Westfälische
Wilhelms-Universität
Münster

Communication and
Networked Systems

Towards Energy-Aware Ant Routing in Wireless Multi-Hop Networks

Michael Frey, Mesut Günes, Friedrich Grosse
frey@informatik.hu-berlin.de, mesut.guenes@uni-muenster.de,
friedrich.grosse@fu-berlin.de

Technical Report

12/2014

Communication and Networked Systems (ComSys)
Institute of Computer Science
University of Münster

Contents

List of Figures

Acronyms

ACLR ant colony optimization based location aware routing for wireless sensor networks. 28, 30

ACO ant colony optimization. 5, 18, 30

AHP analytic hierarchy process. 30

ARA ant routing algorithm. i, 4, 8–15, 18, 20, 21, 23, 26, 32

ARAMA ant aouting algorithm for mobile ad hoc networks. 28, 30

BANT backward ant agent. 8, 9, 12–14, 20–22, 29

CC communication complexity. 14

EARA energy-aware ant routing algorithm. i, 4, 15, 18, 20, 21, 23, 26, 31, 32

EDS energy dead series. 23, 25

EEABR energy-efficient ant-based routing algorithm. 29, 30

ETX expected transmission count. 30

FANT forward ant agent. 8, 9, 12–14, 20, 21, 28

MANET mobile ad-hoc network. i, 2, 10, 31

MCP multi-constrained path problem. 29–31

MD minimum delay. 30

NIC network interface card. 16

NP nondeterministic polynomial time. 30, 31

OLSR optimized link state routing protocol. 30, 31

PDR packet delivery rate. 23, 24

PEANT periodic energy ant agents. 18, 19, 21, 24

QOLSR QoS-enhanced OLSR. 30, 31

QoS Quality of Service. 29–31

RSP restricted shortest path. 29, 30

TC time complexity. 14

WMHN wireless multi-hop network. i, 2–4, 29–31, 33

WMN wireless mesh network. i, 2, 10, 32

WSN wireless sensor network. i, 2, 10, 29–32

Abstract

A wireless multi-hop network (WMHN) is particularly described by two properties: i) full radio networks and ii) multi-hop communication. There are several instances of WMHNs in literature like wireless mesh networks (WMNs), wireless sensor networks (WSNs), and mobile ad-hoc networks (MANETs). All these types of WMHNs share the aforementioned properties and thus researchers can study them in similar ways. In this study, we address two key issues in these networks: routing and energy awareness in the routing.

For this we describe ant routing algorithm (ARA), a highly adaptive, efficient, and scalable routing protocol based on ant algorithms, which are a class of swarm intelligence algorithms. Furthermore, we discuss the integration of energy awareness in ARA and describe the resulting energy-aware ant routing algorithm (EARA). The goal of EARA is to maximize the network lifetime which is generally restricted by the limited energy available to nodes.

CHAPTER 1

Introduction

1.1 Wireless Multi-Hop Networks

Wireless multi-hop networks (WMHNs) are entirely wireless networks that do not require any infrastructure. In these networks the most important characteristic is the cooperative “forwarding” operation of all nodes to enable network wide communication. The global result of this cooperative forwarding is the *routing* service of the network. By this cooperation, nodes which are far away from each other can communicate, since intermediate nodes forward packets. The intermediate nodes are nodes without any distinguishing properties than the others. Therefore, all nodes in a wireless multi-hop network share both roles in a network. They are hosts (end nodes) as well as infrastructure nodes fulfilling the tasks of routers and switches known from wired networks.

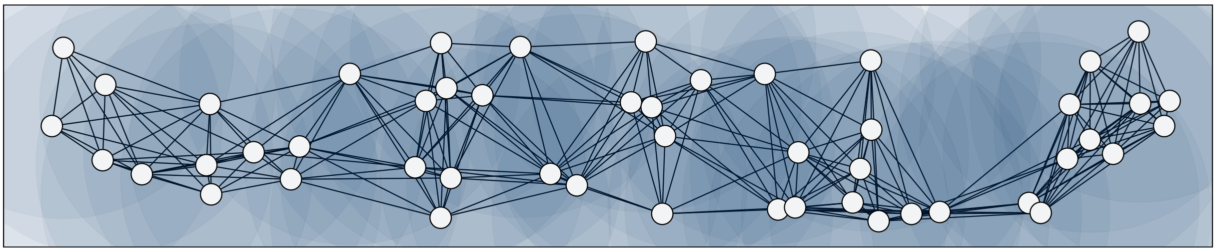
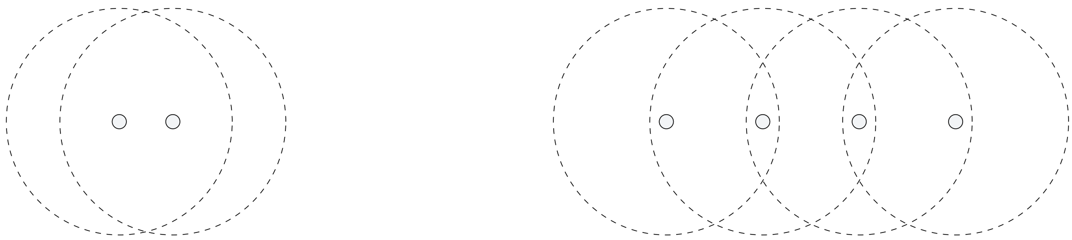


Figure 1.1: *A wireless multi-hop network consisting of a large number of nodes. The nodes function as end hosts as well as infrastructure nodes, i.e., forwarding packets for other nodes.*

The most prominent instances of WMHNs are wireless mesh networks (WMNs), wireless sensor networks (WSNs), and mobile ad-hoc networks (MANETs). This family of networks is flexible, since it does not require a lot of pre-planning. However, this flexibility adds new complexity into the network and its management, which results in more complicated protocols. Unfortunately, the nodes in these networks are usually battery operated and thus contain limited resources in terms of computation, memory, etc. Figure 1.1 shows a large wireless multi-hop network. In real applications a WMHN may consists of hundreds or thousands of nodes.



(a) *Single-Hop Communication* **(b)** *Multi-Hop Communication*

Figure 1.2: *Communication in a wireless multi-hop network. A source and destination node can communicate in two ways: (a) If they are in their mutual radio range. (b) There exists a chain of neighbors, that interact as intermediate forwarders.*

1.2 Communication in WMHNs

We consider the communication between a *source* and a *destination* node. The transmission of packets depends on the used technology, but it is not really important for our consideration. We assume that a node transmits packets via suitable MAC and PHY layers. Furthermore, two nodes can communicate with each other if they are mutually in their radio ranges (Figure 1.2). In this case we call the nodes as *neighbor* and assume a wireless link between them. A pair of source and destination nodes can communicate with each other in two ways: i) source and destination nodes are neighbors, ii) there exist a chain of neighboring nodes between the source and destination node. We denote the latter communication as *multi-hop communication.* Starting from a source each node forwards a packet hop-by-hop until it reaches its destination.

1.3 Open Research Problems in WMHNs

Wireless multi-hop networks introduce new problems and challenges. The problems are on all layers of the ISO/OSI communication model. On the network layer routing and addressing are particularly affected [15, 17]. The focus of this study is on routing and energy awareness of routing in WMHNs.

The task of the routing is to provide "cost effective" routes among pairs of source and destination nodes. A *routing metric* allows to grasp the notion of cost effectiveness on a technical level. It allows the comparison of different routes and hence supports the "forwarding process" in the selection of routes. There are plenty of routing metrics studied for classical networks as well as for WMHNs [27]. The most used routing metric in WMHNs is the *shortest path*, since it is simple to implement and supports the comparison of different routing protocols if other properties are neglected [8]. There is a broad discussion about this issue, whether the shortest path routing metric yields acceptable results [1].

Unfortunately, the shortest path routing does not take a important resource of WMHNs into consideration, the available energy. It may happen that an important node is overloaded by the forwarding process and thus depletes its energy reserves fully and thus cannot be part of the network anymore. In the worst case this could result in the splitting of the network into unconnected sub-networks.

Since most of the nodes in WMHNs will be battery operated the available energy of the nodes and thus the lifetime of a node is almost defined by its battery capacity. One can extend the lifetime of a node by means of a higher battery capacity or through better usage of the available energy. The latter one is more important, since nodes in a WMHN have to fulfill the service of the lacking infrastructure, i.e., they have to participate in the forwarding process.

1.4 Contribution

The contribution of this paper is energy-aware ant routing algorithm (EARA) which is an energy aware extension of the ant routing algorithm (ARA). ARA is an on-demand routing algorithm based on swarm intelligence, that adapts the ant foraging method to route finding in WMHNs. We presented ARA with promising results in [15, 16, 18]. In these studies the authors consider a single routing metric, the shortest path metric.

Although ARA suggests the inclusion of additional routing metrics, we are not aware of any studies which take more than one routing metric into consideration. In this paper we study a routing with two routing metrics: shortest hop and the estimated energy fitness of a path.

1.5 Structure of the Paper

The remainder of this text is organized as follows. In chapter 2 we present the basics of ant algorithms and discuss briefly swarm intelligence. Subsequently, in chapter 3 we describe the routing algorithm ARA in detail. chapter 4 discusses the integration of the energy awareness in ARA and presents EARA. In chapter 5 we present an overview about the related work and discuss adaptation of ant algorithms for routing and constrained-based routing. Finally, we give a short a summary, some conclusions, and planned future work in chapter 6.

CHAPTER 2

Ant Algorithms

Ant algorithms are swarm intelligence algorithms which are based on the behavior of ants and colonies. Swarm intelligence algorithms solve complex tasks by cooperation. Ant algorithms were first proposed as ant colony optimization (ACO) metaheuristic in [3, 7] and are multi-agent systems where agents show the behavior of individual ants. However, an individual ant is not aware of the problem to be solved, instead each ant behaves independently, but in cooperation with all other ants in the swarm. The solution to the problem *emerges* by this cooperation.

2.1 Ant Foraging Behavior

The basic principle of the ant algorithm is inspired by the foraging behavior of ants. While ants search for food, they start from their nest and seek randomly for food. Ants deposit a special hormone called *pheromones* on their way towards food. In addition, ants are attracted by pheromones and in turn reinforce them. The concentration of pheromones on a certain path is an indication of its usage. Over time the concentration of pheromone decreases due to diffusion effects.

Figure 2.1 shows a scenario with two routes from the nest to the food. At the intersection the first ants randomly select a branch (see Figure 2.1a)). Since the lower route is shorter than the upper one, the ants which take this path will reach the food place first. On their way back to the nest, the ants again have to select a path. After a while the pheromone concentration on the shorter path will be higher than on the longer path, since the ants using the shorter path will increase the pheromone concentration faster (see Figure 2.1b)). Thus, eventually all ants will only use this path (see Figure 2.1c)).

