Ant Colony Optimization for Routing in Mobile Ad Hoc Networks in Urban Environments

Gianni A. Di Caro, Frederick Ducatelle, and Luca M. Gambardella

Technical Report No. IDSIA-05-08
May 2008

IDSIA / USI-SUPSI
Dalle Molle Institute for Artificial Intelligence
Galleria 2, 6928 Manno, Switzerland

This report represents the draft English version of the book chapter (in French) *Optimisation par colonie de fourmis pour le routage dans les rseaux mobiles ad hoc en environnement urbain*, that will appear in the book: Nicolas Monmarché, Frédéric Guinand, and Patrick Siarry, Eds., *Fourmis artificielles, des bases algorithmiques aux concepts et ralisations avancs*, Hermès Science Publications, 2008.

Contents

Ant Colony Optimization for Routing in Mobile Ad Hoc Networks in Urban Environments

Gianni A. Di Caro, Frederick Ducatelle, and Luca Maria Gambardella
Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA)
Galleria 2, CH-6928 Manno-Lugano, Switzerland
{gianni, frederick, luca}@idsia.ch

1 Introduction

In this report, we describe the application of Ant Colony Optimization (ACO) [14, 15] to a dynamic on-line optimization problem, namely *routing in mobile ad hoc networks in urban environments*. Routing is the task of finding and using paths to direct data flows through a network while optimizing one or more performance measures. This often comes down to a problem of finding minimum cost paths between pairs of source and destination nodes in the network. Hence, the problem of routing maps rather well to the solution model most commonly used in ACO, which is inspired by the ability of certain types of ants in nature to find the shortest path between their nest and a food source through a distributed process based on stigmergic communication [7]. An important aspect of routing, which sets it apart from many other applications of ACO, is that it is typically a distributed and dynamic problem, which means that the description of the problem changes over time and decentralized solutions must be adopted. This is because the situation in the network changes, e.g. because the traffic process at the nodes varies, or because there are link or node failures. As a consequence, the optimization algorithm for routing needs to adapt continuously.

Here, we focus on routing in a specific type of communication networks, namely *mobile ad hoc networks* (MANETs) [31]. These are networks that consist entirely of wireless nodes, placed together an ad hoc manner (i.e., on-the-fly, or with minimal prior planning) and without the support of a fixed communication infrastructure. All nodes are mobile, and can enter or leave the network at any time. Data are forwarded among the nodes of the network in multi-hop fashion. An example could be a network created by users carrying WiFi enabled laptops or palmtops operating in ad hoc mode, or a network created among moving cars that are enabled with wireless technology (in this case we also talk of vehicular networks [24]). MANETs are substantially different from more traditional wired communication networks such as the Internet. They are highly dynamic, have severe restrictions on the effective usable bandwidth (mainly due to the sharing of the wireless medium) (e.g., [18, 23]), have limited battery power available at each node, are based on the use of possibly unreliable wireless communication channels, etc. Algorithms and protocols for MANETs should be adapted to deal with these challenging properties. In this report we show how techniques from ACO can be applied to support routing in this kind of networks. We focus in particular on MANETs deployed in urban environments, which are confronted with specific conditions in terms of the network node movement patterns and the wireless radio propagation.

In the rest of this report we first provide some more background on MANET routing. Then, we explain the main ideas behind existing work on ACO for routing. Next, we present the *AntHocNet* routing algorithm, which applies ACO routing in MANETs. After that, we describe the properties of the urban scenarios used in our studies, and finally we present the results of a set of experiments in which we study the general characteristics of urban MANETs, and we evaluate the performance of the AntHocNet routing algorithm in these scenarios compared to AODV [29], a state-of-the-art MANET algorithm.

2 Routing in mobile ad hoc networks

The task of routing is particularly hard in MANETs. Due to the ad hoc and dynamic nature of these networks, the topology can change continuously, and paths between sources and destinations that were initially efficient can quickly become inefficient or even infeasible. This means that routing information should be updated more regularly than in traditional wired telecommunication networks. However, this can be a problem in MANETs, because they typically have limited bandwidth and node resources, and make use of possibly unreliable wireless communication channels. New routing algorithms are therefore needed, which can give adaptivity in an efficient and robust way.

A lot of research has been carried out on routing in MANETs in recent years (see [4] for a survey). Existing MANET routing algorithms can be classified as being *proactive, reactive* or *hybrid* [31]. Proactive algorithms try to maintain up-to-date routes between all pairs of nodes in the network at all times. The advantage is that routing information is always readily available when data need to be sent, while the main disadvantage is that the algorithm needs to keep track of all topology changes, which can become difficult when there are a lot of nodes or when they are very mobile. Examples of proactive algorithms are Destination-Sequence Distance-Vector routing (DSDV) [28] and Optimized Link State Routing (OLSR) [9]. Reactive algorithms only maintain routing information that is strictly necessary: they set up routes on demand when a new communication session is started, or when a running communication session falls without route. This approach is generally more efficient, but can lead to higher delays as routing information is often not immediately available when needed. Examples of reactive routing algorithms include Dynamic Source Routing (DSR) [22] and Ad-hoc On-demand Distance-Vector routing (AODV) [29]. Finally, hybrid algorithms use both proactive and reactive elements, trying to combine the best of both worlds. An example is the Sharp Hybrid Adaptive

Routing Protocol (SHARP) [30].

So far, most of the work has been carried out in simulation and has addressed simplified scenarios considering open space environments, random mobility patterns, and generic traffic loads. In this report we introduce a novel hybrid routing algorithm and we study its behavior and performance in a simulated urban environment considering realistic radio propagation, user mobility, and data traffic. Urban environments pose a number of unique challenges that set them apart from the open space cases. This kind of studies is fundamental to pave the road to the practical application of MANETs in real-world environments, which is still very limited.

3 Ant Colony Optimization for routing: general principles

ACO routing algorithms take inspiration from the behavior of ants in nature and from the related field of ACO to solve the problem of routing in communication networks. The main source of inspiration is found in the ability of certain types of ants (e.g. the family of Argentine ants Linepithema Humile) to find the shortest path between their nest and a food source using a volatile chemical substance called *pheromone*. Ants traveling between the nest and the food source leave traces of pheromone as they move. They also preferentially go in the direction of high pheromone intensities. Since shorter paths can be completed faster, they receive higher levels of pheromone earlier, attracting more ants, which in turn leads to more pheromone. This positive reinforcement process allows the colony as a whole to converge on the shortest path. It forms the basis of most of the work in the field of ACO.

