ucla.edu`

Abstract. We investigate the use of biologically inspired routing heuristics in the field of inter-domain routing in sensor networks. Instead of relying on classical topology control techniques for routing in sensor networks, the use of geographical coordinates has been investigated for self-organized and fully distributed message forwarding. However, the identification of the nodes' positions is either expensive in terms of necessary equipment or message exchange. Therefore, the use of virtual coordinates has been investigated in this domain. The key advantage is that these virtual identifiers can also be used for data management similar as in a Distributed Hash Table (DHT). It is, however, extremely challenging to provide routing functionality between multiple independent networks or network domains. In previous work, we developed the Virtual Cord Protocol (VCP) that provides all the means for creating and maintaining such virtual identifiers and that is even able to route between neighboring network domains. This paper extends VCP by providing a generalized inter-domain routing framework using Ant Colony Optimization (ACO) for optimizing routes between multiple network domains. In extensive simulations, we evaluated this routing bio-inspired heuristic. The obtained results clearly demonstrate that ACO is very efficient even in highly mobile scenarios.

Key words: Inter-domain routing, virtual cord protocol, ant colony optimization, sensor networks

1 Introduction

Several classes of different routing techniques have been investigated in the field or sensor networks. The key objective is to cope with heterogeneity of nodes, dynamics of the environment, and, most importantly, the limited available energy resources [1]. Early approaches mainly focused on establishing routing tables similar to routing protocols studied in the field of Mobile Ad Hoc Networks (MANETs). However, it turned out that the inherent protocol overhead for topology control is not adequate to operate sensor networks for a longer lifetime [8]. Therefore, stateless approaches have been investigated such as geo-routing, where the content is represented by geographic coordinates of the destination. In this



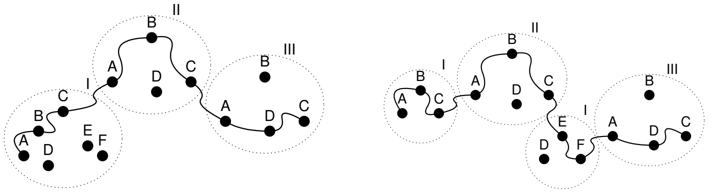
(a) (b)

Fig. 1: Schematic representation of partially connected sensor network domains

case, all nodes have geographic position identifiers (learned for example from GPS). Such position-based routing solutions inherently improve the situation as simple greedy routing towards the destination can be used. However, such approaches only work well if the network is dense, as routing holes cause geographic routing to rely on inefficient face routing methods. Additionally, GPS is required for all the nodes have to be able to precisely obtain their geographic locations, and a Geo Location Service (GLS) is necessary to find the destination coordinates [16]. Recently, a number of improvements to overcome geo-routing holes have been proposed. One idea is to “re-arrange” the nodes’ positions appropriately to prevent routing holes [15].

A conceptually more innovative concept is to rely on virtual coordinates only and to create an overlay network that connects the nodes and guides the search. Protocols like Virtual Cord Protocol (VCP) [3] and Virtual Ring Routing (VRR) [4] build their own coordinate system, which is completely independent of the geographic node positions. Furthermore, the virtual node positions can be used as IDs in a Distributed Hash Table (DHT) to efficiently store and retrieve data. Current work on virtual coordinate based approaches focuses on two aspects: The provided quality of service, which is mainly an issue of optimizing the delivery ratio or even providing guarantees [18, 19], and the reliability of the system as a whole, using data replication and other redundancy increasing techniques [2]. Such solutions are inherently self-organizing and scale extremely well even for large-scale networks [10].

Many scenarios can be envisioned in which multiple (virtual coordinate based) networks have to be established and maintained separately, yet with a strong demand to support routing among these different networks in case of connectivity. The problem is illustrated in Figure 1. As shown in Figure 1a, multiple network domains may be operated by protocols such as VCP, even though communication between the domains also becomes necessary. As depicted in Figure 1b, the connectivity between such domains might not be constantly available, e.g. domains move according to a group mobility model [22]. However, we assume that the network integrity (in terms of an ordered overlay) for a single domain is almost always ensured. Whenever two networks get into each other’s physical radio communication range, data can be exchanged between the



(a) (b)

Fig. 2: Topology changes at domain level: domain I splits and gets re-located

domains. A key challenge is to provide inter-domain routing between different local networks because the virtual coordinates are usually managed locally in each domain.