What happens if there exist a close dead end without food between the nest and the food place? One might assume the ants would amplify the path to the dead end because of their random behavior. This is prohibited, since ants with food deposit more pheromone as ants without food. Hence, ants amplify routes leading to food much more stronger.

The foraging behavior of the ants can be used to find the shortest path in networks. Especially the dynamic component of this approach allows a high degree of adaptation to changes in the topology of wireless multi-hop networks.

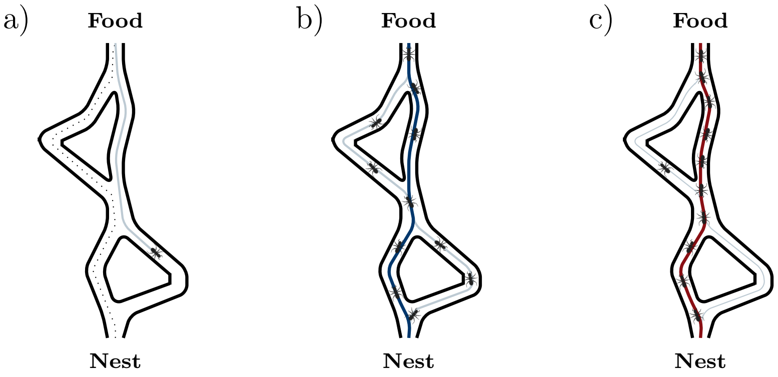


Figure 2.1: *Model of the ant foraging behavior in a scenario with two possible routes between the nest and the food. a) The ants start from the nest to seek for food by random search. b) Both routes are found by the ants. The pheromone concentration differs on both routes. c) Eventually all ants take the shortest route after an initial searching time.*

2.2 A Simple Ant Algorithm

Let $G = (V, E)$ be a connected graph with $n = |V|$ nodes. The simple ant colony optimization meta-heuristic can be used to find the shortest path between a source node s and a destination node d on the graph G. The path length is given by the number of nodes on the path. Each edge $e(i, j) \in E$ has a set of variables $\varphi^i_{d,j}$, called artificial pheromone, which are modified by the ants when they visit the node i on their travel to the destination d and are relayed to j. The pheromone concentration $\varphi^i_{d,j}$ is an indication of the usage of the edge $e(i, j)$. Initially $\varphi^i_{d,j}$ is constant for each edge $e(i, j)$. An ant located in node i uses the pheromone $\varphi^i_{d,j}$ of node $j \in N_i$ to compute the probability of node j being the next hop. N_i is the set of one-step neighbors of node i. The transition probabilities $p^i_{d,j}$ of a node i, i.e., the probability that the ant selects node j after it has visited i, are defined as follows

$$p^i_{d,j} = \begin{cases} \frac{\varphi^i_{d,j}}{\sum_{k \in N_i} \varphi^i_{d,k}} & \text{if } j \in N_i \\ \\ 0 & \text{if } j \notin N_i \end{cases} \tag{2.1}$$

and the normalization condition

$$\sum_{j \in N_i} p^i_{d,j} = 1.$$

During the route finding process, ants deposit pheromone on the edges. In the simplest version of the algorithm, the ants deposit a constant amount $\Delta\varphi$ of pheromone, i.e., the amount of pheromone of the edge $e(i, j)$ when the ant is moving from node i to node j is changed as follows:

$$\varphi^i_{d,j} \leftarrow \varphi^i_{d,j} + \Delta\varphi$$

Like real pheromone the artificial pheromone concentration decreases with time. In the simple ant algorithm the pheromone concentration after τ time units is given by:

$$\varphi^i_{d,j}(t + \tau) := (1 - q) \cdot \varphi^i_{d,j}(t), \qquad q \in (0, 1] \tag{2.2}$$

2.3 Extensibility of the Approach

The simple ant algorithm can be extended to include other parameters like energy or bandwidth into the forwarding process. For this we extend Equation 2.1 and obtain Equation 2.3. We describe the additional parameter by γ. The free selectable parameters α and β allow the weighting of the parameters φ and γ, which will represent routing metrics in our case.

$$p^i_{d,j} = \begin{cases} \frac{[\varphi^i_{d,j}]^\alpha [\gamma]^\beta}{\sum_{k \in N_i} [\varphi^i_{d,k}]^\alpha [\gamma]^\beta} & \text{if } j \in N_i \\ \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

2.4 Why ant algorithms are suitable for wireless multi-hop networks

The simple ant algorithm introduced in the previous section illustrates different reasons why this kind of algorithms could perform well in wireless multi-hop networks. We discuss some by relating them to important properties of wireless multi-hop networks.

- **Dynamic Topology:** This property is responsible for the poor performance of "classical" routing algorithms in wireless multi-hop networks. The ant algorithm is based on autonomous agent systems imitating individual ants. This allows a high adaptation to the current topology of the network.

- **Local Work:** In contrast to other routing approaches, the ant algorithm relies only on local information, i.e., the algorithm does not need to share routing tables or other information blocks with other nodes.

- **Link Quality:** It is possible to integrate the connection/link quality into the computation of the pheromone concentration, especially into the evaporation process. This will improve the decision process with respect to the link quality. It is important to note that the approach can manipulate the pheromone concentration independent of the ants, e.g., if a node detects a change of the link quality.

- **Support for Multi-Path Routing:** Each node has a routing table with entries for all its neighbors which also contains the pheromone concentration. A node selects the next hop on basis of the pheromone concentration. Thus, the approach supports multi-path routing.

- **Multi Route Metric Routing:** The original ant algorithm supports the inclusion of additional parameters. Hence, the algorithm can consider more than one routing metric in the forwarding process. In chapter 4 we will study a variant with two routing metrics.

CHAPTER 3

The Ant Routing Algorithm

In this section we discuss the adaptation of the simple ant algorithm for wireless multi-hop networks. The result is the *Ant Routing Algorithm* (ARA). We described it first in [18] and with some extensions in [16], and finally in more detail in [15]. Our discussion in this text is mainly based on the latter two, however with a focus to wireless multi-hop networks. ARA was initially developed for mobile ad-hoc networks (MANET), but it is also suitable for wireless sensor networks (WSN), and wireless mesh networks (WMN). Thus, we consider in this text also the adaptation of the approach for these networks and discuss also the complexity of the approach, which may help to judge the suitability to these networks.

3.1 The Three Phases of Routing

We distinguish three phase of operation of a routing protocol for wireless multi-hop networks: i) route discovery, ii) route maintenance, and iii) route failure handling. Therefore, ARA consists, as other routing protocols, of three phases, which we discuss next.

3.1.1 Route Discovery

ARA setups new routes in the route discovery phase. The algorithms discovers new routes by the use of forward ant agents (FANTs) and backward ant agents (BANTs). A FANT is an agent which establishes the pheromone track in the network back to the source node. In analogous, a BANT establishes the pheromone track back to its origin, namely the destination node. In the protocol implementation the FANT is a small packet with a unique sequence number. Nodes are able to distinguish duplicate packets on the basis of the sequence number and the source address.

A node which receives a FANT for the first time creates a record in its routing table. An entry in the routing table is a triple `(destination address, next hop, pheromone value)`. The node interprets the source address of the FANT as destination address, the address of the previous node as the next hop, and computes the pheromone value. Subsequently, the node relays the FANT to its neighbors. ARA identifies duplicate FANTs by the unique sequence number, and removes them. Eventually, the destination node extracts the information of the FANT, creates a BANT and returns it to the source node. The BANT's task is similar to that of the FANT, i.e., to establish a pheromone track to

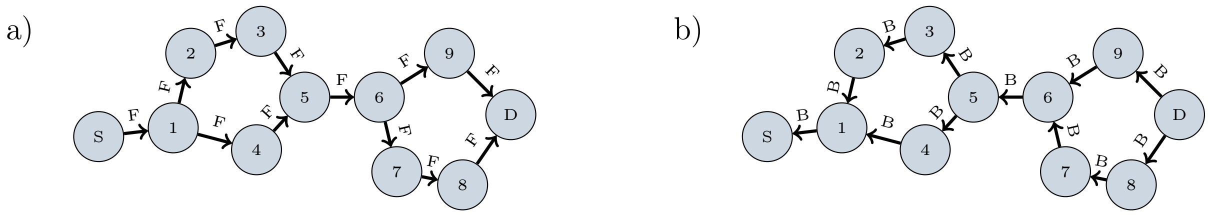


Figure 3.1: *Route discovery phase in ARA. a) A source node* S *sends a forward ant (F) to the destination node* D*. Other nodes relay the forward ant and initialize their routing table and the pheromone values. b) The backward ant (B) has the same task as the forward ant. The destination node* D *sends the backward ant to the source node* S*.*

this node in the network. When the source node receives the BANT from the destination node, the path is established and data packets can be sent.

Figure 3.1 demonstrates the route discovery phase of ARA. Figure 3.1 a) shows the establishment of the pheromone track back to the source node S. The forward ant only creates one pheromone track to the source node in node 6, but two tracks in node 5, via node 3 and node 4. Figure 3.1 b) depicts analogous situation for the backward ant. It only creates one pheromone track to the destination node D in node 5 and two tracks in node 6. Thus, multi-path routing is also supported by ARA.

3.1.2 Route Maintenance

The second phase of ARA is route maintenance. This phase is responsible for the maintenance of the routes during communication. ARA does not need any special packets for that purpose. Once the FANT and BANT established the pheromone tracks for the source and destination nodes ARA uses regular data packets to maintain the path. As in biological systems, established paths do not keep their initial pheromone values forever. When a node i forwards a data packet to destination d to a neighbor node j, it increases the pheromone value of the entry $(d,\ j,\ \varphi)$ by $\Delta\varphi$, i.e., ARA reinforces this path to the destination by the data packet. Likewise, the next hop j increases the pheromone value of the entry $(s,\ i,\ \varphi)$ by $\Delta\varphi$, i.e., the backward path to the source node is also reinforced. We model the evaporation process of the real pheromone by decreasing the pheromone values according to Equation 2.2.

This method for route maintenance could lead to undesired loops. ARA prevents loops by a simple method which is also used during the route discovery phase. Nodes can detect duplicates of data packets, based on the source address and the sequence number. If a node receives a duplicate packet, it will set a special flag and send the packet back to the previous node. The previous node deactivates the link to this node, i.e., the entry in the routing table, thus data packets are not sent to this direction any more.

