

1-1-2012

# Abstract ACO-EEAODR: An Ant-Swarm Inspired Energy-Efficient Ad Hoc On-Demand Routing Protocol for Mobile Ad Hoc Networks

Waqas Shah  
*Ryerson University*

Follow this and additional works at: <http://digitalcommons.ryerson.ca/dissertations>



Part of the [Computer Sciences Commons](#)

---

## Recommended Citation

Shah, Waqas, "Abstract ACO-EEAODR: An Ant-Swarm Inspired Energy-Efficient Ad Hoc On-Demand Routing Protocol for Mobile Ad Hoc Networks" (2012). *Theses and dissertations*. Paper 1787.

This Thesis is brought to you for free and open access by Digital Commons @ Ryerson. It has been accepted for inclusion in Theses and dissertations by an authorized administrator of Digital Commons @ Ryerson. For more information, please contact [bcameron@ryerson.ca](mailto:bcameron@ryerson.ca).

**ACO-EEAODR: AN ANT-SWARM INSPIRED ENERGY-EFFICIENT AD HOC  
ON-DEMAND ROUTING PROTOCOL FOR MOBILE AD HOC NETWORKS**

By

**Waqas Shah**

**B.C.I.T., NED University of Engineering & Technology,**

**Karachi, Pakistan, December 2006**

A thesis

presented to Ryerson University

in partial fulfillment of the  
requirements for the degree of

Master of Science

in the Program of  
Computer Science

Toronto, Ontario, Canada, 2012

© Waqas Shah 2012

## **Author's Declaration**

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

## **Abstract**

# **ACO-EEAODR: An Ant-Swarm Inspired Energy-Efficient Ad Hoc On-Demand Routing Protocol for Mobile Ad Hoc Networks**

©Waqas Shah, 2012

Master of Science  
Computer Science  
Ryerson University

As the world's economic activities are expanding, the energy comes to the fore to the question of the sustainable growth in all technological areas, including wireless mobile networking. Energy-aware routing schemes for wireless networks have spurred a great deal of recent research towards achieving this goal. Recently, an energy-aware routing protocol for MANETs (so-called energy-efficient ad hoc on-demand routing protocol (EEAODR) for MANETs was proposed, in which the energy load among nodes is balanced so that a minimum energy level is maintained and the resulting network lifetime is increased. In this paper, an Ant Colony Optimization (ACO) inspired approach to EEAODR (ACO-EEAODR) is proposed. To the best of our knowledge, no attempts have been made so far in this direction. Simulation results are provided, demonstrating that the ACO-EEAODR outperforms the EEAODR scheme in terms of energy consumed and network lifetime, chosen as performance metrics.

## **Acknowledgements**

It is my honour to express my deep gratitude to people who made this challenge possible for me. I would like to thank my supervisors Dr. Isaac Woungang and Co-supervisor, Dr. Alex Ferworn, for giving me the opportunity to work under them. They offered me such a challenging topic for my thesis and with their strong support and constant guidance; I am finally able to achieve my goals. They did not only guide me in the course of this thesis, but also provided me an opportunity to be benefited from their vast knowledge.

I would also like to thank Dr. S. K. Dhurandher as it is his work that provided me a basis to put together this research thesis. His enormous knowledge about energy-efficient routing algorithms in mobile ad hoc networks, and wireless routing in general, was a source of immense help and it is his time and guidance that made this thesis better.

Lastly, and most importantly, I would like to thank my mother Tanveer Anjum, my father Azhar Hussain Shah and my wife Maria Batool. It is their prayers and support that gave me the courage and ability to fulfil my goals.

## **Dedication**

To

My supervisor Dr. Isaac Woungang

I would not be here without his support and strong belief in me.

# Contents

Abstract .....	iii
Acknowledgements .....	iv
Dedication .....	v
List of Tables .....	vii
List of Figures .....	vii
List of Abbreviations .....	ix
<b>1 Introduction.....</b>	<b>1</b>
1.1 Context .....	1
1.2 Research Problem.....	2
1.3 The Approach.....	2
1.4 Thesis Contributions: .....	3
1.5 Thesis Outline .....	3
<b>2 Background Research.....</b>	<b>5</b>
2.1 Background .....	5
2.2. Related Work .....	8
<b>3 Methodologies.....</b>	<b>13</b>
3.1 EEAODR Algorithm Design.....	13
3.2 ACO-EEAODR Algorithm Design .....	15
<b>4 Performance Evaluation.....</b>	<b>27</b>
4.1 Simulation Tool.....	27
4.2 Performance Metrics .....	27
4.3 Simulation Parametes.....	28
4.4 Simulation Results.....	29
<b>5 Conclusions .....</b>	<b>38</b>
References.....	39