Inter-domain routing in MANETs has been first discussed in [5]. Four challenging issues have been identified: addressing, membership management, handling domain-level topology changes, and routing between the networks. As Internet-based protocols have been considered, the addressing and membership management basically targeted the IP address assignment procedure and the resulting routing problems. A cluster-based solution for inter-domain routing in MANETs has been described in [21]. Here, especially the issue of domain-level topology changes has been addressed. Using bloom filters, the effort for necessary topology updates was greatly reduced. Figure 2 outlines some of the most typical problems. At a macroscopic level, domain management techniques must be developed taking care of splitting and merging domains, and of domain-wide topology changes. On a microscopic level, different nodes will have to provide gateway functionality as soon as physical connection is available. The inter-domain routing is responsible for establishing adequate paths.

Motivated by this work, we investigated the issue of inter-domain routing for virtual coordinate based routing protocols, in particular focusing on our VCP approach [12, 13]. We were able to show that inter-domain routing in virtual coordinate environments can be established exploiting available DHT-based data management operations. Inter-domain routing between neighboring domains becomes feasible with only marginal overhead.

In this paper, we extend this previous work introducing a generalized routing framework for inter-domain routing in virtual coordinate based networks using bio-inspired techniques. We show how to establish topology information on a *macroscopic* domain level as well as on a *microscopic* gateway level. Furthermore, we used a routing heuristic based on Ant Colony Optimization (ACO) [9] to optimize both the macroscopic and the microscopic behavior even in very dynamic environments. The development of such self-organizing algorithms strongly depends on an optimal calibration of the system parameter [11]. Thus, we first investigated the configuration of the ACO algorithm using empirical studies. Using these results, we performed a detailed performance analysis of the developed ACO heuristics based inter-domain routing scheme. The results clearly indicate that the developed algorithm is extremely stable and robust to topology changes.

2 Virtual Cord Protocol

The Virtual Cord Protocol (VCP) has been developed for efficient routing and data management in sensor networks. In previous work, we demonstrated that VCP outperforms MANET-based solutions as well as other virtual coordinate-based protocols such as VRR [2, 3]. We continued this research by studying inter-domain routing between multiple VCP domains [12, 13]. In the following, after briefly outlining the concepts of VCP, we present our generalized inter-domain routing framework.

2.1 VCP cord management

The main idea is to arrange all the nodes in the network in form of a virtual cord. The topology of this cord must not be "optimal" in any sense, because routing is organized by exploiting information about the physical neighbors for greedy forwarding. Nevertheless, the cord ensures the availability of at least one path between any two nodes in the network for guaranteed delivery. The cord is established using periodic `HELLO` messages. Besides the assigned virtual address, these messages carry all relevant information including the physical and the virtual neighbors. Based on received `HELLO` messages (at least one is required) in the last `HELLO` interval, a new node can determine its position in the cord. A cord is formed according to a number of simple rules. Basically, new nodes either join at one end of the cord, or get integrated if at least two other nodes that are virtual neighbors in the cord are detected. A special rule is applied if the node has connectivity to a non-end node but not to its virtual neighbors. Then, a *virtual position* is generated at the discovered potential neighbor that is close to its virtual coordinate. This address allows the new to join between the real and the virtual position in the cord, i.e. to extend the cord without disrupting it. An application-dependent hash function is used for associating data items to nodes; thus, both pushing to a node and pulling data from a node are supported. The same mechanism can also be used for service discovery.

2.2 Basic Inter-Domain Routing

The basic inter-domain routing solution for VCP relies on unique domain identifiers. This can be performed in VCP during the cord setup phase by assigning this ID to the start node. The periodically exchanged `HELLO` messages also contain the domain ID, thus, joining nodes also received this domain identifier. Furthermore, if two networks are getting into each other's communication range, a node receiving `HELLO` messages from another domain automatically becomes a gateway node. It then stores this information into the local DHT by hashing the identifier of the router towards the neighboring network. If the gateway no longer receives `HELLO` messages from the detected neighbor, it removes the gateway information from the DHT. The basic procedure is depicted in Figure 3 (`G` and `R` denoting gateway and router nodes, respectively).

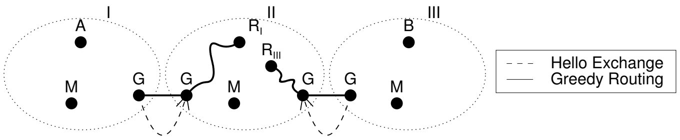
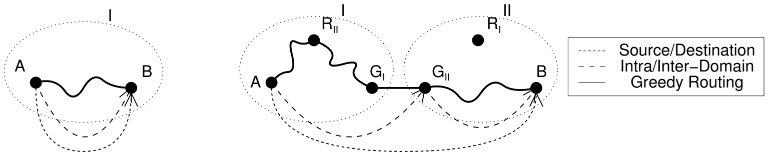


Fig. 3: Exchange and processing of received `HELLO` messages



(a) Intra-domain (b) Inter-domain between neighboring domains

Fig. 4: VCP routing within a domain and between neighboring domains

Direct communication between two nodes in arbitrary domains requires global topology information, i.e. the gateway information needs to be distributed into all VCP domains. VCP's greedy routing is only used within a domain (Figure 4a).