3.1.3 Route Failure Handling

The third and last phase of ARA handles routing failures which are especially caused by network topology changes, which are common in wireless multi-hop networks. Another important reason is node mobility, which is particularly common in mobile ad-hoc networks. ARA assumes that the MAC layer notifies the routing layer when a packet to a neighbor could not be transmitted. This enables ARA to recognize a route failure through

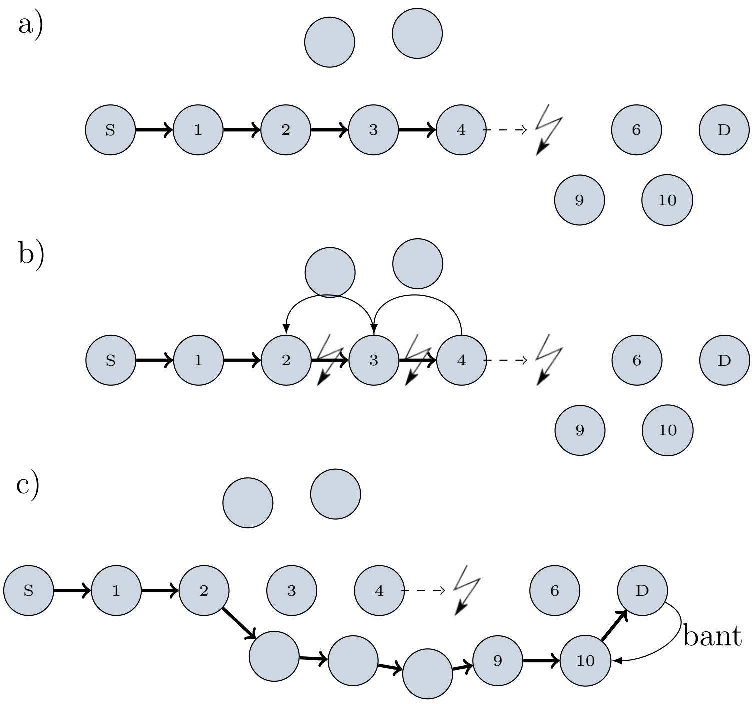


Figure 3.2: *Route failure handling of ARA*

a missing acknowledgment on the MAC layer. In the case without MAC layer notification ARA has to ensure the connectivity of the (mobile) nodes by itself to be able to detect link failures. One approach to detect link failures are hello-messages.

Figure 3.2 shows an example of the route failure handling of ARA. The node which detects a routing failure, which in fact is a link failure, first deactivates this link in its routing table. Subsequently, the node searches for an alternative link in its routing table. If there is another route to the destination it will send the packet via this path. If there are entries in the routing table, the node will not try all instead only one trial is allowed. In the case that the node does not have any other paths to the destination, it informs its neighbors about the event. The neighbors delete the paths destined to that certain destination over this node. The neighbors can transport that data packet over an alternative path if they know one, but only one trial is allowed. If they do not have an alternative path they relay the data packet into the vicinity of the source node. Eventually, a path to the destination will be find or the source node recognizes its data packet and initiates a new route discovery phase. The relaying of the data packet toward the source node is restricted by the distance which it has already traveled towards the failure position.

3.1.4 Summary and Discussion

Macker et. al. discuss in [24] some requirements for routing algorithms for MANETs. To some degree these requirements are also important for the other types of wireless multi-hop networks like WSN and WMN. The mentioned requirements are the following:

- **Distributed Operation:** In ARA, each node owns a set of pheromone counters $\varphi_{i,j}$ in its routing table for a link between node ν_i and ν_j. Each node controls the pheromone counter independently when ants visit the node on route search for, or when the node detects a link failure.

- **Loop-free:** The use of unique sequence numbers of route finding packets avoids loops.
- **Demand-based Operation:** Routes are established by manipulating the pheromone counters $\varphi_{i,j}$. Over time, the amount of pheromone decreases to a minimum value when ants do not visit this node. A route finding process is only run upon demand by a sender.
- **Sleep Period Operation:** Nodes are able to sleep when the amount of pheromone in their routing table has reached a lower threshold. Other nodes will then not consider this node, unless packets are destined to it.

Additionally, ARA has the following properties:

- **Locality:** The routing table and the statistic information block of a node are local, and they are not transmitted to any other node.
- **Multi-path:** Each node can have multiple paths to a certain destination. The choice of a certain route depends on the environment, e.g., on the link quality to the relay node.
- **Sleep mode:** In sleep mode a node only processes packets if its a destination node for this packets. This saves energy and power.

3.2 Packet Forwarding

So far, we discussed the adaptation of the ant algorithm to wireless multi-hop networks and the protocol of ARA, but we did not describe how the forwarding is really done. The forwarding decision in each node is local and depends on the knowledge of the node without exchanging any special data. ARA supports two different modes of operation which we describe after the definition of the routing table structure.

3.2.1 Routing Table

Each node uses the pheromone values in its routing table as shortest path metric. The routing table of a node has the following structure:

t_{la}		
d_1	nh_1	φ_{d_1,nh_1}
d_1	nh_2	φ_{d_1,nh_2}
$\vdots$	$\vdots$	$\vdots$
d_k	nh_m	φ_{d_k,nh_m}

The entries in routing table have the following meanings. d_k is the address of the k-th destination, $nh_m \in N_i$ is the next hop for this destination, and φ_{d_k,nh_m} represents the pheromone value for that next hop. Additionally, the time of the last access to the routing table t_{la} is also stored. Notice that for a single destination d_k there can be multiple next hop entries with different pheromone values. By this ARA provides multi path routing.

3.2.2 Maximum Pheromone Routing

In this mode ARA tries to transport packets always on the best available route. At each intermediate node i on the path ARA selects the next hop j towards the destination d with the highest pheromone value. The forwarding decision is felt according to Equation 3.1. This policy reinforces good paths quickly and at the same time leads to a quick declining of unused paths.

$$p^i_{d,j} = \begin{cases} 1 & \text{if } \varphi^i_{d,j} = \max_{k \in N}\{\varphi^i_{d,k}\} \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

3.2.3 Probabilistic Routing

In the probabilistic mode an intermediate node i chooses node j as next hop depending on the pheromone value $\varphi^i_{d,j}$. The higher the pheromone value of a certain neighbor is, the higher the probability is that ARA selects this node. But, unlike to maximum pheromone routing it is not guaranteed that it is always chosen. We give the forwarding rule in Equation 3.2.

$$p^i_{d,j} = \begin{cases} \frac{\varphi^i_{d,j}}{\sum_{k \in N_i} \varphi^i_{d,k}} & \text{if } j \in N_i \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

When ARA runs in probabilistic routing mode, it works like a multi-path routing algorithm where an algorithm transports packets over multiple available paths between the source and destination nodes at the same time. For the packet transportation ARA uses optimal paths as well as less optimal paths.

The main advantage of the probabilistic routing mode is its robustness against routing failures, e.g., because of network topology changes. Unlike to maximum pheromone routing ARA reinforces – if available – multiple paths. Therefore, a node can better intercept a routing failure by using one of the other available paths to the destination node.

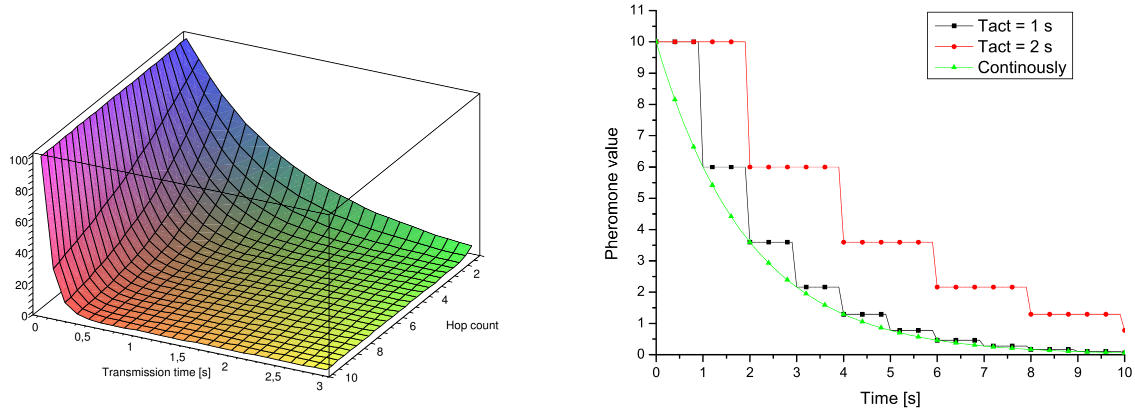
3.3 Pheromone Management

As presented in the previous section the forwarding decision is mainly based on the pheromone value. Thus, the maintenance of pheromone values impact the performance of the routing protocol. Obviously, the pheromone value maintenance consists of two stages. First, the initial pheromone value and second the maintenance of the pheromone value during data communications.

3.3.1 Initialization of the Pheromone Values

There are some candidate functions for the initial pheromone value computation.

- A constant c.
- The number of hops used to transmit the FANT/BANT from source to destination and vice versa.
- The transmission time of the FANT/BANT from source node to destination node and vice versa.



(a) *Initialization of pheromone value*

(b) *The development of the pheromone values*

Figure 3.3: *Pheromone value development. (a) The initialization of the pheromone value depending on the hop count and the transmission time of the FANT. (b) Discrete and continuous decreasing functions for the pheromone value.*

ARA deploys the function in Equation 3.3. The function considers the number of hops n and the transmission time t from source node to the destination node. Figure 3.3a depicts the initial pheromone value as a function of the number of hops and transmission time.

$$\varphi(n,t) = \frac{c}{e^{n \cdot t}} \tag{3.3}$$

3.3.2 Adaptation of the Pheromone Values

In the initial version of ARA [18] we simulated the decreasing of the pheromone values in intervals of a second. Unfortunately, it was shown that this was not appropriate since the alteration of two paths which differed only within an interval were forged. To better adapt the decreasing of the pheromone values to the natural behavior, ARA decreases the pheromone values continually. For this, a node stores the time of the last access to the routing table t_{la}. Before the node performs a routing decision, the node updates the pheromone values in its routing table according to Equation 3.4. Figure 3.3b shows the discrete and continuous updating of the pheromone values over the time.

$$\varphi^{i}_{d,j} = \varphi^{i}_{d,j}(t_{la}) \cdot e^{t_{la}-t} \tag{3.4}$$

3.4 Decreasing the Number of Packets for FANT/BANT

One approach to reduce the overhead ARA causes by the flooding of FANT and BANT in the network is the integration of techniques like Gossiping. We have integrated the technique described in [19].

3.5 Complexity of ARA

In this section we give some complexities of ARA. We follow the definition of complexities and the discussion as in [6,25]. We define the communication complexity (CC) as the number of messages exchanged in performing a protocol operation, and the time complexity (TC) as the number of steps to perform a protocol operation.

We define the communication complexity of the route discovery phase by the exchange of the FANTs and the BANTs, which ARA floods through the network. Given N nodes in the network this results in $O(2N)$ messages, and since ARA employs shortest path as metric the diameter d gives the number of steps to reach the destination from the source node. Therefore, we get as time complexity $O(2d)$ for both of the FANT and the BANT.