As pointed out earlier, the problem of finding shortest paths maps quite well to the problem of routing in networks. Moreover, the ability to solve these problems in a distributed way is important in communication networks, as these usually operate without a point of central control. It comes to no surprise therefore that routing was one of the early application areas of ACO. Early work on ACO routing includes the *Ant-Based Control* algorithm (ABC) [35] for circuit-switched wired networks and the *AntNet* algorithm [11] for packet-switched wired networks. Over the years, many variations and improvements of these algorithms have been proposed, as well as applications to different kinds of network (see for example the *Q-Colony* algorithm for routing in Quality-of-Service networks [40]).

The main idea behind all of these algorithms is that nodes in the network periodically and asynchronously send out *artificial ants* towards possible destination nodes of data. These ants are small control packets, which have the task to find a path towards their destination and gather information about it. Like ants in nature, artificial ants follow and drop pheromone. This pheromone takes the form of *routing tables* maintained locally by all the nodes of the network. They indicate the relative quality of different routes from the current node towards possible destination nodes. Ants normally take probabilistic routing decisions based on these pheromone tables, giving a positive bias to routes of higher pheromone intensity, in order to balance exploration and exploitation of routing information. Often, the tasks of following and updating pheromone are split between a *forward* and *backward* ant, whereby the forward ant finds a path towards the destination and the backward ant travels back over the path to update pheromone tables. The result of the continuous ant sampling process is the routing information in the pheromone tables, which is used to forward data. This can again be done probabilistically, or deterministically following the path with the highest pheromone level.

ACO routing algorithms boast a number of interesting properties compared to traditional routing algorithms. First of all, they are *adaptive*, thanks to the use of continuous path sampling and probabilistic ant forwarding, which leads to an uninterrupted exploration of the routing possibilities. Next, they are robust. This is because routing information is the result of the repeated sampling of paths. On the one hand, the different samples are to some extent redundant, and the algorithm can therefore support packet loss. On the other hand, the use of sampling implies that routing information is based on direct measurements of the real network situation, which enhances its reliability. Finally, ACO routing algorithms can usually set *multiple paths*, over which data packets can be forwarded probabilistically like ants. This can result in *throughput optimization, automatic data load balancing*, and increased *robustness to failures*.

For routing in MANETs, especially the properties of adaptivity and robustness are particulary interesting, as was pointed out in section 2, so that ACO routing algorithms could form a possible solution to deal with the challenges found in such networks. However, the fact that these algorithms rely on the continuous generation of small ant packets to gather routing information can easily lead to excessive overhead in the bandwidth limited MANETs. This is especially problematic since MANETs usually rely on contention based mechanisms for medium access control, which essentially means that a high number of small packets generates much more overhead than a lower number of larger packets. The challenge is to develop an ACO routing algorithm that can offer adaptivity and robustness while keeping overhead limited. In what follows, we describe the AntHocNet routing algorithm, which solves this issue by taking a hybrid approach that combines ant based path sampling with other mechanisms for learning.

4 The AntHocNet routing algorithm

In this section we describe *AntHocNet*, an ACO routing algorithm for MANETs. AntHocNet can be considered a hybrid algorithm, since it contains both reactive and proactive elements. It is *reactive* in the sense that it only gathers routing information about destinations that are involved in communication sessions. It is *proactive* in the sense that it tries to maintain, improve and extend routes while the communication session is going on. This hybrid architecture improves the efficiency by focusing efforts on ongoing sessions. Another factor to help the efficiency is found in the organization of the proactive route maintenance and improvement process, which combines ant-based path sampling with other forms of information gathering. Routing information in AntHocNet is stored in *pheromone tables*. Forwarding of ant and data packets is done in a stochastic way, using these tables. Link failures are dealt with using specific reactive mechanisms, such as local route repair and the use of warning messages. Below, we describe the general working of the AntHocNet routing algorithm. For detailed accounts, we refer to [12, 17, 16].

4.1 Pheromone tables

Routing information is organized in *pheromone tables*. Each node i maintains one pheromone table $\mathcal{T}_i$, which is a two-dimensional matrix. An entry $\mathcal{T}_{ij}^d$ of this matrix contains information about the route from node i to destination d over neighbor j. This information includes the pheromone value τ_{ij}^d, which is a value indicating the relative goodness of going over neighbor j when traveling from node i to destination d, as well as statistics information about the route, and possibly *virtual pheromone*. This latter value is derived from the normal pheromone, which we will from now on refer to as regular pheromone, and is used to support proactive route maintenance an improvements. More details about virtual pheromone will follow in subsection 4.3. Apart from a pheromone table, each node also maintains a *neighbor table*, in which it keeps track of which nodes it has a wireless link to.

4.2 Reactive route setup

At the start of a new communication session, the source node s of the session checks its pheromone table, to see whether it has any routing information available for the session's destination d. If it does not, it starts a reactive route setup process, in which it sends a *reactive forward ant* out to find a route to the d. Since s has no information about where to find d, it broadcasts the ant, i.e. sends it to all its neighbors simultaneously. Each node i in the network that receives a copy of the reactive forward ant checks whether it is the ant's final destination, and if this is not the case forwards the ant in turn. This forwarding can be done by broadcasting, in case i does not have any routing information available for d, or by unicasting, in which case i chooses a next hop for ant probabilistically based on its pheromone entries for d. The probability P_{in}^d of choosing a next hop n from among i's set of neighbors N_i is calculated using equation 1, whereby β_1 is a parameter to tune how much preference is given to higher pheromone values.

$$P_{in}^d = \frac{(\tau_{in}^d)^{\beta_1}}{\sum_{j \in N_i} (\tau_{ij}^d)^{\beta_1}}, \quad \beta_1 \geq 1, \tag{1}$$

The first copy of the reactive forward ant to reach d is converted into a *reactive backward ant*, while subsequent copies are destroyed. The reactive backward ant retraces the exact path that was followed by the forward ant back to s. On its way, it collects quality information about the route, which is calculated based on the measured strength of the wireless signals on the links along the route. At each intermediate node i and at the source s, it updates the pheromone values with respect to d based on this quality information. This update is done using an exponential moving average, as follows:

$$\tau_{in}^{d} = \gamma \tau_{in}^{d} + (1-\gamma) q_{i}^{d}, \quad \gamma \in [0,1], \tag{2}$$

whereby n is the upstream node from i in the route, γ is the smoothing factor of the moving average and q_{in}^{d} is the measured route quality. In this way, a first route between source and destination is established at completion of the reactive route setup process. The full process is repeated later if the source node falls without valid routing information for the destination of the session while data still need to be sent.

4.3 Proactive route maintenance and improvement

Once a first route is made available between the source and destination of a data session, and for the entire duration of the session, the algorithm carries out the proactive route maintenance and improvement process. This process is aimed at finding new and better routes, and extending the single route created by the reactive route setup to a full mesh. It consists of two subprocesses: pheromone diffusion and proactive ant sampling.