# List of Tables

Table 1: Simulation Parameters.....	29
Table 2: Energy consumed in path selection vs. simulation time when using ACO-EEAODR and EEAODR.....	29
Table 3: Energy consumed in path selection for ACO-EEAODR vs. EEAODR.....	30
Table 4: Simulation Time vs. Number of Nodes for EEAODR & ACO-EEAODR.....	32
Table 5: Response Time vs. Simulation Time for EEAODR & ACO-EEAODR.....	33
Table 6: Hop Counts vs. Number of Nodes for EEAODR & ACO-EEAODR.....	35
Table 7: Network Lifetime vs. Number of packets for EEAODR & ACO-EEAODR.....	36



# List of Figures

Figure 1: EEAODR pseudo-code.....	15
Figure 2: The ACO-EEAODR flowchart.....	25
Figure 3: ACO-EEAODR pseudo-code.....	26
Figure 4: Energy consumed in path selection vs. simulation time when using ACO-EEAODR and EEAODR.....	30
Figure 5: Energy consumed in path selection for ACO-EEAODR vs. EEAODR.....	31
Figure 6: Simulation Time vs. Number of Nodes for EEAODR & ACO-EEAODR.....	32
Figure 7: Response Time vs. Simulation Time for EEAODR & ACO-EEAODR.....	34
Figure 8: Hop Counts vs. Number of Nodes for EEAODR & ACO-EEAODR.....	35
Figure 9: Network Lifetime vs. Number of packets for EEAODR & ACO-EEAODR.....	36

# List of Abbreviations

MANETs	Mobile Ad-hoc NETWORKs
ACO	Ant Colony Optimization
MMAS	Min-Max Ant System
ASRank	Ant System Rank
EEAODR	Energy Efficient Ad-hoc On Demand Routing

# Chapter 1

## Introduction

### 1.1 Context

The ubiquity of wireless communications combined with the advent of new wireless technologies have revolutionized network communication. It has enabled the research community to focus on implementing new standards and techniques to deal with low power and energy efficiency for the next generation networks [1]. Over the years, energy efficiency has become an important challenge across all forms of networking. Indeed, from a network design perspective, the development of energy-efficient communication and networking algorithms and protocols for the next generation networks is an important challenge that deserves some attention.

A wireless network can be defined as a network of devices (nodes) that are connected to each other by means of a wireless communication medium (i.e. without wires). This setting can enable ease of communication, particularly in the case of mobile nodes. Wireless communication technologies have recently made great advancements such as Bluetooth, WiFi, WiMax, to name a few. Taking advantage of these technologies, multi-hop wireless ad hoc networks considered as a subclass of wireless networks, have spurred a great deal of interest. In both the research and industrial communities, they have been particularly useful for commercial, healthcare and military applications [2]. This is mainly due to the fact that they are infrastructure-less and require multiple hops for connecting all the nodes to each other. Consequently, the relaying of the messages from one mobile node to another, and the peculiarity of the wireless transmission medium (i.e. noise, interference, to name a few) are some major fundamental issues. In addition to this, when the power source of a node in such networks is less or costly, the energy efficiency

becomes a key concern [2], as it is directly related to the design and operation of such networks, particularly the routing operation.

Routing in MANETs takes place by mutual communication. Due to the lack of infrastructure, various routing algorithms, that include 2-way communication between the nodes for route selection purposes, are used. Each node needs to be equipped with specific configuration which helps the node in route discovery. This involves information regarding the current states of other nodes, information on the links that the node has with other nodes, a routing algorithm that it needs to follow for packet routing. The challenge involved in routing packets in mobile ad hoc networks (MANETs) is the same. Each node needs to be constantly updated with the current network state and active/inactive nodes, and doing so on-the-fly requires a routing algorithm. There are different routing algorithms focusing on different optimization parameters such as time delay, path length, bandwidth, maximum links capacity, etc. In this thesis, our focus is on designing an energy-efficient routing algorithm for MANETs using the Ant Colony Optimization (ACO) approach.

## 1.2 Research Problem

The thesis proposes an energy-efficient routing algorithm for packet routing in MANETs. It takes the battery powers of the nodes in a MANET as the primary criteria for route selection, and optimizes the network performance by finding energy-efficient routes. The approach taken to design an energy-efficient routing algorithm is ACO (Ant Colony Optimization)-based. ACO is a learning algorithm, and improves performance as time passes. ACO stores information regarding past links and their worthiness, and uses that information to find an optimal route. Since energy-efficiency is in question here, the ACO-based algorithm proposed in this thesis concentrates on the battery powers of nodes in a MANET to find the optimal route.