	TC	CC
Initialization	$O(2d)$	$O(2N)$
Post-Failure 1	0	0
Post-Failure 2	$O(l+d)$	$O(x+N)$
Post-Failure 3	$O(2d)$	$O(2N)$

N = Number of nodes in the network
d = Network diameter
x = Number of nodes affected by a change of the network topology
l = Diameter of the set of nodes affected by a change of the network topology

The post-failure cost depends on the particular scenario. There are three cases which we enumerate as 1, 2, and 3 in the table. The differences of the post-failure cases are as follows:

- Post-Failure 1: A route error occurs at node i and node i has another path to the destination and can transport the packet over that one. Therefore, there are no extra signaling effort and hence TC=CC=0.
- Post-Failure 2: A route error occurs at node i and node i does not have another path to the destination, but the neighbors of i have a path to the destination.

  In this case the packet travels along the affected nodes until the algorithm finds a valid path which it uses to transport the packet to the destination. The destination subsequently sends a BANT to reinforce this path, which leads to $O(x+N)$ messages and $O(l+d)$ steps.
- Post-Failure 3: A route error occurs on node i and neither node i nor its neighbors have an other path to the destination. This leads to an timeout at the source node which initiates a route discovery phase.

3.6 Performance evaluation of ARA

For a detailed performance evaluation of ARA we refer the interested reader to [15, 16].

CHAPTER 4

Energy-Aware Ant Routing Algorithm

So far, we discussed the basic idea of Ant Routing Algorithm and studied its potential in finding short routes and the resulting performance. During this study the *routing metric* was, as usual in this topic, the shortest hop metric. Thus, the basic routing approach considers only the number of hops in the process of route discovery and forwarding. However, as discussed in the introduction wireless multi-hop networks typically consists of resource-constrained nodes, therefore it may be wise to consider additional routing metrics to increase the efficiency of these networks.

As mentioned earlier, ARA allows the inclusion of additional metrics in its forwarding decision process. Although, the idea of introducing other metrics into Equation 2.1 looks intuitive and simple, the implementation into ARA is not straightforward. In this chapter we describe the extension of ARA called EARA, which considers the energy fitness of a path in the route process.

For this, we will discuss first some issues in integrating the additional metric into ARA. Subsequently, we will discuss the possibilities and restrictions of the ideas on simulation results.

4.1 Energy Awareness

Typically, devices in wireless multi-hop networks are going to be battery powered. Therefore, the amount of energy which a node has plays a distinct role. Additionally, energy is the only resource in the node which is volatile. If a nodes available energy deplete, it cannot participate in the network any more. We denote a node without energy as a *dead node*. The situation aggravates if dead nodes lead to the splitting of the network topology, i.e., some of the nodes are not reachable from others.

Like other aspects of wireless multi-hop networks, one cannot solve the energy awareness only on one layer of the communication stack, instead all layers have to deal with it to some degree. An efficient energy-aware mobile node requires the consideration of the energy on the physical layer, on the MAC layer, on the networking layer, on the transport layer, and also on the application layer. The danger of losing the gain of one layer at an another layer is high.

In this paper we will consider the energy awareness of a mobile node only on the routing protocol, i.e., on the networking layer. We assume that on the other layer the energy awareness in considered as well, but do not give any specific solutions for them.

4.2 Energy Model

We consider a node during a time interval $[0, t]$ of the length t. For simplicity we normalize the interval and set $t = 1$. We also divide the considered time interval into three intervals of sending (t_s), receiving (t_r), and idle (t_i).

$$t_s + t_r + t_i = 1$$

The energy for the processing of a packet with x bits is

$$e(x) = a \cdot x + b \tag{4.1}$$

whereas a and b are constants, which depend on the particular NIC and the considered operation, i.e., send, receive, and idle. We assume particular constants for the three operations and derive three equations.

$$e_s(x) = a_s \cdot x + b_s \tag{4.2}$$

$$e_r(x) = a_r \cdot x + b_r \tag{4.3}$$

$$e_i(x) = a_i \cdot x + b_i \tag{4.4}$$

We assume in Equation 4.4 $a_i = 0$. Furthermore, we assume a constant data rate d [bps] of the used NIC. The intervals t_s, t_r, t_i map to appropriate number of bits as given by $d \cdot t$. By this we compute the total energy required in the considered time interval of length t.

$$e_t(x) = e_s(x_s) + e_r(x_r) + e_i(x_i) \tag{4.5}$$

As next we look into the dimension of the mentioned terms. The total required energy $e_t(x)$ results in Joule [J] like the other equations in Equation 4.2, Equation 4.3, and Equation 4.4. The packets are [bit]. Thus, we get

$$a = \frac{\mathsf{VAs}}{\mathsf{bit}} = \frac{J}{\mathsf{bit}}, \qquad b = J$$

For instance Equation 4.2 looks with dimensions as follows:

$$e_s(x)[J] = a_s \left[\frac{J}{\mathsf{bit}}\right] \cdot x[\mathsf{bit}] + b_s[J]$$

The remaining task is to determine the constants in the mentioned equations.

4.3 Energy Consumption

The manufacturer of network interface cards (NICs) publish technical data of their products. We use this data in order to estimate the energy consumption of a NIC. Typically, manufacturers publish four interesting and important data. These are the voltage, the consumed power during transmission (tx), receive (rx), and idle times. The following table shows data on energy consumption in commonly used NICs. It originates from data sheets of the manufacturers of different NICs. It is noteworthy, that the most variance of the NICs is in the idle power consumption. However, it is not clear what the given value exactly means. There might be a confusion between sleep and idle state.

NIC	Voltage [V]	Transmit [A]	Receive [A]	Idle [A]
LinkSys WPC11 ver. 4	3.3	0.430	0.140	0.090
Belkin	3.3	0.300	0.230	0.009
Cisco Aironet	3.3	0.539	0.327	0.203
IBM 11b/g Wireless	3.3	0.551	0.390	NA
Ralink RT2571W	5.0	0.450	0.390	0.190
Atheros AR54SAG	3.3	0.758	NA	NA
WSN (MSB-A2)	3.0	0.031	0.016	0.002

Based on the data in the previous table, we derive the following constants:

$$\begin{aligned}
a_s &= \frac{energySupply[V] \cdot energyConsume[A]}{dataRate[\frac{bit}{s}]} \\
&= \frac{energySupply \cdot energyConsume}{dataRate}\left[\frac{VAs}{bit}\right] \\
&= \frac{energySupply \cdot energyConsume}{dataRate}\left[\frac{J}{bit}\right]
\end{aligned}$$

For the case of $energySupply = 3.3$ V, transmission energy consumption of $energyConsume = 0.30$ A, and $dataRate = 2 \cdot 10^6$ bit/s the constant a_s is

$$a_s = \frac{3.3 \cdot 0.30}{2 \cdot 10^6}[J]$$

The disadvantage of this method is, that the power consumption of the CPU and other peripheral devices is not considered. Despite this, it is a good guess of the consumed power.

Before we finish this section, we will exercise a simple scenario to get an idea about the required energy to transmit data for different time scales. First we derive the energy required to send a single packet of various length. We give the required send energy for typical packet sizes in the following table.

Packet size [Byte]	Energy [J]
64	0.00025344
128	0.00050688
512	0.00202752
1024	0.00405504
2048	0.00811008

For a monitoring application with 1 sample/sec we get the following energy requirements for different time intervals from a second to 10 years. The energy values in the table are for two different length of a sample, i.e., a packet, for a small sample size of 64 byte and a large sample size of 512 byte. The energy requirement is only for a single send operation taken from the previous table.

Time interval	Samples	Data		Energy [J]	
		[64 byte]	[512 byte]	[64 byte]	[512 byte]
Second	$1{,}00 \cdot 10^{+00}$	$6{,}40 \cdot 10^{+01}$	$5{,}12 \cdot 10^{+02}$	$1{,}62 \cdot 10^{-02}$	$1{,}04 \cdot 10^{+00}$
Minute	$6{,}00 \cdot 10^{+01}$	$3{,}84 \cdot 10^{+03}$	$3{,}07 \cdot 10^{+04}$	$9{,}73 \cdot 10^{-01}$	$6{,}23 \cdot 10^{+01}$
Hour	$3{,}60 \cdot 10^{+03}$	$2{,}30 \cdot 10^{+05}$	$1{,}84 \cdot 10^{+06}$	$5{,}84 \cdot 10^{+01}$	$3{,}74 \cdot 10^{+03}$
Day	$8{,}64 \cdot 10^{+04}$	$5{,}53 \cdot 10^{+06}$	$4{,}42 \cdot 10^{+07}$	$1{,}40 \cdot 10^{+03}$	$8{,}97 \cdot 10^{+04}$
Week	$6{,}05 \cdot 10^{+05}$	$3{,}87 \cdot 10^{+07}$	$3{,}10 \cdot 10^{+08}$	$9{,}81 \cdot 10^{+03}$	$6{,}28 \cdot 10^{+05}$
Month	$2{,}59 \cdot 10^{+06}$	$1{,}66 \cdot 10^{+08}$	$1{,}33 \cdot 10^{+09}$	$4{,}20 \cdot 10^{+04}$	$2{,}69 \cdot 10^{+06}$
Year	$3{,}11 \cdot 10^{+07}$	$1{,}99 \cdot 10^{+09}$	$1{,}59 \cdot 10^{+10}$	$5{,}05 \cdot 10^{+05}$	$3{,}23 \cdot 10^{+07}$
5 Years	$1{,}56 \cdot 10^{+08}$	$9{,}95 \cdot 10^{+09}$	$7{,}96 \cdot 10^{+10}$	$2{,}52 \cdot 10^{+06}$	$1{,}61 \cdot 10^{+08}$
10 Years	$3{,}11 \cdot 10^{+08}$	$1{,}99 \cdot 10^{+10}$	$1{,}59 \cdot 10^{+11}$	$5{,}05 \cdot 10^{+06}$	$3{,}23 \cdot 10^{+08}$

4.4 Collection and Distribution of Energy Information

In this section we discuss the extension of ARA. While many ACO-based algorithms consider the residual energy of the neighboring nodes in the forwarding process, EARA estimates the energy fitness of a path. EARA uses the estimated energy fitness of a path and the pheromone value of a future next hop in its forwarding decision. Hence, EARA stores the estimated energy fitness of a path together with the pheromone information in an extended routing table.

4.4.1 Routing Table

We extend the routing table of ARA by means of an additional energy information field for the estimated energy fitness of a path. The routing table structure of a node in EARA is as follows.

t_{la}			
d_1	nh_1	φ_{d_1,nh_1}	ξ_{nh_1}
d_1	nh_2	φ_{d_1,nh_2}	ξ_{nh_2}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
d_k	nh_m	φ_{d_k,nh_m}	ξ_{nh_m}

The entries are d_i destination i, nh_j next hop, φ_{d_i,nh_j} pheromone value to nh_j, and the energy fitness ξ_{nh_j} of a path.