Inter-domain routing can be supported using the available router nodes, i.e. nodes storing information about neighboring domains [12]. Figure 4b outlines such a scenario. An indirection to the router node is used together with source routing on domain level. However, no transit domains are supported yet.

2.3 Extended inter-domain routing framework

In order to develop a generalized inter-domain routing framework VCP, we had to define several roles, which need to be executed by the network nodes:

- *Gateway* nodes are responsible for detecting neighboring domains, storing this information in the local DHT, and to provide forwarding capabilities to remote domains.
- *Router* nodes represent a virtual function storing all available gateways to a particular domain. They basically provide all the inter-domain routing functionality using indirections as known from peer-to-peer routing.
- *Moderator* nodes maintain, update, and exchange domain tables with moderators in remote domains. Thus, they are responsible for creating all the relevant domain-level topology information.

Figure 5 depicts the setup of routing information. After detecting neighboring domains using the `HELLO` mechanism, the gateway node forwards this information to a local router node responsible for the detected domain, i.e. a node storing information for the associated hash entry. If the local routing table changes, this information is further forwarded to the moderator node, and, via the basic VCP inter-domain routing also to moderators in the neighboring domains.

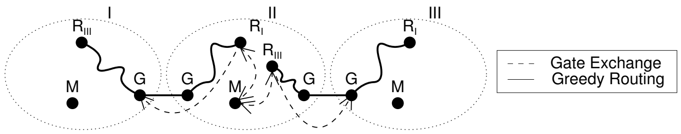


Fig. 5: Routing of `EXCHANGE_DOMAIN_SET` messages

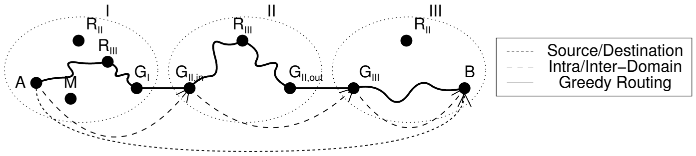


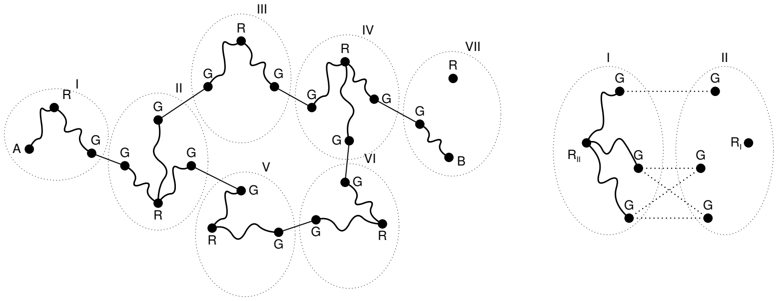
Fig. 6: Inter-domain routing between arbitrary domains

Based on the established routing information, messages can be routed within a domain using the standard VCP greedy routing techniques, between neighboring domains relying on the indirection via the router node, and now, even between arbitrary domains exploiting the knowledge provided by the moderator nodes. Figure 6 outlines the message forwarding over a transit domain. In the source domain, a router node ($\mathrm{R}_{\mathrm{III}}$) has been created by the moderator. Thus, a message towards domain III is first routed to $\mathrm{R}_{\mathrm{III}}$. The indirection points towards an adequate transit domain (here, domain II), to which the message is forwarded using an appropriate gateway node. From within domain II, the message is forwarded as described for inter-domain routing between neighboring domains.

3 ACO-based Routing Heuristic

Organizing inter-domain routing between arbitrary domains in an optimized way has a high complexity: First, the routing tables, i.e. the inter-domain network topology needs to be updated and maintained in order to ensure stable topology information and loop-free routes. This requires an extremely high amount of network traffic for topology control if dynamics and mobile nodes are considered. Secondly, the complexity of the routing tables and the paths that need to be calculated might be too high for use on embedded sensor nodes. Therefore, classical routing algorithms cannot be used, even on domain level. Basically, two problems need to be solved as illustrated in Figure 7:

- First, the inter-domain routing needs to be organized, i.e. the path between source and destination domains. This represents a *macroscopic* view to the routing problem. For example, as shown in Figure 7a, two possible routes exist between domains I and VII.
- Secondly, *microscopic* level problems need to be solved, i.e. which particular gateway node should be used for routing between two connected domains. This problem is outlined in Figure 7b.