The aim of the *pheromone diffusion* subprocess is to spread out pheromone information placed by the ants. It is implemented using update messages that are broadcast periodically and asynchronously by all the nodes of the network to their neighbors. In these messages, a sending node n places the list of destinations it has information about, including for each destination d its best pheromone value $\hat{\tau}_{n}^{d} = \max_{j \in N_n}(\tau_{nj}^{d})$. A node i receiving the message from n can then derive an estimate of the goodness of going from i over n to each of the destinations d, by combining the quality of the link from i to n with the reported pheromone value $\hat{\tau}_{n}^{d}$. The result is the virtual pheromone value ω_{in}^{d}, which is stored in i's pheromone table. This virtual pheromone can in turn be forwarded in the next periodic update message sent out by i, so that eventually a virtual pheromone field pointing to each of the destinations d is created in the MANET. The general way of working of the pheromone diffusion process is based on principles of distributed dynamic programming, and can be found back in many other algorithms, such as for example in *Bellman-Ford* routing algorithms [6]. In reinforcement learning, this approach is referred to as *information bootstrapping* [39]. The advantage of information bootstrapping is that it is a very efficient way of spreading information, as the estimates calculated by the nodes are fully reused by neighboring nodes. Its main disadvantage is that it can be slow to converge to correct estimates and might therefore give suboptimal results in highly dynamic non-stationary environments. This is why we combine it with the second subprocess, the proactive ant sampling.

The *proactive ant sampling* subprocess follows the typical mode of operation of ACO routing algorithms. The source node s periodically sends out proactive forward ants towards the destination d. These ants construct a path in a stochastic way, choosing a new next hop probabilistically at each intermediate node. The probability for an ant in a node i to choose next hop n is given in equation 3. As can be seen from the formula, proactive forward ants can follow both regular and virtual pheromone. Different from reactive forward ants, they are never broadcast: when the reach a node where no routing information about d is available, they get dropped. Once a proactive forward ant reaches the destination, it is converted into a proactive backward ant that travels back to the source and leaves (regular, not virtual) pheromone along the way, just like reactive backward ants. This way, each proactive ant explores a route indicated by virtual pheromone, and then, once it has verified that it is correct and leads to the destination, marks it with regular pheromone.

$$P_{in}^{d} = \frac{(\max(\tau_{in}^{d}, \omega_{in}^{d}))^{\beta_2}}{\sum_{j \in N_i} (\max(\tau_{in}^{d}, \omega_{in}^{d}))^{\beta_2}}, \quad \beta_2 \geq 1, \tag{3}$$

Using these two subprocesses, the proactive route maintenance and improvement process combines two approaches to learning: information bootstrapping and ant based sampling. Information bootstrapping provides an efficient way of spreading routing information, while ant sampling provides robustness and reliability.

4.4 Data forwarding

Data packets are forwarded from their source to their destination like ants, choosing a next hop *probabilistically* at each node. They follow thereby only regular pheromone, as virtual pheromone is potentially unreliable and is only meant to guide the ants' exploration process. The formula used to calculate the probability of each possible next hop for data packets is the same as in equation 1, but with a different parameter β_3 instead of β_1. β_3 is normally higher than β_1, so that data focus more on the best routes. By setting β_3 to infinity, deterministic routing over the best route can be obtained.

4.5 Dealing with link failures

Link failures are detected in AntHocNet via failed transmissions of data or control packets, or through the use of the pheromone diffusion messages playing the role of so-called *Hello messages*. Hello messages are short messages that are periodically sent out by all nodes in the network. The reception of a hello message is indicative of the presence of a wireless link, while the failure to receive such messages points to the absence of a link. In practice in AntHocNet, the function of hello messages is fulfilled by the same periodic update messages that are used for pheromone diffusion. Upon detection of a link failure, nodes control their pheromone table, to see which routes become invalid due to the failure, and whether alternative routes are available for the affected destinations. Then, the node broadcasts a link failure notification message to warn neighboring nodes about all relevant changes in its pheromone table. Neighbors receiving this message update their routing information accordingly and if this leads to the loss of a route for them too, they send out their own notification message. In case the link failure was associated with a failed data packet transmission, the node can also start a local route repair to restore the route to the destination of this data packet. To this end, it sends out a repair forward ant. *Repair forward ants* are similar to reactive forward ants, in the sense that they follow available pheromone information where possible (using equation 1), and are broadcast otherwise, but they have a limited maximum number of broadcasts, so that they cannot travel far from the old failed route. Upon arrival at the destination, the repair forward ant is converted into a repair backward ant that travels back to the node that started the repair process and sets up the pheromone for the repaired route. A last tool in dealing with link failures is the use of *unicast warning messages*. These are needed when data packets for a lost destination still arrive at the node after a link failure notification has already been sent. This can be due to bad reception of the broadcast notification message. In this case, the node sends a warning to the node it received the data from, in order to inform it that it can no longer forward data for this destination.

5 Working in an urban environment

In this report, we are particulary interested in the evaluation of our AntHocNet routing algorithm in MANETs that support *interactive communication in an urban environment*. The use of this specific type of scenarios has a strong influence on important aspects of the working of the MANET. First of all, the structure of the urban environment defines possible movement patterns for the nodes of the network. Second, the presence of buildings and other obstacles has a strong impact on the way radio waves can propagate and hence influences the connectivity between the nodes of the network. Finally, the interactive communication defines the data load and data traffic patterns. These properties set these scenarios apart from the kind of settings applied in most MANET evaluation studies, which rely on open space scenarios, random movement patterns and random communication patterns. We believe that the study of urban scenarios is important as this will become an important application area for MANET technology in the near future. Recent projects with wireless mesh networks, which are static ad hoc networks, in large cities such as Taipei [2] and Philadelphia [3] also point in this direction.

In what follows, we discuss the mentioned aspects of node mobility, radio wave propagation and data communication patterns in turn. We describe their effects and explain how we modeled them in the simulation study that will be presented in the next section. After that, we also give a short overview of other studies that focus on the use of MANETs in urban settings.

Figure 1: The setting of our study: an area of $1561 \times 997\ m^2$ in the center of the Swiss town of Lugano.

5.1 The urban environment and node mobility

The urban setting used in our simulation study is the center of the southern Swiss town of Lugano. Lugano is a relatively small old town presenting an irregular street topology common to most European cities. We focus on an area of $1561 \times 997\ m^2$, which covers most of downtown Lugano. The street structure is shown in figure 1. As shown in the figure, the cityscape is composed of streets (the white lanes) and buildings (the gray polygons). Streets define the open spaces where nodes are free to move. Buildings are in our study inaccessible to the nodes and basically play the role of obstacles that put constraints on node movements and shield radio wave propagation. Other elements are the lake, in the bottom of the image, and urban infrastructures. However, these latter do not play any role and are left in the image for the sole purpose of showing the town organization.