4.4.2 Updating the Energy Information

We introduce periodic energy ant agentss (PEANTs) for updating the energy values in the routing table of a node. The PEANT updates the energy values of all intermediate nodes and reinitialize the energy values similar to the route discovery. However, only destination nodes broadcast PEANTs since only these nodes provide the energy information for paths leading back to the source of the ant agent. A node considers itself a destination node if it received within a threshold a number of packets.

Using a special ant agent for updating the energy information comes at its price. Despite the fact that PEANTs are small control packets consisting of a sequence number, a source address, a time-to-live field, and two energy information fields, broadcasting PEANTs can be a costly operation. Hence, it is important that PEANTs are only sent if the energy state of the network has changed enough to have an impact on the routing decision. The destination nodes keep track of the residual energy of their own battery and broadcast PEANTs if the value of the battery has changed by a given percentage.

4.4.3 Integration into the Forwarding Decision

We described in section 3.2 the forwarding of packets in ARA. Here, we review the forwarding process and discuss the integration of the additional routing metric. For this, we extend the basic Equation 2.1 and consider the estimated energy fitness of path ξ_j over a node j in the forwarding decision.

$$p^i_{d,j} = \begin{cases} \frac{[\varphi^i_{d,j}]^\alpha [\xi_j]^\beta}{\sum_{k \in N_i} [\varphi^i_{d,k}]^\alpha [\xi_k]^\beta} & \text{if } j \in N_i \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

In this equation α and β are constant weights, which help to control to what degree the algorithm considers the shortest path and estimated energy fitness of a path over the next hop in its forwarding decision. The setting of these both constants influences the resulting probability and hence the forwarding.

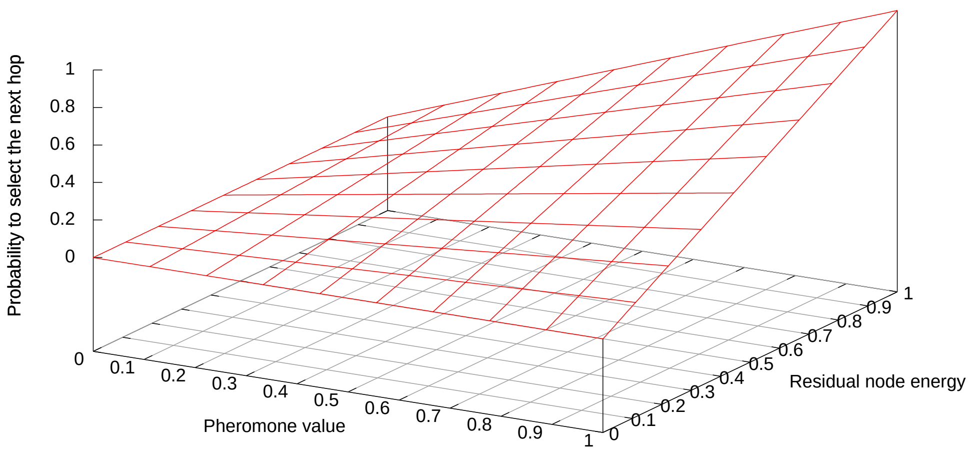


Figure 4.1: *Next hop probability as function of the pheromone value and the residual energy. The higher the pheromone value and residual energy of a neighboring node the higher is the probability a node selects it as next hop.*

Figure 4.1 depicts the probability of selecting the next hop when the algorithms considers the pheromone value and the estimated energy fitness equally. This results in a straight selection of the next hop, since the higher the pheromone and the estimated energy fitness of a path is the higher is the probability to select it in the forwarding process. However, whether this approach also results in an optimal selection of the next hop during a long communication in a way that it optimizes the network lifetime is an open challenging research question.

4.4.4 Path Energy Fitness

In the route discovery phase EARA initializes the energy fitness of a path along with the pheromone values. We define the energy fitness ξ_{init} in Eq. (4.7). It takes the average normalized residual energy of all intermediate nodes ξ_{avg} of an ant agents path and the lowest residual energy ξ_{min} an ant agent encountered into account. The energy fitness also considers a penalty term $\frac{\xi_{avg}-\xi_{min}}{b}$ for low ξ_{min} values. If ξ_{min} is close to ξ_{avg} there will be less reduction. Furthermore, f ξ_{min} is equal to ξ_{avg} the penalty term collapses to zero. The constant b in the penalty term controls the impact of ξ_{min}.

$$\xi_{init} = \xi_{avg} - \frac{\xi_{avg} - \xi_{min}}{b} \mid b \in \mathbb{R} \geq 1 \tag{4.7}$$

For any value $1 \leq b < 2$ the penalty term gives more weight to ξ_{min} value in the initialization. All values $b > 2$ result in ξ_{init} values that are closer to the average ξ_{avg}. Considering the second case will result in a better energy initialization.

4.4.5 Path Energy

We define a path P as a sequence of nodes n_i in the network and denote it as $P = [s, n_1, \ldots, n_k, d]$, where s is the source, d the destination, and n_i the intermediate nodes. In the path P, there are k intermediate nodes. We denote the residual energy level of node n_i at time t as $\xi_{i,t}$. The energy information of the path P at the source node s can be calculated on the basis of the information carried in the FANT and BANT. The energy of the path P is the sum of all energy values $\xi_{i,t}$ of the intermediate nodes.

$$\xi_{P,t} = \sum_{i=1}^{n_k} \xi_{i,t} \tag{4.8}$$

4.4.6 Integrating the Path Energy Fitness

The integration of the energy fitness has some problematic aspects. We will discuss these problems and present some solutions with assets and drawbacks.

- **Route Discovery:** During the route discovery ARA broadcasts FANTs and BANTs into the network in order to discover possible paths between the source and destination. EARA extends the FANT and BANT packets by two additional fields, the average residual energy of the nodes ξ_{avg} and the lowest residual energy an ant agent encountered ξ_{min}.

  In the case of a FANT, the source initializes the fields. Each intermediate node relaying the FANT updates the fields with its own energy information. A BANT is treated in the same way.

- **Route Maintenance:** In the route maintenance phase ARA uses data packets to update the pheromone values in the forwarding nodes.

  Obviously, the pheromone values and the energy values of the nodes on a path show antithetical behavior. While the pheromone values grow large the energy values decrease going to zero.

  A node, the source node as well as intermediate nodes, always increases the pheromone value for the next hop, when it transmits a packet to it. But, it is not that simple for the energy fitness, since it concerns all nodes on the path. Without an explicit feedback, the source node of a flow is not feasible to get a good estimation of the energy on the used path.

We introduce PEANTs for updating the energy fitness of a path. Only destination nodes send PEANTs.

- **Route Failure:** A routing failure occurs when a transmission of a packet on the way was unsuccessful.

  If the local failure handling is not successful, the source node will send out a FANT, which in turn refreshes the pheromone values as well as the energy fitness. However, the route failure handling involves additional packets (send, receive) which furthermore consume energy.

- **Transmission Probability:** A node chooses its next hop in a probabilistic fashion. Links with a high pheromone value are more likely to be chosen. This behavior is modeled in the transmission probability (see (3.2)). If one considers to add further heuristics to the transmission probability, the parameters of this function have to be normalized.

  The extended transmission probability (see (4.6)) considers not only the pheromone value, but also the estimated energy fitness of a path. The maximum available energy of a node is typically known and hence allows to estimate the upper bounds of the estimated energy fitness. However, the maximum pheromone level is not. It is highly dependent of different parameters such as the ongoing traffic in the network or the parameters of the evaporation or reinforcement functions. In simulation scenarios we consider the maximum time-to-live (maxTTL) as a suitable heuristic. The maxTTL influences the TTL live of all received packets and its value depends also on the number of nodes deployed in an area. In addition to the maxTTL and the number of nodes we also consider the deployment area for the pheromone normalization. However, for the usage of the algorithm in real world scenarios we have to find a different approach. Monitoring the traffic over a certain period of time might be a feasible approach.

4.5 Estimation of the Path Energy during Communication

4.5.1 Energy Consumption of a Node

As mentioned earlier, the collection of energy information in a wireless multi-hop network can be costly. Therefore, EARA integrates this costly operation into the FANT and BANT of ARA. But, this has also its drawbacks, since the energy of a node is not constant and decreases over time. A realistic estimation of the energy of the nodes on a certain path is therefore important to support the energy policy of the routing.

Unfortunately, the estimation of the residual energy of a node is not simple, since there are factors which affect the energy consumption of a node. We can summarize these factors with respect to the three states of a node.

- **Sending:** Sending is the most costly operation of a node. A node sends a packet, if it has got a data from the application layer or if it has to forward a packet.

- **Receiving:** Although the receive power of a node is less than that of a send operation, the number of receives can be quiet higher than send operations, since a node has to check all packets originated from its neighborhood, whether the node is the destination or not.

  If a node is in the communication range of k other nodes, it may receive up to k times a packet, whereas the sending occurs only once.

- **Idle:** If a node is not sending or receiving any packets, it is in the idle mode. In this mode, a node consumes the least energy. The consumed power in the this mode depends heavily on the supported sleep modes of the platform and can vary.

We consider the energy of node i in a time interval $[0, \Delta t]$ of the length Δt, which we normalize to $[0, 1]$. Let p_t the residual energy of node i at time t.

We define the idle time t_i of the node during this time interval as the time without send and receive operations.

$$t_i = 1 - t_s - t_r$$

The total consumed power of node i during the time interval is:

$$e(\Delta t) = e_s(t_s) + e_r(t_r) + e_i(t_i) \tag{4.9}$$

The energy of the node at the end of the considered time interval Δt is:

$$p_{\Delta t} = p_t - e(\Delta t)$$

The total energy of a path has to be corrected by the sum of the consumed energies at all nodes.

4.5.2 Path Energy Estimation

We consider a path $P = [s, n_1, \dots, n_k, d]$ between the source s and the destination d with k intermediate nodes. The energy of the path P at the sender after receiving the BANT is as follows:

$$\xi_P = \sum_{i=1}^{n_k} \xi_i \tag{4.10}$$

This time impinges the initialization of the path energy at the source node, and thus we define:

$$\xi_P(t_0) = \sum_{i=1}^{n_k} \xi_i \tag{4.11}$$

We are interested in the energy value of this path after sending a packet over this path at the source node s and also on the intermediate nodes n_i. Let c_{packet} be the length of the packet in bits and Δt be the transmission time from s to d. Further, let $p_i(\Delta t)$ the consumed energy of node i during the transmission time Δt. The total amount of consumed energy on the path P is:

$$p(\Delta t) = \sum_{i=1}^{k} p_i(\Delta t) + p_s(\Delta t) + p_r(\Delta t) \tag{4.12}$$

Hence, the path energy at time $t_0 + \Delta t$ is:

$$\begin{aligned} p_{t_0+\Delta t} &= p_{t_0} - p(\Delta t) \\ &= p_{t_0} - \sum_{i=1}^{k} p_i(\Delta t) - p_s(\Delta t) - p_r(\Delta t) \end{aligned} \tag{4.13}$$

Here, we define $p_i(\Delta t)$ as in Equation 4.9. Now, our estimation of the consumed energy of a node depends only on the number of send and receive operations during the considered time Δt for each node as in Equation 4.12.