(a) Macroscopic level (b) Microscopic level

Fig. 7: Routing problems at macroscopic (domain) and microscopic (gateway) level associated with inter-domain routing

In this paper, we propose a routing heuristic based on ACO that is able to cope with these two problems while ensuring a high degree of robustness to topology changes. Also, the selected solution is rather accurate, i.e. selected routes are close the theoretical shortest path. In the following, we outline the ACO based routing heuristic.

3.1 ACO

Ant Colony Optimization (ACO) is a biologically inspired technique simulating the foraging process of social insects [9, 14]. ACO uses a graph $G(N, E)$, N denoting the nodes and E undirected edges, respectively. Two nodes $i, j \in N$ are neighbors if $(i, j) \in E$. Each edge E_{ij} is annotated with some cost. A path is a sequence of nodes and edges between a source and a destination node. The objective of ACO is to find a path between source and destination with minimal costs.

During initialization, each edge $(i, j) \in E$ in the graph G is associated with some initial pheromone level (weight) τ_{ij}:

$$\tau_{ij} \leftarrow \tau_0, \forall (ij) \in E \tag{1}$$

A complete iteration of the ACO algorithm consists of three steps:

1. *Stepwise probabilistic solution*
   Setting up a path is based on stepwise estimation for each edge (i, j) according to Equation 2. Here, $\mathcal{N}_i^k$ depicts the neighborhood of the k-ths ant at node i.

$$p_{ij}^k = \begin{cases} \dfrac{\tau_{ij}^\alpha}{\sum_{l \in \mathcal{N}_i^k} \tau_{il}^\alpha} & j \in \mathcal{N}_i^k \\ 0 & j \notin \mathcal{N}_i^k \end{cases} \tag{2}$$

2. *Deterministic pheromone update*
   After finding a solution, the ant returns. On this path, loops are eliminated by checking whether a path includes the same node twice. Furthermore, the returning ant updates the pheromone level for all edges (i, j) on the path. The new pheromone concentration is calculated according to Equation 3.

$$\tau_{ij} \leftarrow \tau_{ij} + \Delta\tau^k \tag{3}$$

3. *Pheromone evaporation*
   In order to make the algorithm robust even in case oh high dynamics in the topology, the pheromone needs to be evaporated over time for all the edges. Basically, the pheromone level is decremented over time by some value $\rho \in (0, 1]$ as shown in Equation 4.

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij}, \forall (ij) \in E \tag{4}$$

The algorithm converges if a solution reaches some certain quality level or if no more changes are performed.

ACO has already been successfully applied to several problems in networking. Most importantly, early approaches to routing need to be named such as the AntNet [6] proposal. Here, ACO has been used to set up probabilistic routing tables for standard Internet routing. This work has been directly used in the AntHocNet [7] algorithm, which has been designed for use in MANET environments, thus, in very dynamic networks with rapidly changing network topologies. It turned out that ACO was perfectly able to handle these dynamics.

Hierarchical solutions relying on a combination of ACO and table-driven routing on a higher layer have been investigated for example in the HopNet approach [20]. In this paper, we use a similar scheme but using ACO on the higher (domain) level. As a further step, even combined routing and task allocation in mobile sensor networks has been investigated [17]. In this work, not only routing in mobile networks has been considered but also the distribution of multiple tasks to sensor nodes generating network traffic with different profiles (bursty, constant but high traffic volume, etc.). Obviously, ACO seems to be a perfect candidate for handling dynamics in the network topology with low overhead.

3.2 Optimized Inter-Domain Routing

In order to apply ACO to the problem of inter-domain routing in virtual coordinate based networks, we need to construct a graph, define a solution for the pheromone update, and find appropriate parameters for this update. We interpret the entire network as graph G and each VCP domain as a node $k \in G$. For each available gateway between two domains, we draw an edge $(i, j) \in E$ connecting the domains i and j. Thus, different to the classical ACO, we allow multiple edges between nodes. The resulting algorithm follows the ACO principles:

1. Set up a probabilistic route, estimate resulting costs;

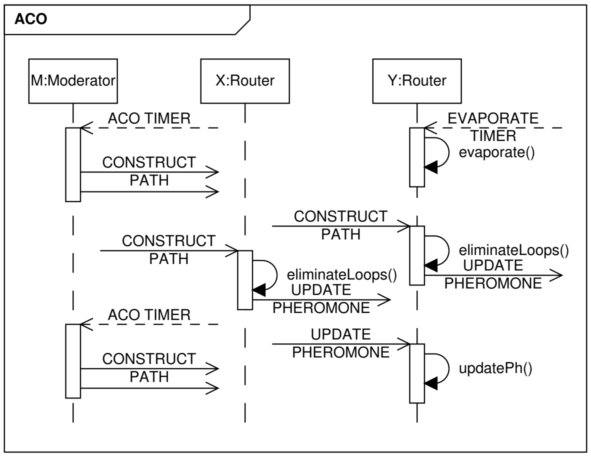


Fig. 8: Estimation of ACO-based routing heuristics

2. Prune routing loops, update the costs for each gateway such that the probability to chose the best gateway reflects the estimated costs;
3. Update the costs while the algorithm is running to incorporate dynamic topology changes.