To define node movement patterns, we used an urban mobility model based on the so-called *random waypoint model* (RWP) [22]. Under the RWP model, nodes choose a random destination and speed, move in a straight line to the chosen destination at the chosen speed, and then pause for a certain time before picking a new destination and speed. In our urban version of RWP, destinations are only chosen from among the open spaces in the town, and nodes do not move along a straight line to their destination, but instead follow the shortest path through the streets of the town. In order to define node destinations and movements, we derived a graph representing the street structure of the town. Destinations were chosen from among all points that are located on an edge or in a vertex of the graph, and shortest paths were calculated in the graph using Dijkstra's algorithm [27]. We have chosen maximum node speeds that correspond to realistic inner city movements: from 3 m/s (10.8 km/h) to represent pedestrians or cyclists up to 15 m/s (54 km/h) to represent cars. The pause time of our RWP model is 30 s. Finally, we keep 20% of the nodes static, to represent immobile network users. These can for example be wireless access points placed by shop or restaurant owners, or infrastructure nodes provided by the town authorities.

5.2 Radio propagation

Wireless communication in an urban environment is strongly conditioned by the way radio waves interact with the objects they encounter. The most basic effect is that waves produced at street level are blocked by buildings, so that connectivity in urban wireless networks is restricted compared to open space scenarios. Some urban simulation studies for MANETs in the literature only account for this effect, using open space propagation models along the line of sight (LoS) and blocking any non-LoS communication (see e.g. [25]). In our study, we use a more detailed approach, which incorporates also other propagation effects. The most important of these effects is reflection off buildings: as radio rays bounce off building walls, they can travel around corners into side streets. Also, reflection allows a signal to travel further along the LoS through a street than it would in open space, since multiple reflected rays are tunneled in the same direction. This means that crude approximation models that do not account for reflection are too restrictive. Another important effect is diffraction, which allows rays to bend around corners to a certain extent. This further improves connectivity into side streets. Other effects include scattering, which is the reflection off small objects and uneven surfaces, and signal variations over time due to changes in the environment, such as the passing of vehicles or people. Both of these last effects are hard to model correctly and greatly increase the computational complexity (see [37]), and are therefore not taken into account here.

Making detailed calculations of radio wave propagation in an environment with many obstacles is a computationally intensive task, especially when many simultaneous transmitters and receivers are involved, as is the case in MANET experiments. For our simulation study, we needed to make these calculations in an efficient way and hence decided to do them off-line in a preprocessing step. We started from the two-dimensional map of the center of Lugano, and assumed each building on the map to be of a height that is sufficient to block radio communication going over it (a height of 5 meters already makes diffraction over a building impossible [37]). Then, we defined a discretization of the space in which the nodes move, in order to allow an efficient preprocessing: we chose sample points at regular intervals of 5 meters along the streets of the town, resulting in 6070 different positions. In each of these points, we placed a transmitter sending at 2.4 GHz and with a power of 10 mW, and we calculated with which strength its signal was received in each of the other points. All radio propagation calculations were done using the *WinProp* tool [5]. This is a commercial software package to model ray propagation in urban environments. WinProp makes use of raytracing, a technique derived from the field of computer graphics in which the trajectories of all the waves going between a transmitter and a receiver are first calculated individually, and then combined in order to derive the resulting signal [36]. The results of the preprocessing calculations were stored in a matrix of 6070×6070 entries. During the simulation, the signal strength between a transmitting node a and a receiving node b was approximated by the precalculated signal strength between a transmitter in the sample point closest to a and a receiver in the point closest to b. This allows very fast calculations, while only causing an inaccuracy of maximally 2.5 meters on each side.

5.3 Data traffic

For data traffic we use patterns that can reflect realistic applications of the network. We assume that the MANET will in the first place be used to support interactive communication between users. We model this type of applications using *bidirectional point-to-point data communication sessions*. The data packet size is 160 bytes. We consider a range of different data rates, from 1 packet every 30 seconds, representing an interactive *SMS conversation*, up to 25 packets per second, which is sufficient to support good quality *voice-over-IP (VoIP)* applications. In order to represent silent periods in the interactive communication, only 40% of all scheduled packets are sent. This corresponds to the typical proportion of send time in VoIP traffic [21].

5.4 Related work on the simulation of MANETs in urban environments

While there exists a lot of work on the evaluation and comparison of different routing algorithms for MANETs (e.g., see [8, 10]), most of it is carried out using open space scenarios with random mobility and idealized signal propagation models. Only recently some studies have appeared that investigate the use of MANETs in urban environments. Here we provide a short overview of this work. None of these studies include the evaluation of an ACO routing algorithm.

One could make a distinction between simulation studies that make rather *rough approximations* of how an urban setting influences node mobility and radio wave propagation, and studies that aim at *high accuracy*, thereby sacrificing efficiency. The work presented in [20] belongs to the first category. The authors propose a scenario with randomly placed building blocks, whereby node mobility is limited to paths between the buildings and radio wave propagation is implemented using a simple LoS approach: nodes can only communicate when there is no building between them. The authors investigate how this setup influences the behavior of the AODV routing protocol in comparison to open space scenarios and notice a severe drop in performance. The authors of [25] follow a similar approach, but with buildings placed according to a regular grid pattern, rather than randomly. They evaluate how the performance of the DSR routing protocol is influenced by the urban setting and find a drop in performance that is however not as strong as the one observed in [20]. Finally, the authors of [19] use a similar grid town pattern, but with a different radio propagation model: radio signals are weakened with a fixed amount for every corner they take. The aim of the paper is to investigate whether a MANET could be used to support communication between a fleet of taxis. Node movement patterns in the grid world are based on data about the behavior of real taxis. The authors find that a high density of users in the system is critical for good performance.

Among the studies that apply a higher level of accuracy, we find in the first place the work presented in [38] and other papers by the same authors. In this work, real and very detailed town maps are used, and radio propagation is modeled using a raytracing approach with some limited preprocessing. Node movement patterns are based on models derived from diverse research areas including urban planning, meeting analysis and time use. The authors study the use of a static ad hoc network running AODV in an area of central London. The highly accurate simulations take very long, in the order of tens of processor days. The conclusion is also here that high node density is a critical factor for good performance. Another study that uses a very detailed simulation model is the one described in [34]. The authors also use raytracing, and apply a preprocessing step that is based on using a discrete set of transmitter locations but an unlimited number of receiver locations. Their simulations are reasonably efficient: they are only about a factor 1.5 slower than comparable open space simulations. The authors compare the performance of AODV in the urban scenario to that in an open space scenario, and observe a large drop in performance.

The work presented in this report is in approach and level of detail similar to the last described work. It allows to get a reasonable feel for what the effects can be of an urban environment, while making enough abstraction to get an efficient simulation.