4.6 Performance Evaluation

4.6.1 Simulation Setup

We carried out the simulation using `libARA` [11], a framework for the implementation and evaluation of ant routing algorithms. `libARA` is based on `OMNeT++` and `INETMANET`. The simulation area is of the size of 2000 m × 400 m. About 70 mobile nodes move according to the Random Waypoint mobility model 900 s long with a velocity chosen from the interval of $[1, 5]$ m/s. We also study pause times of 0 s, 300 s, 600 s and 900 s. The source and destination nodes are unique. All presented results are averages of 30 simulation replications. We study the performance of ARA and the energy aware version EARA in three different settings, which are as follows:

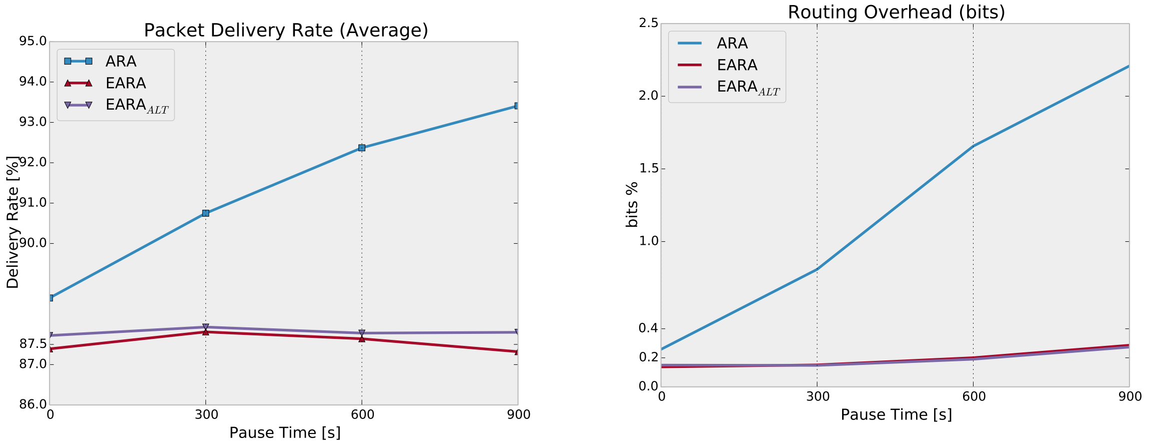
- **ARA:** We use ARA in its standard setting where a node decides stochastically for every packet its next hop.
- **EARA $\alpha = 1, \beta = 2$:** We set the α and β parameter of the transmission probability (Equation 4.6) to 1 and 2, respectively.
- **EARA $\alpha = 1, \beta = 5$:** We set the α and β parameter of the transmission probability (Equation 4.6) to 1 and 5, respectively.

We use the energy dead series (EDS) in order to study the performance of ARA and EARA. For the EDS we log for each simulation run the time when a node dies due to energy depletion. We sort the times ascending and calculate after all simulation runs the averages for each entry. Hence, EDS is a function of time. It allows us to study how mobile nodes exhaust their energy. A node without energy cannot participate in the network anymore, i.e., it cannot forward packets. In the worst case, this can lead to the splitting of the network topology.

4.6.2 Packet Delivery Rate and Routing Overhead

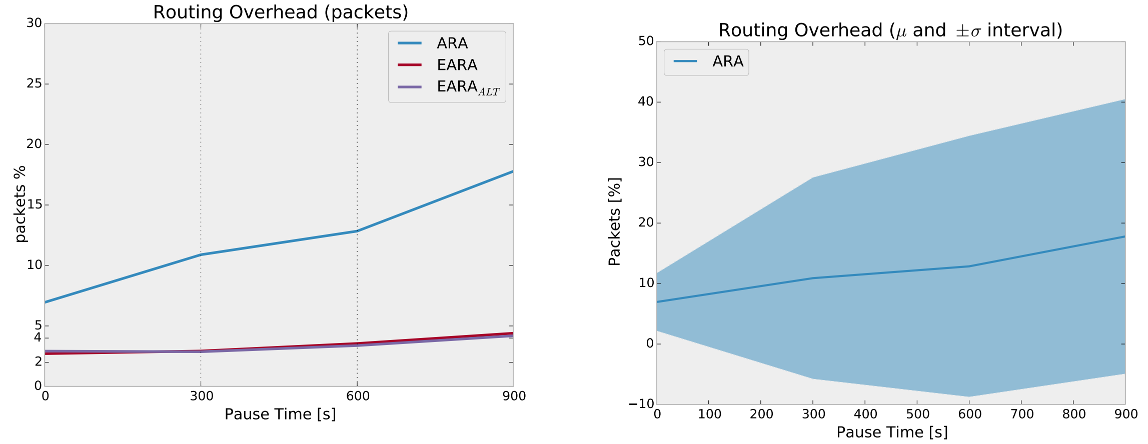
The packet delivery rate (PDR) and the routing overhead in packets and in bits give a general impression on the performance of the routing algorithm. Fig. 4.2 depicts the PDR and the routing overhead in bits and packets for ARA and EARA. The PDR of EARA in both weight settings in Fig. 4.2a is lower than the PDR of ARA, but still around 90 %. The EARA setting refers to a weight setting of $\alpha = 1$ and $\beta = 2$ while EARA$_{\text{ALT}}$ refers to a weight setting of $\alpha = 1$ and $\beta = 5$ respectively. Despite the weight setting the PDR of both EARA settings differs up to 2 % at max. In addition, the PDR of EARA seems to be rather static, despite the node mobility. We consider this a direct result of the PEANTs. The PDR of ARA increases with lower mobility. This is a direct result of less changes in network topology and hence, less measurable route discoveries. However, EARA sends out periodically PEANTs to update the energy information. Hence, the PEANTs are similar to a regular route discovery.

We measure the routing overhead by means of the sent bits and sent packets which are control bits or packets respectively (in percent). Fig 4.2b and Fig. 4.2c show the routing overhead in bits, respectively packets. We expected that the routing overhead of EARA would be higher than the overhead of EARA. This is due to the fact that we introduce with PEANTs special control packets which result in more control traffic in the network and hence, a higher overhead. However, the overhead for EARA is rather static while the overhead of ARA seems to grow with less mobility. We investigated the logs of the simulation runs and apparently routing failures and subsequent route failure messages caused a lot of the routing overhead. However, we have conduct further analysis on this issue. Previous simulation results, such as in [12], have shown more accurate results.



(a) *Packet Delivery Rate for ARA and EARA*

(b) *Routing Overhead in Bits for ARA and EARA*



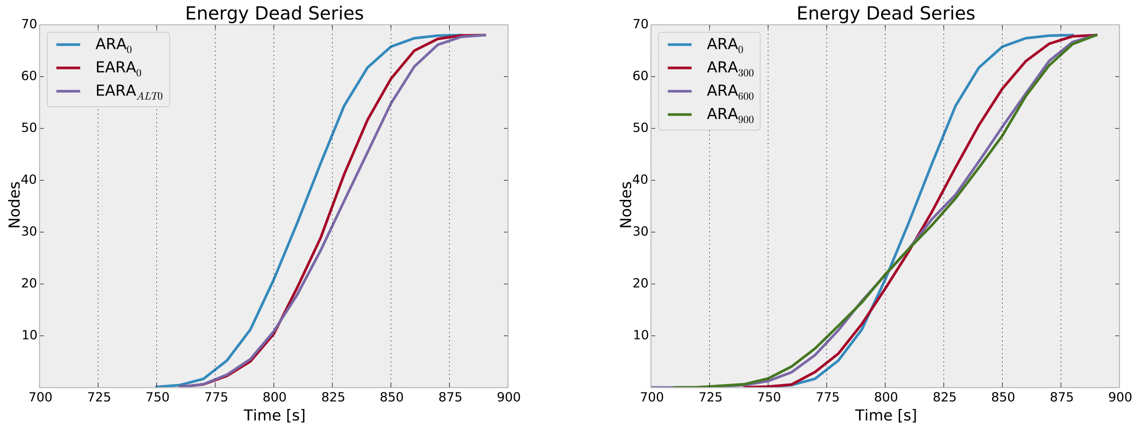
(c) *Routing Overhead in Packets for ARA and EARA*

(d) *Detailed Overhead of Packets in ARA*

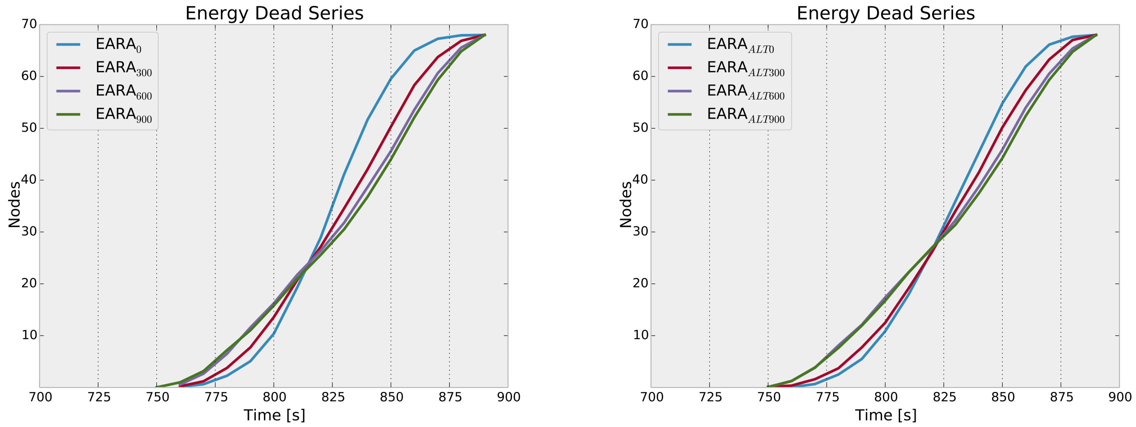
Figure 4.2: *The packet delivery rate and routing overhead of ARA and EARA in bits and packets.*

4.6.3 Energy Dead Series

The EDS allows to draw conclusions on the network lifetime, i.e. how many nodes are alive until at a certain point of time and can participate in the overall task of the network. Fig. 4.3b to 4.3d depict the EDS for every scenario separately. In all scenarios, nearly all nodes die at the end. This does not hold true for the source and destination nodes. We set the available energy of source and destination node twice as high as every other node in order to avoid that both nodes die before the simulation finishes. In order to compare the EDS of ARA and EARA more easily Fig. 4.3a shows the EDS of ARA and EARA with a pause time of 0 s. This figure includes for EARA both weight settings. While both EARA settings are close in terms of dead nodes at a certain point of time, both perform better than ARA. A careful study of both EARA settings in Fig. 4.3c and Fig. 4.3d allows to draw the conclusion that both settings show the same tendency across pause times, but EARA with an alternative parameter setting of $\alpha = 1$ and $\beta = 5$ outperforms the initial EARA setting.