ACO is thus used to weight the used gateways in order to find a shortest (cost minimal) path between two domains. In the scope of this paper, we use the hop count as a routing metric to derive the costs for each gateway. Topology control and minimal cost routing is performed using artificial ants transmitted between the VCP routing domains. This technique is very robust to changes in the network topology. Domains that become connected to the network can be quickly integrated using such explorer ants. Removing a domain because no more gateways are available leads to a short period of inconsistent routing (which is typical for ACO-based heuristics). However, as the costs decrease quickly, this has no influence on active parts of the network.

Figure 8 shows a sequence diagram outlining the stepwise creation of a solution as well as the update of the costs for the gateway nodes. After receiving the `ACO_TIMER` signal, the moderator of a domain initiates the setup of paths to each reachable domain. Thus, it sends `CONSTRUCT_PATH` to the router nodes responsible for the respective destination domains. In order to cope with the dynamic topology at domain level, the `CONSTRUCT_PATH` messages can limit the maximum costs and the maximum number of hops.

When a message arrives at the destination router, the path is cleaned up and an `UPDATE_PHEROMONE` message is sent back to the destination using source routing along the stored nodes. The pheromone value τ_G represents the cost of each gateway entry in the routing table. After receiving an `UPDATE_PHEROMONE`, the cost value of the gateway is updated according to Equation 5. The initial cost value is a small value τ_0, K represents the path costs.

Parameter	Values
t_{ACO}	$\mathbf{1}$
ρ	$0.05, 0.1, \mathbf{0.5}, 0.75$
$t_{Evaporate}$	$0.2, 0.5, \mathbf{1.0}$
$\Delta\tau$	$0.2, 0.5, \mathbf{1.0}$

Table 1: ACO parameter selection

$$\tau_G \leftarrow \tau_G + \Delta\tau, \text{ with } \Delta\tau = \frac{C}{1+K}, \; C = \text{const} \tag{5}$$

The use of path costs in the pheromone update as some nice properties [9]: The quality of a solution is increased, a good solution can already be found using only a few explorer ants, and the quality of the solution becomes almost independent of the parameter α in Equation 2. Thus, the cost of a gateway in the routing table is directly proportional to the length of the entire path to the destination domain.

The evaporation process runs in parallel with the cost update. The parameter ρ influences the speed and quality of the routing convergence. For $\rho = 0$, no convergence is to be expected and for large ρ, the algorithm quickly converges to suboptimal solutions. Furthermore, the degree of mobility needs to be considered for identifying an optimal value for ρ.

We analyzed all the ACO parameters using some initial simulations. More details on the simulation framework and the used parameters are discussed in Section 4. In this initial simulation, we used a network consisting of four domains of nine nodes. We calculated the optimal paths offline using the Dijkstra shortest path algorithm. All the analyzed parameters are listed in Table 1.

The parameter t_{ACO} depicts the time between the periodic evaluations of the routing table. The evaporation factor ρ describes the evaporation speed. The delay between two evaporations is defined by $t_{Evaporate}$. We further analyzed the impact of C indirectly represented by $\Delta\tau$.

As selected quality metrics, we analyzed the success rate, path length, and the difference between the discovered paths to the shortest path. Table 1 also shows the best parameters for our scenario (printed in bold) w.r.t. the selected quality metrics. We used these parameters for the performance evaluation in Section 4. Please note that if completely different scenarios are to analyzed, the parameter selection needs to be repeated.