## 1.3 Approach

We propose an ACO-EEAODR algorithm that takes a continuous learning approach wherein information about all the nodes in the network (such as routing information) is constantly updated. Based on this information, route discovery is done. Since this algorithm is based on

ACO, it follows the ACO fashion which includes defining an initial state, a final state, a State Transition Rule which leads to the decision of next hop for the data packet that is to be routed. The theory of ACO [3] has been applied to the routing of control packets that perform route discovery and store additional information about the individual nodes and the overall network, while performing route discovery. This information is used by later data transfer requests to propose a valid and optimal path faster. Even if a valid path is already stored, the states before and after the data transfer are recorded in order to determine the validity of the path.

## **1.3 Thesis Contributions:**

The thesis contributions are as follows:

- Design a new cost function for calculating the cost of a path, and an ACO-based algorithm for route selection
- Perform simulations with this new algorithm to validate the results.
- Using a network simulator named GloMoSim, programmed the algorithm in C language.

## **1.4 Thesis Outline**

This thesis is organized as follows:

### **Chapter 1: Introduction:**

This Chapter describes the research problem, its importance, as well as the motivation of this research.

### **Chapter 2: Related Work**

This Chapter describes some background information and related works.

### **Chapter 3: Methodology**

This Chapter is the core of our research work, where the EEAODR scheme and our ACO-based EEAODR scheme are presented.

#### **Chapter 4: Performance Evaluation**

This chapter presents our simulation results comparing EEAODR againstACO-EEAODR based on predefined performance metrics.

#### **Chapter 5: Conclusion**

This Chapter concludes our work and highlights few interesting questions that have arisen from the work carried in this thesis.

# Chapter 2

## Background and Related Work

### 2.1 Background

#### *2.1.1. Approaches to Achieve Energy-Efficient Routing in Wireless Ad Hoc Networks*

Apart from the proper selection of hardware, techniques for reducing the energy consumption in wireless ad hoc can be achieved by carefully designing all the layers in the system, namely (a) the physical layer - where an energy-efficient radio should be implemented in such a way so as to achieve a dynamic power management; (b) the Medium Access Control (MAC) layer - where the goal is to design protocols that minimize the time required by the radio to be powered; (c) the logical link control layer - where suitable error-control and flow control mechanisms should be implemented for controlling the conditions of the radio link so as to minimize the energy consumption; (d) the network and transport layer - where the rate of energy consumption is one of the important factors for the design of efficient energy-aware routing schemes. The latter is the focus of this Chapter.

Typically, the investigation of energy consumption in wireless ad hoc networks requires the study of coupling among the above layers [3]. The need for this study is because energy consumption does not occur only through transmission, but through processing as well [2]. This adds some complexity in the study of upper layer designs, particularly when energy efficiency is incorporated as a factor when performing the routing operations at the network layer [2,4], in combination with selecting the transmission power [5-8] (which itself affects the MAC and physical layers). The goal is to minimize the energy consumption at each node, which constitutes an important topic. Several design approaches, which can be referred to as energy-aware multi hop routing schemes, have been proposed, each of which addresses in its own fashion the trade-off between the transmission power, the energy expenditure, and the route selection process. The goal is to minimize the total energy consumption of the network subject to delay or throughput.

It also involves minimizing the delay (or maximizing the throughput) per joule of expended energy. These energy expenditures are affected by the design choices made at each of the upper layer levels mentioned above.

Few representative approaches are as follows: (a) approaches that favors the design of paths with minimal energy cost [9,10]; (b) approaches that promote the design of paths that avoid as much as possible the nodes with low remaining battery capacity [11]; (c) approaches that advocate the design of minimal energy cost paths composed of nodes with a battery level higher than a prescribed threshold [12]; and (d) approaches where the routing paths are selected according to the remaining battery capacity at each node, the sending rate per node, and the energy cost of hops [13]. In all these approaches, the following requirements should be met by when designing the routing protocol: (a) minimize the energy consumption in the network; (b) effectively deal with congestion avoidance; (c) successfully identify the packets from the source node and deliver them successfully to the destination; (d) ensure a proper balancing of load on individual nodes, i.e. deal effectively with the network management load.

In our work, we focus our attention on energy-aware routing schemes, particularly the use of Ant Colony Optimization (ACO)-based techniques for energy saving in the process of routing the data from one node to another in wireless ad hoc networks. Our goal is to introduce an enhancement to an existing energy-aware routing scheme (EEAODR) [14] using an ACO heuristic.