(a) *Energy Dead Series for ARA and EARA.* **(b)** *Energy Dead Series for ARA*



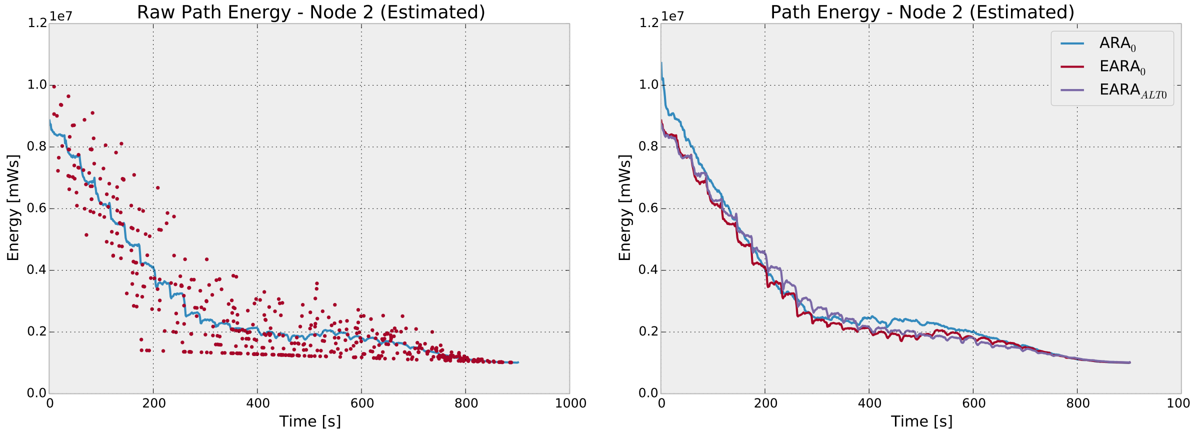
(c) *Energy Dead Series for EARA* $\alpha = 1$ *and* $\beta = 2$ **(d)** *Energy Dead Series for EARA* $\alpha = 1$ *and* $\beta = 5$

Figure 4.3: *Energy Dead Series for ARA and EARA. For the sake of clarity we only consider scenarios with a pause time of 0 s in Fig 4.3a.*

4.6.4 Path Energy

The path energy is a difficult metric for multiple reasons. First, for every simulation run in our study different paths are chosen, so we cannot take the path energy and create averages. Second, the random placement of the nodes and the used mobility model result in a slightly unpredictable behavior. Hence, We perform a Nadaraya-Watson kernel regression on the data gathered for every scenario in our study. The kernel regression is not an exact and precise estimate and thus, it is difficult to compare the results for each scenario. However, it allows to get a basic understanding. Fig. 4.4a shows the kernel regression for the path energy in scenario $\text{EARA}_{\text{ALT0}}$ and the raw data. We can clearly recognize the wide spread of data points.

Fig. 4.4b depicts the path energy for ARA and EARA. Here, we focus for the sake of clarity on scenarios with a pause time of 0 s. The path energy decreases over time, which is most likely a result over the available energy depleting on every node. Another explanation is that the number of nodes on a path reduces over time. There have been studies on mobility models and for some models, the nodes tend to move towards the center of the simulation area. Hence, we also analyzed the hop count and the number of nodes on a path indeed decreases over time. However, this could also be the result of dead nodes. There also seem to be a better utilization of the energy in ARA, but since the kernel regression is not as precise as the figure suggests, a more in-depth analysis is required.



(a) *Path Energy* $\text{EARA}_{\text{ALT0}}$ *and raw data* **(b)** *Path Energy for ARA and EARA.*

Figure 4.4: *Path Energy for ARA and EARA. For the sake of clarity we only consider scenarios with a pause time of 0 s in Fig 4.4b.*

4.7 Summary

In this chapter we introduced EARA, an energy-aware variant of ARA. EARA estimates the energy fitness of a path and considers that information in its routing decision process. Although, the inclusion of additional metrics into the forwarding process seems simple, it comes with its own challenges. Particularly, estimating the energy fitness of a path and disseminating that information. In addition, the nature of the pheromones and the energy information are total different, which aggravate the inclusion of the energy information in the forwarding process.

The results we obtain from simulations are promising, but not exhaustive in terms of different performance metrics for routing algorithms. Particularly, the path energy and the used mobility models require further analysis. For this we plan additional simulation studies with larger networks and different traffic patterns.

CHAPTER 5

Related Work

5.1 Introduction

The discussion about related work is two-fold. The first part focuses on energy-aware ant routing algorithms which not only consider the artificial pheromone value of an link towards the destination, but also the energy level of the forwarding node or nodes. While the actual decision for a path in ant routing algorithms is probabilistic, these algorithms favor paths with a high pheromone and energy value. We interpret the combination of both parameters as a constraint which an algorithm needs to maximize. Therefore, the second part gives an overview of the state of the art in constraint-based routing.

5.2 Energy-Aware Ant Routing Algorithms

Ant Routing Algorithm for Mobile Ad Hoc Networks

Hussein et. al. propose in [20] the ant aouting algorithm for mobile ad hoc networks (ARAMA). The transmission probability of ARAMA contains a heuristic parameter in its numerator which is a place holder for information's of a neighbor node. This might include for example the battery's remaining energy or processing power of a neighbor node or the link's bandwidth or signal-to-noise ratio. The algorithm adds additional data to the FANTs in order to optimize for a specified set of parameters. The authors denote this data as the *node's local and global path information*. Since adding data to the FANT increases the overall size of the ant agent, the authors propose the usage of a normalized index. The algorithm compares the index to a reference value at the destination. The reference value is the best index value a destination node received over a period of time. The authors use a simplified battery model of a Lucent WaveLAN card [10]. In addition, the authors only consider send and receive operations for consuming energy. While the authors emphasize that it is possible to use different metrics for the heuristic value the author only consider the remaining energy of a node and the number of hops.

Ant Colony Optimization Based Location Aware Routing for Wireless Sensor Networks

The ant colony optimization based location aware routing for wireless sensor networks (ACLR) [26] tries to minimize the delay of transmissions while maximizing the overall network lifetime. Each node is aware of its own location and every node considers in its routing decision only neighbored nodes as next hop which are closer to the base station. The algorithm stores the location information, the residual energy of a node, the neighbors of the node and the base station in a memory block. It also considers two additional factors for the transmission probability: the location and the residual energy of a node. The authors define the location as a function which considers the distance from the current node and the next hop to the base station. The algorithm applies weights to the parameters of the transmission probability.

Energy-Efficient Ant-Based Routing

The energy-efficient ant-based routing algorithm (EEABR) [4] is a routing algorithm for WSNs. The principles of AntNet [5] form the basis for this algorithm. EEABR tries to minimize memory requirements as well as the overall energy consumption of the original AntNet algorithm. Here, the BANT only holds information on the last two visited nodes. A node which receives a BANT saves the nodes the BANT previously visited. In addition, the transmission probability considers the artificial pheromone value and the remaining energy of the next hop. The amount of pheromones an ant agent deposits on a path relates to the minimum and average value of the energy levels of nodes on this path. Ant agents loose part of the pheromone strength during their way to the source node. This behavior aims in a better pheromone adaption, in particular if a sink is free to move around. The authors used ns-2 for the evaluation of the algorithm.

5.3 Constrained-Based Routing

Typically, algorithms in best-effort routing try to find good paths in a network based on a single measure, the number of hops. While for most cases in wired networks this is sufficient, in WMHNs multiple resources demand for optimization at the same time. Examples for these resources are i) the available energy, ii) the bandwidth, and iii) the load on the medium.

Quality of Service (QoS) in routing considers both, the requirements of an application and the availability of network resources. One example for an WMHN where QoS is essential are WSNs. Here, sensor nodes monitor the environment and send data to a sink node. However, an equilibrium between sensing and sending data and the overall lifetime of the network has to be found, since sensor nodes are constrained in terms of available energy. QoS in routing opposes a set of challenges such as the dissemination of dynamic state-dependent information, aggregation of states and computation of constrained paths [23]. We can summarize multiple constraints for the routing of data as multi-constrained path problem (MCP). Furthermore we extend the definition of an graph (see 2.2) in order to define the MCP [23]. Let $\#\mathfrak{m}$ be the number of QoS measures. An $\#\mathfrak{m}$-dimensional weight vector consisting of non-negative QoS weights as components characterizes each edge $\mathfrak{e}$. Researchers distinguish between additive and non-additive QoS measures. For example, the delay is an additive measure while the available bandwidth is an non-additive measure. The QoS of an path for an additive measure is the sum of the corresponding weights along the path. For non-additive measures the minimum or maximum value of the weights along a path denotes the QoS of such an path. A special case of the MCP is the restricted shortest path (RSP) where two weights, a delay and a cost weight, are assigned to each edge in a graph. The goal of the RSP is to find the shortest path from a source to a destination node which minimizes the total cost while having a delay below a given threshold. Researchers consider the MCP and its variants nondeterministic polynomial time (NP)-complete [13].

Kuipers et. al. provide in [23] an overview over constraint-based path selection algorithms for QoS routing. The authors focus on heuristics for the MCP. In [22] the same authors give an performance evaluation RSP and MCP heuristics.

The QoS-enhanced OLSR (QOLSR) [2] is a routing schema for the optimized link state routing protocol (OLSR) which uses bandwidth and delay as a metric. The proposed algorithm finds a path with the maximum bandwidth and if there are multiple paths, it choses the path with the shortest delay. The authors emphasize that it is not feasible to treat each metric individually due to algorithm complexity. This is also pointed out in [23].

In [21] the authors apply two methods in order to integrate multiple metrics in an OLSR based routing protocol. First, the authors apply analytic hierarchy process (AHP), a technique from the area of decision making. Second, the authors use pruning in order to remove links below a certain threshold. The latter step aims in reducing the amount of matrix computations which are the result of possible paths between source and destination. The authors use expected transmission count (ETX) and minimum delay (MD) as metrics in a case study.