4 Simulation Results

We investigated the feasibility and the performance of the inter-domain routing concept for VCP in several simulation scenarios. We used our implementation of VCP for the simulation tool OMNeT++ to analyze the behavior of the dynamic gateway configuration and the performance of the inter-domain routing using

Input Parameter	Value
Number of Nodes	10+10, 100+100, 40
Speed	fixed, $1\,\mathrm{m\,s^{-1}}$, $3\,\mathrm{m\,s^{-1}}$, $6\,\mathrm{m\,s^{-1}}$
Query period	$1\,\mathrm{s^{-1}}$
Initialization time	100 s
mac.bitrate	$2\,\mathrm{Mbit\,s^{-1}}$
mac.broadcastBackoff	31 slots
mac.maxQueueSize	14 packets
mac.rtsCts	false

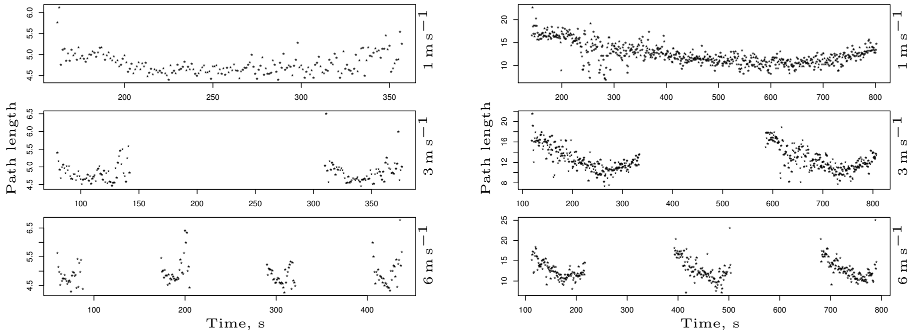
Table 2: Simulation parameters

indirections. The basic simulation parameters are summarized in Table 2. We simulated VCP over IEEE 802.11b wireless LAN.

In a first set of simulation experiments, we validated and compared the enhanced inter-domain routing algorithm for VCP to previous results obtained for inter-domain routing between neighboring domains [12]. We first used two networks consisting of 10 nodes each. We allow an initial setup time of 100 s to establish two VCP networks, one for each group. Within this time, a node in the mobile network creates and inserts data items in this VCP domain. After the initialization, the mobile group moves towards the stationary group. After some time, the first nodes get into the radio range of the other group and they start to set up gateway information, and to exchange data packets. The simulation time has been chosen such that for the slow $1\,\mathrm{m\,s^{-1}}$ scenario the mobile domain completely passes the stationary domain. At higher speeds, multiple of such connections occur. In the second scenario, we used network 100 instead of 10 nodes in both networks to evaluate the impact of a larger number of gateway nodes and longer communication paths.

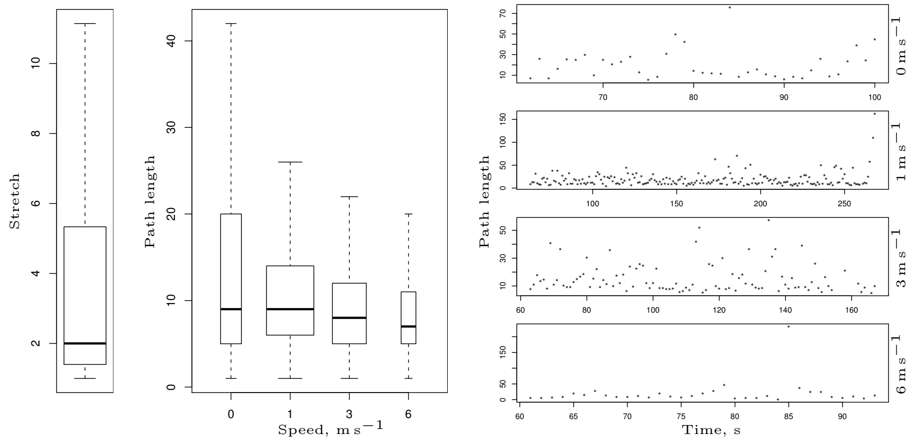
We evaluated a number of measures such as the available gateways, success rate, communication delays, and path length. All the results clearly show the feasibility of ACO to quickly find adequate routes in this simple setup. Exemplary, we analyzed the path length between source and destination nodes located in opposite domains. Figure 9 shows the simulation results for a sample run in form of a time series plot (we selected speeds of $1\,\mathrm{m\,s^{-1}}$, $3\,\mathrm{m\,s^{-1}}$, and $6\,\mathrm{m\,s^{-1}}$ according to the experimental setup in [21]). As can be seen, after both domains are getting into radio communication range, more and more gateways become available, resulting in decreasing path lengths. When both domains start to depart again, the path lengths are increasing due to the reduced number of gateway nodes. All the other metrics behave as expected from the earlier experiments without ACO optimization. All the other results (data not shown) validate the simple routing behavior.