### ***2.1.2 Ant Colony Optimization Paradigm***

Studies of insect colonies such as bees or ants have revealed that these insects have some complex collective behavior and management structure [15,16] that resemble the properties of dynamic distributed systems. This has led to the introduction of the so-called Ant Colony Optimization (ACO) meta-heuristic [17], which can be defined as a common framework for a wide set of heuristic algorithms (so-called ACO algorithms) that can be applied to different types of problems, provided that an adequate model be defined. The ACO approach is based on iteratively constructing multiple solutions to an optimization problem, by progressively learning the search space and by using this knowledge to control the solution construction processes, which are driven by a stochastic decision policy. Eventually, this strategy leads to the design of optimal solutions to the targeted problem.



The growing interest in ACO algorithms has led to the development of many successful algorithms for discrete optimization problems [18], continuous optimization problems [19] - i.e. problems with continuous decision variables, dynamic optimization problems [20,21] - i.e. problems in which the search space changes over time, leading to potential changes in the definition of the problem instance as well as the quality of the solutions already found, to name a few. One of the prominent and widely used such algorithms is the Ant system [22] and its enhancements [23-26] which have proved suitable to design high performance routing protocols for communication networks [27-29], including wireless networks [30-34], and to find optimal solutions to other types of problems such as scheduling problems, assignment and layout problems, machine learning problems [35], to name a few. In this chapter, our focus is on investigating the use of ACO for designing energy-efficient routing protocols for mobile ad hoc networks.

### ***2.1.3 Why Using ACO-Based Approaches to Develop Energy-Efficient Routing Protocols for Mobile Ad Hoc Networks***

Mobile wireless ad hoc networks have gained an unprecedented popularity due to recent advances in wireless technologies and the suitability of such networks for use in situations where pre-deployed infrastructure does not exist. However, due to its rapidly changing and unpredictable nature, taking advantage of these features poses a number of challenges, few of which may lead to conflicting objectives. Examples of such challenges are: (a) achieving effective distribution and forwarding of data among nodes; (b) achieving load balancing among the nodes, (c) achieving minimal energy consumption; (d) achieving effective data routing; (e) ensuring congestion avoidance; (f) achieving minimum delay while ensuring a longer network lifetime, and (g) ensuring fault tolerance and achieving scalability.

A few of the existing protocols for energy-efficient routing were designed to effectively handle both the routing and the overall network management through routing. Also, their applications in real-life dynamic environments were not always guaranteed. Due to the intrinsic design features of ACO-based algorithms and their suitability in solving a variety of static, continuous, and dynamic optimization problems [31-34], ACO-based heuristic methods had appeared to be a natural way to attempt to build the best possible optimized solutions for the targeted challenges or combination of challenges. Focusing on energy consumption, the concept

of pheromone trails [31], which has led to the foundation of the design of ACO-based algorithms, can effectively be used to design energy-aware routing protocols that significantly reduce the energy consumption of wireless ad hoc networks.

Various energy-efficient routing protocols have been proposed in the literature, some of which have implemented routing algorithms using ACO framework. In this thesis, we investigate energy-aware routing schemes designed using ACO-based heuristics.

## 2.2 Related Work

Since the nodes in the MANETs are mobile, the network topology may change rapidly and unpredictably over time and because the network is decentralized which means that all network activity including route discovery, topology discovery, delivering messages and route maintenance must be executed by the nodes themselves, i.e. the nodes should be capable of performing the routing functionalities. A dynamic routing protocol is needed for these networks to function properly.

There are many types of routing protocols that exist for routing in ad hoc networks. The goal of these protocols is to establish and maintain reliable paths in a timely scheme to support robust and efficient sessions. They can be classified on the basis of Topology as Proactive, Reactive and Hybrid Protocols.

- Proactive routing also called table-driven routing maintains routing information independently of need for communication. They continuously evaluate routes. Thus, no latency in route discovery occurs in proactive protocols. They have large network capacity to keep current information.
- Unlike proactive routing, reactive routing also called on-demand routing do not make the nodes initiate a route discovery process until a route to a destination is required. This leads to higher latency than with proactive protocols, but lower overhead. Hybrid protocols combine both proactive and reactive strategies [2].
- Proactive approach uses a routing table which is maintained via periodic updates from all the other nodes in the network, irrespective of the fact that the network may not be active in terms of data traffic. While the on-demand approach, sends out requests for routes to

the destination only if the source node has data packets which are to be sent to the destination. That's why the proactive approach is more expensive in terms of energy consumption as compared to the reactive approach because proactive approach involves large routing overhead.