6 Experimental evaluation

In this section we present the results of a simulation study in which we evaluate the use of AntHocNet in the urban setup outlined above. We first describe technical details about the setup of the study, and then investigate some general properties of the wireless network under the urban scenario as compared to an equivalent open space scenario, in order to give the reader a better understanding of the conditions in which these experiments are run. After that, we discuss a number of tests in which we compare AntHocNet to the AODV routing algorithm. AODV is an important reference in the field of MANET routing: it is the most studied algorithm around, and is one of the candidates for standardization by the MANET working group of the IETF [1]. We do tests with varying data send rates, varying numbers of communication sessions, varying network node densities, and varying node speeds. Finally, we also investigate in more detail whether it is possible to support voice communications with these routing algorithms in the given urban scenario.

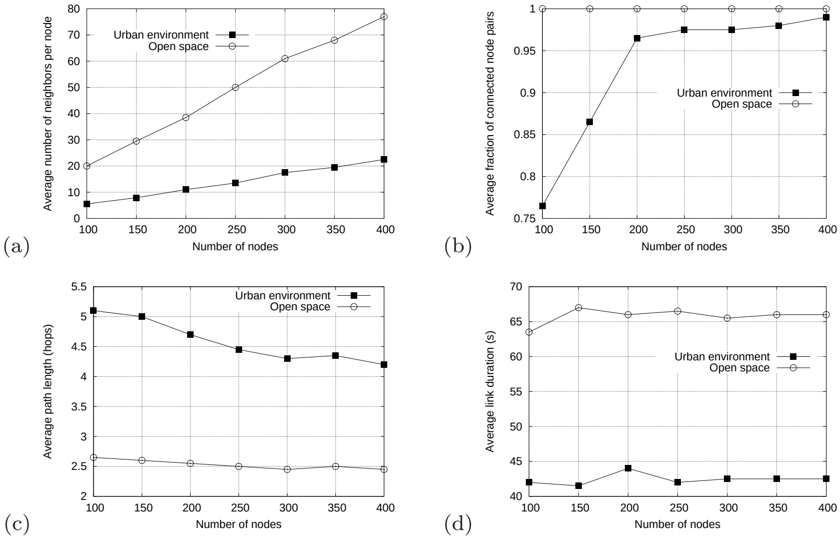
6.1 Technical details about the simulation setup

We run simulations of 500 s each, and do 20 individual runs with different random seeds for each data point. We normally use MANETs of 300 nodes, but also carry out experiments varying the number of nodes from 100 up to 400. All simulations are carried out using the *QualNet* network simulator [33]. Urban node movement patterns were fed to the simulator as mobility traces, and adaptations were made to the simulator code in order to load and use the precalculated radio wave propagation data (see section 5). For the network protocols situated at the different layers of the network protocol stack, we follow choices that are common in MANET research studies and make use of the implementations available in QualNet. At the physical layer we use the IEEE 802.11 protocol sending at a frequency of 2.4GHz and with a bit rate of 2Mbps. At the MAC layer, we use the IEEE 802.11 DCF protocol. Finally, at the transport layer, we use the UDP protocol.

6.2 General network properties

We study how the properties of the network formed between the MANET nodes are affected by the fact that we work in an urban environment. The data shown here were obtained by running simulations with an increasing number of nodes in both the urban scenario and an open space scenario of the same dimensions. During these simulations, no data traffic was sent, and all nodes were moving with a speed between 1 and 3 m/s. We report results for the *average number of neighbors* each node has, the *connectivity* (i.e., the fraction of node pairs between which a path exists), the *average length of the shortest path between each pair of nodes*, and the *average link duration*.

The average number of neighbors per node are shown in figure 2a. We can see that it is a lot lower in the urban scenario than in the open space scenario. This is due to the limited radio propagation caused by the shadowing by buildings of the town. This means that for an equal number of nodes per square meter, nodes in the urban scenario locally experience a lower density. As a consequence, connectivity is worse in the urban scenario, but radio interference is also lower. The lower connectivity in the urban scenario is confirmed by the results of figure 2b, where we report the fraction of node pairs between which a path exists. While the open space scenario is always fully connected, the urban scenario has limited connectivity when there are few nodes in the network. In figure 2c, we report the average number of hops on the shortest path between connected node pairs. Also this value is very much affected by the environment: paths are about double as long in the urban scenario. Finally, in figure 2d, we report the average duration between the appearance and disappearance of a link, which is a measure for how dynamic the network is [32]. The average link duration is independent of the number of nodes: it is constant on about 65 s in

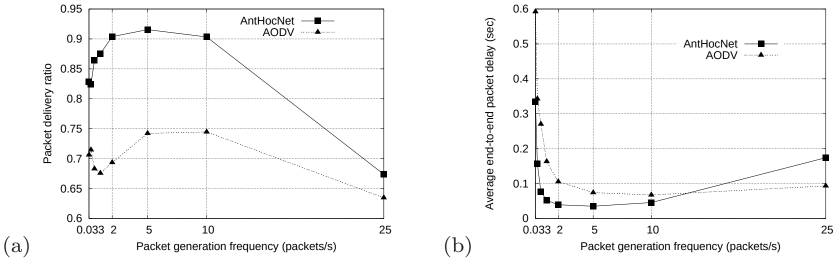


(a) (b) (c) (d)

Figure 2: Properties of MANETs with increasing number of nodes in the urban setting and in an open space environment. We report averages for (a) the number of neighbors per node, (b) the fraction of node pairs between which a path exists, (c) the number of hops on the shortest path between nodes, and (d) the link duration.

open space, and 43 s in the urban scenario. This means that the change rate of the network is higher in the urban environment.

6.3 Data send rate



(a) (b)

Figure 3: Results for AntHocNet and AODV with increasing data send rates in the urban scenario: (a) delivery ratio and (b) average end-to-end delay.

We compare the performance of AntHocNet and AODV in the urban scenario with 300 nodes and with increasing data send rate. We use 10 parallel bidirectional data sessions of 0.033 packets/s (1 packet every 30 seconds) up to

25 packets/s, using as intermediate values 0.2, 0.5, 1, 2, 5 and 10 packets/s. As *performance metrics*, we use the *delivery ratio*, which is the fraction of successfully delivered data packets, and the *average end-to-end delay*, which measures the average time between data packet generation and delivery at the destination.

The results are shown in figures 3a (delivery ratio) and 3b (average delay). We can see that both algorithms have bad performance for both metrics at the lowest data rates, better performance at intermediate rates, and worse performance again at the highest rates. The bad performance at the lowest rates is due to the fact that both AntHocNet and AODV need to set up a route between source and destination prior to communication. When data packets are sent sporadically, previously constructed routes can rarely be reused, and new route setups are often necessary, creating excessive network load. As data rates increase, subsequent packets can profit from previous route setups. In AntHocNet this effect is visible at lower rates than for AODV because the proactive route maintenance and improvement process keeps routes alive for longer periods. At the highest rates, both algorithms have a decrease in performance because the high load of data packets starts to interfere with the control packets. In general, we can see that AntHocNet outperforms AODV for most data rates (except for delay at the highest data rate). This shows that it is better equipped to deal with the specific properties of the urban scenario with its longer paths and less good connectivity. Nevertheless, it must be noted that neither of the algorithms is able to provide a delivery ratio that is sufficient to support a VoIP application (see also further in subsection 6.7).