In a second set of experiments, we evaluated the capabilities of the extended inter-domain routing framework using ACO for inter-domain routing including transit domains. We therefore created a setup including 10 VCP domains, 9 being



(a) Path length, $10 + 10$ nodes (b) Path length, $100 + 100$ nodes

Fig. 9: Distribution of the path length for the two domain scenario

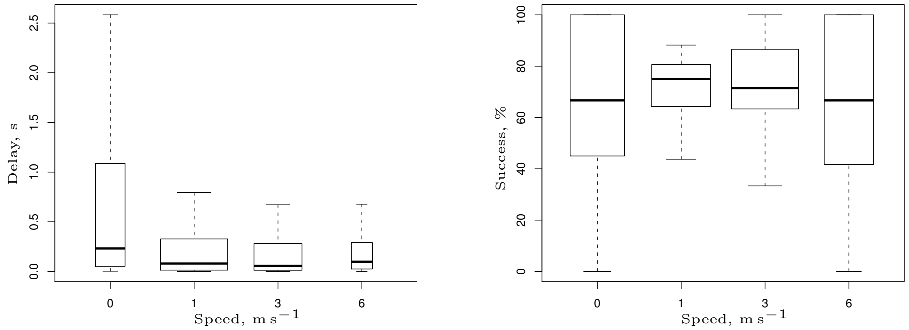


(a) Path stretch (b) Statistical distribution of path lengths (c) Time series plot of selected paths

Fig. 10: Distribution of the path length in the multi-network scenario

stationary arranged in form of a rectangle and one network domain being mobile, moving close to the border of the rectangle. Thus, the inter-domain routing framework had to keep track with a rather high degree of system dynamics, i.e. topology changes on domain level due to mobility.

As can be seen in Figure 10, the discovered paths have been quite stable even though the network topology continuously changed on domain level. ACO very quickly reacted on these changes and enforced the use of alternate paths. Figure 10a shows the path stretch, i.e. the distance of discovered paths from the theoretical shortest path. On average, roughly a factor two has to be considered, which is extremely promising for fully self-organizing routing protocols in highly



(a) Query delay (b) Success rate

Fig. 11: Query delay and success rate in the multi-network scenario

dynamic environments. Figures 10b and 10c show further statistics (statistical distribution, selected time series plots) of the path length for different speeds of the mobile domain.

We finally evaluated the query delay and the success rate for the multi-network scenario. As shown in Figure 11a, the transmission delay decreases with increasing dynamics, i.e. higher speeds of the mobile network domain. This effect was expected as the ACO algorithm needs some time to reinforce the use of alternate paths by updating the associated pheromone level. The setup for $6\,\mathrm{m\,s^{-1}}$ shows reduced delays, which is due to the scenario setting. The used MAC protocol tried to resent packets and the mobile domain returned to its previous position before the MAC layer gave up. The same effect can be seen in the plot of the success rate in Figure 11b. The success rate drops for increasing speed from a median of about 90 % in the stationary case to 80 % for the mobile $3\,\mathrm{m\,s^{-1}}$ case. However, for $6\,\mathrm{m\,s^{-1}}$ example, the success rate increases again to more than 90 %.

5 Conclusion and Further Challenging Issues

We studied the capabilities of a bio-inspired routing heuristic, the Ant Colony Optimization approach, for inter-domain routing in virtual coordinate-based network environments. We first established a generalized routing framework that is able to maintain information about inter-connected domains. In particular, the framework provides a microscopic view on gateways directly connecting neighboring domains and a macroscopic view on the high-level domain topology. Using indirections, routes can be established between arbitrary nodes in any domain. The ACO-based routing heuristic provides means for routing well suited even in networks with significant dynamics, i.e. established and broken connections between multiple domains due to node movements. According to our simulation results, the involved overhead is rather small and the obtained route information are close to shortest path solutions (which could only be calculated theoretically, because complete topology information cannot be centrally collected in such dynamic environments).

Open issues to be studied include the scalability of the approach w.r.t. the supported number of connected network domains. Furthermore, domain splitting is not yet considered. Split detection can be supported based on the neighborhood management. The main problem is to distinguish between a node failure and a domain split.

Acknowledgments

This work was partially supported by BaCaTec (project "support for inter-domain routing and data replication in virtual coordinate based networks").