5.4 Summary

The presented energy-aware routing algorithms based on the ACO metaheuristic are only an small excerpt of ant-based routing algorithms for WMHNs. Ant routing algorithms share commonalities and researchers studied these class of algorithms extensively, both analytically and in simulation. In particular, studies on ant routing algorithms based on simulation often use simple radio wave propagation models or make unrealistic assumptions which do not hold true for the real world. Since the first publications on ant routing based algorithms, particularly ARA [18] and AntHocNet [9] succeeding publications proposed mostly only minor extensions and improvements.

ARAMA was one of the first ant-based algorithms which considered energy as an criteria for routing. The algorithm adds local and global path information to the forward ant agents in order to spread information about certain variables of a node. While the authors propose to use a normalized index in order to reduce the size of the ant agents, the question about the actuality of the data remains open. In addition, trying to provide a global view of the network is unrealistic.

The ACLR introduces a routing scheme based on the location of nodes and the residual energy of nodes. Unfortunately a energy model is missing. However, the authors consider the energy consumption in the process of packet delivery in their simulation study. The authors also consider different weights for different scenarios, but do not provide a study. While it is reasonable to adapt the weights to different scenarios, the overall impact of different weights on the performance of the algorithm requires further analysis.

EEABR tries both, minimizing the packet size and maximizing the network lifetime. The algorithm provides a function for the deposition of pheromones. It considers the remaining energy of different nodes on a path towards a sink. However, using the AntNet algorithm as a basis for EEABR is question worthy. The authors of AntNet designed the algorithm for wired networks. In addition, the authors used for their simulation-based study ns-2 which is to some extent questionable. Typically, applications in WSNs interact heavily with the network. However, it is not feasible to model such an interaction in ns-2. Additionally, the energy models, MAC protocols and packet forms provided in ns-2 differ from the models and packet formats found in most sensor networks.

While energy-aware ant routing algorithms do not only consider the artificial pheromone value of an link towards a destination, but also the energy value of the node or nodes we can interpret the routing decision as a constraint problem. The constraint-based search for paths in a network can be defined as a MCP. While the MCP is NP-complete, different heuristics can be applied. Kuipers et. al. present in [23] heuristics for the MCP and evaluate them in [22]. Dependent of the concrete application of additive and non-additive QoS measures, one can apply different approaches.

QOLSR combines a non-additive and additive QoS measure, delay and bandwidth, in OLSR. The algorithm searches first for the path with the highest bandwidth and selects in a next step the path with the lowest delay. The approach is simple and might applicable to other pairs of additive/non-additive metrics.

In [21] the authors apply a technique from decision making and pruning in order to reduce the computational overhead. This results in a total reduction of matrix computations by a factor of $1.9 \cdot 10^2$. While the initial results are promising and an application to MANETs and WSNs nearby, the approach might not applicable at all to MANETs or WSNs. Typically, network topologies in MANETs are dynamic and the time required for computations might still be to high. In addition, the computational power of WSNs are probably to low for complex matrix computations.

EARA aims in maximizing the overall network lifetime. Hence, the algorithm tries both: use the shortest path towards a destination and balance the energy consumption across nodes at the same time. Pheromone values of good paths increase over time while the remaining energy of a relaying node on a path decreases. One can express maximizing the network lifetime as a MCP. However, heuristics in constrained-based routing do not take the unreliable shared medium into account. This is not only a concern in terms of information dissemination (required for the heuristics), but also for currentness of collected data. In particular, asymmetric unstable links and topology changes are a major concern in WMHN. However, we are not ware of any further investigation of the application of MCP heuristics on ant routing algorithms with multiple routing metrics.

CHAPTER 6

Summary

In this paper we presented and studied an energy-aware variant of ARA called EARA. EARA considers two routing metrics in its forwarding process: i) number of hops between source and destination, ii) estimated energy fitness of a path. The goal was to study the impact of both routing metrics on the network performance in terms of the packet delivery rate, overhead and the network lifetime.

We used for our study in this paper `libARA`, a framework for the implementation and evaluation of ant routing algorithms. Our results are promising and indicate that the integration of the second routing metric into ARA works. However, the study is not exhaustive and hence, we have to conduct further investigations in order to conclude significant knowledge.

In the future we will extend the study of EARA manifold. First we will study larger simulation set-ups to see to what degree the algorithm and protocol scales. We will also study the performance of EARA for specific settings of different kinds of wireless multi-hop networks like, WSN and WMN.

Acknowledgements

This work originates back to one of the authors research stay in 2005 at the International Computer Science Institute, Berkeley. Many of the outlined problems and challenges in WMHN are still unsolved. Work on estimating the energy fitness of paths was conducted in [14]. Final work was carried out by one of the authors at the SICS Swedish ICT in Stockholm, Sweden. This work was partially funded by a research grant from the Deutscher Akademischer Austausch Dienst (German Academic Exchange Service, DAAD).

Bibliography

[1] Akyildiz, I.F., Wang, X.: Wireless Mesh Networks. Wiley & Sons (March 2009)

[2] Badis, H., Munaretto, A., Agha, K., Pujolle, G.: QoS for Ad Hoc Networking Based on Multiple Metrics: Bandwidth and Delay. In: Proc. MWCN (2003)

[3] Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm intelligence: from natural to artificial intelligence. Oxford University Press (1999), iSBN: 0-19-513159-2

[4] Camilo, T., Carreto, C., Silva, J., Boavida, F.: An energy-efficient ant-based routing algorithm for wireless sensor networks. Ant Colony Optimization and Swarm Intelligence pp. 49–59 (2006)

[5] Caro, G.D., Dorigo, M.: Antnet: Distributed stigmergetic control for communication networks. Journal of Artificial Intelligence Research 9, 317–365 (1998)

[6] Corson, M.S., Ephremides, A.: A distributed routing algorithm for mobile wireless networks. ACM-Baltzer Journal of Wireless Networks 1(1), 61–81 (February 1995)

[7] Dorigo, M., Di Caro, G.: The ant colony optimization meta-heuristic. In: Corne, D., Dorigo, M., Glover, F. (eds.) New Ideas in Optimization, pp. 11–32. McGraw-Hill, London (1999)

[8] Draves, R., Padhye, J., Zill, B.: Comparison of routing metrics for static multi-hop wireless networks. In: SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications. pp. 133–144. ACM, New York, NY, USA (2004)

[9] Ducatelle, F.: Adaptive routing in ad hoc wireless multi-hop networks. Ph.D. thesis, Università della Svizzera Italiana, Istituto Dalle Molle di Studi sullIntelligenza Artificiale (2007)

[10] Feeney, L.: An energy consumption model for performance analysis of routing protocols for mobile ad hoc networks. Mobile Networks and Applications 6(3), 239–249 (2001)

[11] Frey, M., Grosse, F., Günes, M.: libARA: A Framework for Simulation and Testbed Based Studies on Ant Routing Algorithms in Wireless Multi-hop Networks. In: Proceedings of the 7th International Conference on Performance Evaluation Methodologies and Tools. pp. 304–309. ValueTools '13, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), Brussels, Belgium (2013), `http://dx.doi.org/10.4108/icst.valuetools.2013.254382`

[12] Frey, M., Grosse, F., Günes, M.: Energy-Aware Ant Routing in Wireless Multi-Hop Networks. In: IEEE International Conference on Communications (ICC). Sydney, Australia (June 2014)

[13] Garey, M., Johnson, D.S.: Computers and intractability: A guide to the theory of NP-Completeness. Freeman (1979)

[14] Grosse, F.: Optimization And Evaluation Of Energy Aware Ant Routing Algorithm Strategies Based On Network Simulations. Master's thesis, Department of Computer Science, Freie Universität Berlin, Berlin, Germany (2013)

[15] Günes, M.: Routing and Addressing in mobile multi-hop Ad-hoc Networks (in german). Ph.D. thesis, RWTH Aachen University, Aachen, Germany (January 2004)

[16] Günes, M., Kähmer, M., Bouazizi, I.: Ant-routing-algorithm (ARA) for mobile multi-hop ad-hoc networks – new features and results. In: Belghith, A., Tabbane, S., Ali, N.B., Gazdar, A. (eds.) Proceedings of the 2nd Mediterranean Workshop on Ad-Hoc Networks (Med-Hoc-Net'2003). pp. 9–20. Mahdia, Tunesia (25-27, June 2003)

[17] Günes, M., Reibel, J.: An IP Address Configuration Algorithm for Zeroconf. Mobile Multi-hop Ad-Hoc Networks. In: Proceedings of the International Workshop on Broadband Wireless Ad-Hoc Networks and Services. ETSI, Sophia Antipolis, France (September 2002)

[18] Günes, M., Sorges, U., Bouazizi, I.: ARA - The Ant-Colony Based Routing Algorithm for MANETs. In: Olariu, S. (ed.) Proceedings of the 2002 ICPP Workshop on Ad Hoc Networks (IWAHN 2002). pp. 79–85. IEEE (August 2002)

[19] Haas, Z.J., Halpern, J.Y., Li, E.L.: Gossip-based ad hoc routing. In: Proceedings IEEE INFOCOM 2002, The 21st Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE, New York, USA (June 23-27 2002), `http://www.ieee-infocom.org/2002/papers/822.pdf`, iSBN 0-7803-7477-0

[20] Hussein, O., Saadawi, T., Lee, M.: Probability routing algorithm for mobile ad hoc networks' resources management. Selected Areas in Communications, IEEE Journal on 23(12), 2248–2259 (2005)

[21] Jr., W.A.M., Aguiar, E., Abelem, A.J.G., Stanton, M.: Using Multiple Metrics with the Optimized Link State Routing Protocol for Wireless Mesh Networks (2008), `http://www.lbd.dcc.ufmg.br:8080/colecoes/sbrc/2008/064.pdf`

[22] Kuipers, F., Korkmaz, T., Krunz, M., Van Mieghem, P.: Performance evaluation of constraint-based path selection algorithms. Network, IEEE 18(5), 16–23 (2004)

[23] Kuipers, F., Van Mieghem, P., Korkmaz, T., Krunz, M.: An overview of constraint-based path selection algorithms for qos routing. Communications Magazine, IEEE 40(12), 50–55 (2002)

[24] Macker, J.P., Corson, M.S.: Mobile ad hoc networking and the IETF. Mobile Computing and Communications Review 2(1), 9–14 (1998)

[25] Royer, E.M., Toh, C.K.: A review of current routing protocols for ad hoc mobile wireless networks. IEEE Personal Communications pp. 46–55 (April 1999)

[26] Wang, X., Li, Q., Xiong, N., Pan, Y.: Ant Colony Optimization-Based Location-Aware Routing for Wireless Sensor Networks. Wireless Algorithms, Systems, and Applications pp. 109–120 (2008)

[27] Yang, Y., Wang, J., Kravets, R.: Designing routing metrics for mesh networks. Proceedings of the IEEE Workshop on Wireless Mesh Networks (WiMesh). IEEE Press (2005)