6.4 Number of data sessions

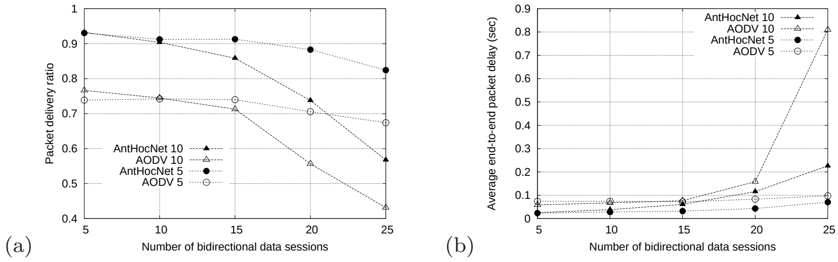
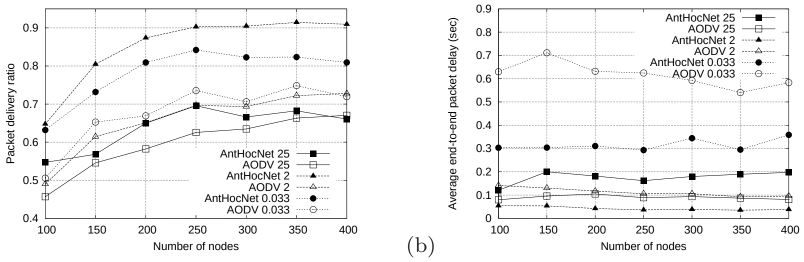


Figure 4: Results for AntHocNet and AODV with increasing number of data sessions in the urban scenario. We use data rates of 5 and 10 packets/s, indicated by different curves in the figure. Data plots show (a) delivery ratio and (b) average end-to-end delay.

Here, we report the results of tests in which we vary the number of simultaneous data sessions, from 5 up to 25 sessions, with as intermediate values 10, 15 and 20 sessions. We used two different values for the data send rate: 5 and 10 packets/s. In figure 4, we report delivery ratio and average end-to-end delay. Both performance metrics show similar patterns: performance is good when there are few data sessions, and deteriorates as the number of sessions increases. At higher data rates, this deterioration is much more pronounced, due to the higher radio interference caused by the increased amount of traffic in the network. In general, we can see that AntHocNet outperforms AODV for both performance metrics in all the studied scenarios, showing its superior ability to deal with the difficult conditions of the urban scenario.

6.5 Node density



(a) (b)

Figure 5: Results for AntHocNet and AODV with increasing number of nodes in the urban scenario. We use data rates of 0.033, 2 and 25 packets/s, indicated by different curves in the figure. Data plots show (a) delivery ratio and (b) average end-to-end delay.

In these tests we increase the number of nodes from 100 to 400 with increments of 50. Since the network area size of our urban scenario is fixed, increasing the number of nodes comes down to increasing the node density. Scenarios with higher node density normally have better connectivity and shorter paths, but also more radio interference between nodes (see also subsection 6.2). We use 10 bidirectional data sessions, and do tests with three types of data load: low (0.033 packets/s), medium (2 packets/s) and high (25 packets/s). The results are shown in figure 5, where we again report delivery ratio and average end-to-end delay.

In terms of delivery ratio, we observe a fixed pattern for both algorithms and for each of the data rates: performance improves with increasing node density, up to 250 nodes, after which it stabilizes. The delivery ratio is best for low and medium data rates, and worse for the highest rate, confirming that it is difficult to support the data rates needed for VoIP traffic. AntHocNet outperforms AODV for all data rates and all node densities, except for the highest rate at the highest density, where performances are comparable. This shows that AntHocNet can deal better with the conditions of the urban scenario, until the point where both algorithms start suffering too much from the increased radio interference due to both the increased node density and the increased data load. It is also interesting to see that the trend followed by the delivery ratio graphs is the same as that of the pairwise connectivity in the urban scenario, shown in figure 2b: first steeply increasing and then stabilizing. This shows that connectivity is an important factor to get good performance in urban scenarios (this has also been observed by other authors, e.g. see [38]). In terms of end-to-end delay, we can see performances that are relatively stable with respect to the node density. AntHocNet outperforms AODV for low and medium data rates. For the highest data rate, AODV has better delay. At this rate, however, both algorithms deliver only a limited fraction of the sent data packets.

6.6 Node speed

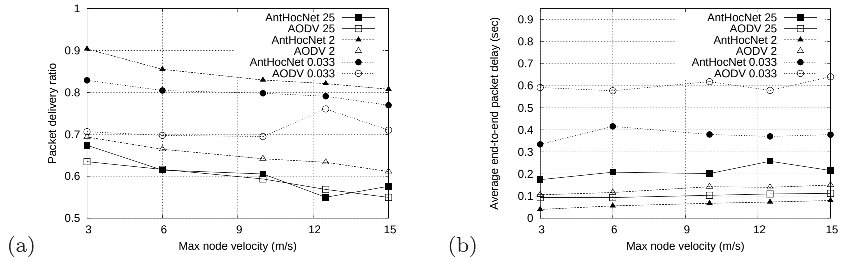


Figure 6: Results for AntHocNet and AODV with increasing maximum node speed in the urban scenario. We use data rates of 0.033, 2 and 25 packets/s, indicated by different curves in the figure. Data plots show (a) delivery ratio and (b) average end-to-end delay.

In these experiments we vary the maximum node speed: we increase it from the 3 m/s, which correspond to a slow cycling speed, up to 15 m/s, which is a reasonable maximum speed for cars in an urban environment. As intermediate values we use 6, 10 and 12.5 m/s. We keep using 10 bidirectional data sessions and again do tests with three types of data load: low (0.033 packets/s), medium (2 packets/s) and high (25 packets/s). In figure 6, we report the delivery ratio and the average end-to-end delay. As can be expected, delivery ratios go down with increasing node speeds under all data send rates for both algorithms. Apart from that, we get a confirmation of earlier results. Delivery ratios are highest for medium and low data rates, and lowest for the highest data rate. AntHocNet delivers more packets than AODV, except for the highest data rate, where delivery ratios are comparable. In terms of delay, we get a similar picture. Delay goes up with increasing nodes speeds, although the effect is minimal. The best delays are obtained for the low and medium data rates, and the worst for the highest rates. AntHocNet outperforms AODV at low and medium data rates, while AODV is the best at the highest data rate.