References

1. K. Akkaya and M. Younis. A Survey of Routing Protocols in Wireless Sensor Networks. *Elsevier Ad Hoc Networks*, 3(3):325–349, 2005.
2. A. Awad, L. R. Shi, R. German, and F. Dressler. Advantages of Virtual Addressing for Efficient and Failure Tolerant Routing in Sensor Networks. In *6th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS 2009)*, pages 111–118, Snowbird, UT, February 2009. IEEE.
3. A. Awad, C. Sommer, R. German, and F. Dressler. Virtual Cord Protocol (VCP): A Flexible DHT-like Routing Service for Sensor Networks. In *5th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2008)*, pages 133–142, Atlanta, GA, September 2008. IEEE.
4. M. Caesar, M. Castro, E. B. Nightingale, G. O'Shea, and A. Rowstron. Virtual Ring Routing: Network routing inspired by DHTs. In *ACM SIGCOMM 2006*, Pisa, Italy, September 2006. ACM.
5. C.-K. Chau, J. Crowcroft, K.-W. Lee, and S. H. Y. Wong. Inter-Domain Routing Protocol for Mobile Ad Hoc Networks. In *ACM SIGCOMM 2008, 3rd ACM International Workshop on Mobility in the Evolving Internet Architecture (ACM MobiArch 2008)*, pages 61–66, Seattle, WA, 2008. ACM.
6. G. Di Caro and M. Dorigo. AntNet: Distributed Stigmergetic Control for Communication Networks. *Journal of Artificial Intelligence Research*, 9:317–365, December 1998.
7. G. Di Caro, F. Ducatelle, and L. M. Gambardella. AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks. *European Transactions on Telecommunications, Special Issue on Self-organization in Mobile Networking*, 16:443–455, 2005.
8. I. Dietrich and F. Dressler. On the Lifetime of Wireless Sensor Networks. *ACM Transactions on Sensor Networks (TOSN)*, 5(1):1–39, February 2009.
9. M. Dorigo and T. Stützle. *Ant Colony Optimization.* MIT Press/Bradford Books, 2004.
10. F. Dressler. *Self-Organization in Sensor and Actor Networks.* John Wiley & Sons, December 2007.

11. F. Dressler and O. B. Akan. A Survey on Bio-inspired Networking. *Elsevier Computer Networks*, 54(6):881–900, April 2010.
12. F. Dressler, A. Awad, and M. Gerla. Inter-Domain Routing and Data Replication in Virtual Coordinate Based Networks. In *IEEE International Conference on Communications (ICC 2010)*, Cape Town, South Africa, May 2010. IEEE.
13. F. Dressler, A. Awad, R. German, and M. Gerla. Enabling Inter-Domain Routing in Virtual Coordinate Based Ad Hoc and Sensor Networks. In *15th ACM International Conference on Mobile Computing and Networking (MobiCom 2009), Poster Session*, Beijing, China, September 2009. ACM.
14. F. Ducatelle, G. A. Di Caro, and L. M. Gambardella. Principles and applications of swarm intelligence for adaptive routing in telecommunications networks. *Swarm Intelligence*, 4(3):173–198, 2010.
15. R. Flury, S. V. Pemmaraju, and R. Wattenhofer. Greedy Routing with Bounded Stretch. In *28th IEEE Conference on Computer Communications (IEEE INFOCOM 2009)*, Rio de Janeiro, Brazil, April 2009. IEEE.
16. Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker. Geographic Routing Made Practical. In *USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI 2005)*, San Francisco, CA, 2005. USENIX.
17. T. H. Labella and F. Dressler. A Bio-Inspired Architecture for Division of Labour in SANETs. In F. Dressler and I. Carreras, editors, *Advances in Biologically Inspired Information Systems - Models, Methods, and Tools*, volume 69 of *Studies in Computational Intelligence (SCI)*, pages 209–228. Springer, Berlin, Heidelberg, New York, July 2007.
18. C.-H. Lin, B.-H. Liu, H.-Y. Yang, C.-Y. Kao, and M.-J. Tasi. Virtual-Coordinate-Based Delivery-Guaranteed Routing Protocol in Wireless Sensor Networks with Unidirectional Links. In *27th IEEE Conference on Computer Communications (IEEE INFOCOM 2008)*, Phoenix, AZ, April 2008. IEEE.
19. M.-J. Tsai, F.-R. Wang, H.-Y. Yang, and Y.-P. Cheng. VirtualFace: An Algorithm to Guarantee Packet Delivery of Virtual-Coordinate-Based Routing Protocols in Wireless Sensor Networks. In *28th IEEE Conference on Computer Communications (IEEE INFOCOM 2009)*, Rio de Janeiro, Brazil, April 2009. IEEE.
20. J. Wang, E. Osagie, P. Thulasiraman, and R. K. Thulasiram. HOPNET: A Hybrid ant colony OPtimization routing algorithm for Mobile ad hoc NETwork. *Elsevier Ad Hoc Networks*, 7(4):690–705, 2009.
21. B. Zhou, Z. Cao, and M. Gerla. Cluster-based Inter-domain Routing (CIDR) Protocol for MANETs. In *6th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS 2009)*, pages 19–26, Snowbird, UT, February 2009. IEEE.
22. B. Zhou, K. Xu, and M. Gerla. Group and Swarm Mobility Models for Ad Hoc Network Scenarios Using Virtual Tracks. In *MILCOM 2004*, Monterey, CA, 2004.