6.7 Supporting VoIP traffic

In a final set of experiments, we investigate the issue of delivering sessions with VoIP level data rates. In subsection 6.3, we saw that both algorithms have trouble when data rates go up to the high levels needed to support voice

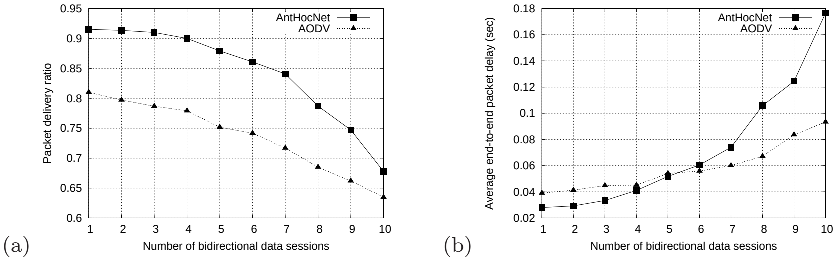


Figure 7: Results for AntHocNet and AODV with VoIP level data rates (25 packets/s) and varying numbers of data sessions in the urban scenario: (a) delivery ratio and (b) average end-to-end delay.

communication: AntHocNet drops to a delivery ratio of 67% and a delay of 0.19 s, while AODV has a delivery of 63% and a delay of 0.1 s. To support good quality VoIP, a delivery ratio of 90% is needed, while the end-to-end delay should not exceed 0.15 s (see [26]). These combined standards are not met by either of the algorithms. Here, we investigate this in more detail.

We do tests varying the number of data sessions from 1 to 10. All sessions are bi-directional and send 25 packets/s. The results are show in figure 7. When few sessions are started, both algorithms obtain an end-to-end delay that is well inside the requirements for VoIP, while AntHocNet also reaches the requirements for delivery ratio. Then, when the number of sessions is increased, both algorithms fail to reach the combined VoIP requirements. These reported results are averages over all the started communication sessions though. In order to get a more precise view, we investigate for each scenario how many of the individual sessions reach the cited requirements. These results are shown in figure 8. Using AntHocNet, a considerable number of sessions could obtain VoIP quality. This number first grows as more sessions are started up, and then drops again as data packets from too many sessions start to interfere with each other. For AODV, the number of sessions receiving the required service quality always remains low.

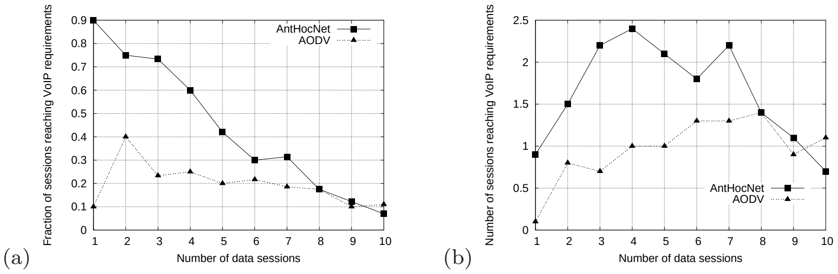


Figure 8: Sessions reaching VoIP quality requirements in terms of delivery ratio and delay: (a) as a fraction of the total number of started sessions and (b) in absolute numbers.

The results show that using AntHocNet it is in principle possible to support VoIP in the given urban scenario, despite the fact that on average it does less good at these data rates than AODV. However, not all sessions can get the required levels of service, and when too many sessions are started, all of them suffer. This indicates that it might be useful to refuse some sessions to start (e.g., those that would likely not receive the required quality because they need paths that go through highly congested or poorly connected parts of the network) in order to be able to deliver a good service to others. This points to the importance of the use of *admission control* in resource limited urban MANETs. For existing work on admission control in MANETs, see e.g. [13].

7 Conclusions

In this report, we have presented an application of ACO to the highly dynamic on-line optimization problem of routing in MANETs in a realistic urban environment. We have first given a short introduction to the area of MANET routing in general. Then we have explained the ideas behind the application of ACO to routing. We have highlighted the mechanisms that make this approach unique, and have discussed their advantages and disadvantages compared to more traditional routing algorithms, especially with respect to the specific conditions found in MANETs. After that, we have presented AntHocNet, a hybrid routing algorithm for MANETs that is based on ACO routing. AntHocNet is a concrete example of how ACO routing techniques can be adapted to work in highly challenging environments. Concretely, the algorithm combines ACO routing with other approaches to learning in order to get adaptivity and robustness while maintaining an efficient working. Next, we have described the urban environment we work in, and we have explained how it influences the conditions of the MANET deployment and its differences with respect to open space scenarios. We have also discussed how we modeled the MANET behavior in the urban environment in terms of radio propagation and mobility patterns, and how we simulated it in an efficient way. Finally, we have presented a number of test results in which we compared the performance of AntHocNet in the given urban MANET scenario to that of AODV, a state-of-the-art algorithm in this area, considering realistic models for interactive communications and different configurations for node density, speed, and traffic loads. The results show that AntHocNet is better able to deal with the difficult conditions that are found in the urban MANET. The only exception is the case with very high data rates, corresponding to the traffic generated by voice communications, where both algorithms incur high packet losses due to the increased radio interference, and AntHocNet suffers a higher average delay. However, when looking at individual sessions in these same scenarios, AntHocNet is more often than AODV able to offer the level of service required to support good voice communication.

In this report, we describe the application of Ant Colony Optimization (ACO) [14, 15] to a dynamic on-line optimization problem, namely *routing in mobile ad hoc networks in urban environments*. Routing is the task of finding and using paths to direct data flows through a network while optimizing one or more performance measures. This often comes down to a problem of finding minimum cost paths between pairs of source and destination nodes in the network. Hence, the problem of routing maps rather well to the solution model most commonly used in ACO, which is inspired by the ability of certain types of ants in nature to find the shortest path between their nest and a food source through a distributed process based on stigmergic communication [7]. An important aspect of routing, which sets it apart from many other applications of ACO, is that it is typically a distributed and dynamic problem, which means that the description of the problem changes over time and decentralized solutions must be adopted. This is because the situation in the network changes, e.g. because the traffic process at the nodes varies, or because there are link or node failures. As a consequence, the optimization algorithm for routing needs to adapt continuously.

Here, we focus on routing in a specific type of communication networks, namely *mobile ad hoc networks* (MANETs) [31]. These are networks that consist entirely of wireless nodes, placed together an ad hoc manner (i.e., on-the-fly, or with minimal prior planning) and without the support of a fixed communication infrastructure. All nodes are mobile, and can enter or leave the network at any time. Data are forwarded among the nodes of the network in multi-hop fashion. An example could be a network created by users carrying WiFi enabled laptops or palmtops operating in ad hoc mode, or a network created among moving cars that are enabled with wireless technology (in this case we also talk of vehicular networks [24]). MANETs are substantially different from more traditional wired communication networks such as the Internet. They are highly dynamic, have severe restrictions on the effective usable bandwidth (mainly due to the sharing of the wireless medium) (e.g., [18, 23]), have limited battery power available at each node, are based on the use of possibly unreliable wireless communication channels, etc. Algorithms and protocols for MANETs should be adapted to deal with these challenging properties. In this report we show how techniques from ACO can be applied to support routing in this kind of networks. We focus in particular on MANETs deployed in urban environments, which are confronted with specific conditions in terms of the network node movement patterns and the wireless radio propagation.

In the rest of this report we first provide some more background on MANET routing. Then, we explain the main ideas behind existing work on ACO for routing. Next, we present the *AntHocNet* routing algorithm, which applies ACO routing in MANETs. After that, we describe the properties of the urban scenarios used in our studies, and finally we present the results of a set of experiments in which we study the general characteristics of urban MANETs, and we evaluate the performance of the AntHocNet routing algorithm in these scenarios compared to AODV [29], a state-of-the-art MANET algorithm.

References

[1] The internet engineering task force mobile ad-hoc networking page (MANET). Available from: `http://www.ietf.org/html.charters/manet-charter.html`.

[2] Taipei-wlan. Available from: `http://wlan.taipei-elife.net/english/main.html`.

[3] Wireless philadelphia. Available from: `http://www.wirelessphiladelphia.org`.

[4] M. Abolhasan, T. Wysocki, and E. Dutkiewicz. A review of routing protocols for mobile ad hoc networks. *Ad Hoc Networks*, 2:1–22, 2004.

[5] AWE Communications. *WinProp software suite.*

[6] D. Bertsekas and R. Gallager. *Data Networks.* Prentice–Hall, Englewood Cliffs, NJ, USA, 1992.

[7] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems.* Oxford University Press, 1999.

[8] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, 1998.

[9] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, A. Qayyum, and L. Viennot. Optimized link state routing protocol. In *Proceedings of IEEE INMIC*, 2001.

[10] S. R. Das, C. E. Perkins, and E. M. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, March 2000.

[11] G. Di Caro and M. Dorigo. AntNet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research (JAIR)*, 9:317–365, 1998.

[12] 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 (ETT)*, 16(5), 2005.

[13] Y. Dong, D. Makrakis, and T. Sullivan. Effective admission control in multihop mobile ad hoc networks. In *Proc. of the International Conference on Communication Technology (ICCT)*, 2003.

[14] M. Dorigo, G. Di Caro, and L. M. Gambardella. Ant algorithms for distributed discrete optimization. *Artificial Life*, 5(2):137–172, 1999.

[15] M. Dorigo and T. Stützle. *Ant Colony Optimization.* MIT Press, Cambridge, MA, 2004.

[16] F. Ducatelle. *Adaptive Routing in Ad Hoc Wireless Multi-hop Networks.* PhD thesis, Università della Svizzera Italiana (USI), Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), 2007.

[17] F. Ducatelle, G. Di Caro, and L. M. Gambardella. Using ant agents to combine reactive and proactive strategies for routing in mobile ad hoc networks. *International Journal of Computational Intelligence and Applications (IJCIA)*, 5(2), 2005.

[18] P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, March 2000.

[19] E. Huang, W. Hu, J. Crowcroft, and I. Wassell. Towards commercial mobile ad hoc network applications: A radio dispatch system. In *Proceedings of the sixth ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, May 2005.

[20] A. Jardosh, E. M. Belding-Royer, K. C. Almeroth, and S. Suri. Towards realistic mobility models for mobile ad hoc networks. In *Proceedings of MobiCom*, 2003.

[21] W. Jiang and H. Schulzrinne. Analysis of on-off patterns in voip and their effect on voicetraffic aggregation. In *Proceedings of the Ninth International Conference on Computer Communications and Networks*, 2000.

[22] D. B. Johnson and D. A. Maltz. *Mobile Computing*, chapter Dynamic Source Routing in Ad Hoc Wireless Networks. Kluwer, 1996.

[23] J. Jun, P. Peddabachagari, and M.L. Sichitiu. Theoretical maximum throughput of IEEE 802.11 and its applications. In *Proceedings of the 2nd IEEE International Symposium on Network Computing and Applications (NCA)*, April 2003.

[24] J. Luo and J.-P. Hubaux. *Embedded Security in Cars*, chapter A Survey of Research in Inter-Vehicle Communications, pages 111–122. Springer Berlin Heidelberg, 2006.

[25] S. Marinoni and H. H. Kari. Ad hoc routing protocol performance in a realistic environment. In *Proceedings of IEEE ICN*, April 2006.

[26] Ian Marsh, Fengyi Li, and Gunnar Karlsson. Wide area measurements of voip quality. In *Proceedings of the 4th International Workshop on Quality of Future Internet Services*, 2003.

[27] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization.* Prentice-Hall, New Jersey, 1982.

[28] C. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, pages 234–244, 1994.

[29] C. E. Perkins and E. M. Royer. Ad-hoc on-demand distance vector routing. In *Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999.

[30] V. Ramasubramanian, Z. J. Haas, and E. G. Sirer. Sharp: A hybrid adaptive routing protocol for mobile ad hoc networks. In *Proceedings of The Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2003.

[31] E. M. Royer and C.-K. Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, 1999.

[32] N. Sadagopan, F. Bai, B. Krishnamachari, and A. Helmy. PATHS: analysis of PATH duration statistics and their impact on reactive MANET routing protocols. In *Proceedings of MobiHoc'03*, pages 245–256, 2003.

[33] Scalable Network Technologies, Inc. *QualNet Simulator, Version 3.8*, 2005. Available from: `http://www.scalable-networks.com`.

[34] A. Schmitz and M. Wenig. The effect of the radio wave propagation model in mobile ad hoc networks. In *Proceedings of ACM MSWiM*, October 2006.

[35] R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz. Ant-based load balancing in telecommunications networks. *Adaptive Behavior*, 5(2):169–207, 1996.

[36] P. Shirley and R. Morley Keith. *Realistic Ray Tracing.* A.K. Peters, 2001.

[37] V. Sridhana and S. Bohacek. Realistic propagation simulation of urban mesh networks. Technical report, University of Delaware, Department of Electrical and Computer Engineering, 2006.

[38] V. Sridhara, J. Kim, and S. Bohacek. Performance of urban mesh networks. In *Proceedings of ACM MSWiM*, October 2005.

[39] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction.* MIT Press, 1998.

[40] S. Tadrus and L. Bai. A QoS network routing algorithm using multiple pheromone tables. In *Proceedings of the IEEE/WIC International Conference on Web Intelligence*, pages 132–138, Halifax, Canada, October 2003.