There are many power aware routing algorithms that choose several metrics for choosing the best routing algorithm. The most popular metric they choose is the shortest hop routing. And the algorithms which choose the shortest hop metric are Dynamic Source Routing (DSR) [36], Destination Sequenced Distance Vector (DSDV)[39], Temporally Ordered Routing Algorithm (TORA) [37], Wireless Routing Protocol (WRP) and the Ad-hoc On-demand Distance Vector (AODV) [38]. In these routing protocols DSDV is the table driven routing protocol and AODV, DSR and TORA are on demand approach.

A major drawback of existing MANET protocols is that they do not consider traffic load when making route selection. When they select the route based on the shortest number of hops, there could be the possibility that they use energy unevenly among the nodes, and thus it can cause some nodes to spend their whole energy earlier than expected.

In order to deal with resource management challenges in dynamic networks such as MANETs, the network should be equipped with new management and control strategies that can strengthen its adaptability to changes in traffic, topology, services, its flexibility in terms of transmission technology, topology characteristics, self-organizational behavior, while maintaining an acceptable performance level. Due to its nature-inspired features, and its similarity features with dynamic complex systems, the ACO framework [40] can provide a viable and effective alternative to help achieving the above-mentioned goals. Indeed, the central principle of ACO algorithms consists of designing lightweight agents that can continuously learn the current network status and use this knowledge to adapt the characteristics and decision policies inherent to network operations, in such a way to optimize the overall network performance and efficiency. In this regard, a few ACO parameters are of utmost importance since they have a direct impact on the quality of the obtained optimized solutions, namely, (a) the pheromone variables associated with each node in the network, (b) the decision policy parameters, and (c) the progressive learning phase of these parameters.

In the sequel, our focus is on applications of ACO-based heuristics [41] to routing problems

in mobile ad hoc networks, where the goal is on achieving the network's energy efficiency. Such algorithms are described, along with a flavor on how the aforementioned ACO parameters interact with each other to yield the optimized solutions. It should be emphasized that each ACO implementation involves two main phases [42]: the route discovery phase and the route maintenance phase, and uses some initial settings. Typically, mobile agents (so-called ants) are implemented as 'intelligent control packets'. They are allocated the task of discovering a suitable path for the routing of data between a source-destination node pair in the network. Additionally, they are also responsible for collecting useful information such as congestion status, amount of energy consumption, traffic load, etc., at each node along the selected routing path. This information is used to learn the network dynamics, and to continuously update the node's routing policy (indirectly embedded in the pheromone variables) in order to keep track of the changes in the network. This iterative process eventually leads to optimized solutions to the routing problem when the convergence occurs.

In [14], Dhurandher et al. investigated the problem of energy-constraint routing in ad hoc networks, and proposed an ant swarm-based algorithm that integrates both the power consumption at each node when routing a data packet and multi-path transmission features of artificial ants. In their proposed scheme, the energy usage is minimized by means of the path discovery process, inspired from the features of AntHocNet, and designed based on parameters such as route hop count and minimum battery energy remaining from the weakest node of the route. On the other hand, multi-path transmission is used to divert the packet flow in case of link failure (assumed to occur one at a time), leading to less number of dead nodes compared to the AntHocNet scheme. The merit of this protocol is that energy-awareness is used as a factor to increase the time that the protocol takes to judge the best possible route to be used for the data packets transmission. As pointed out by the authors, their proposed protocol was not tested in a real test-bed environment using in real-life scenario applications.

In [14], Dhurandher et al. proposed an ACO-based energy-efficient routing protocol for wireless ad hoc networks, which combines the effect of power consumption when transmitting a packet, the residual battery capacity of a node, and the multi-path transmission properties of artificial ant swarms. In their scheme, the path discovery phase is inspired from AntHocNet, but with distinct functionality in the sense that the routes are maintained through new pheromone reinforcement and evaporation techniques, leading to the use of multi-path transmission through

the “good routes” only rather than all the possible paths. Even though this scheme showed good promises, the effectiveness of the proposed scheme was not tested in real test-bed using practical scenarios.

In [42], Kim et al. investigated the problem of designing an energy saving routing scheme for energy-efficient networks. The problem was formulated as an energy-consumption minimized network (EMN) optimization problem. It was based on the concept of traffic centrality of a node, defined as a measure involving the traffic volume (in bytes) on a link and the density of traffic carried on that link. The optimization problem is then solved using the ACO method by centralizing traffic flow to denser links only. The optimized energy efficiency level produced by the proposed ACO-Energy Saving Routing (A-ESR) algorithm is dependent on a controlling factor that was used to weigh the traffic centrality. However, the authors neglected to indicate how the value of this factor can be allocated in a dynamic manner.

In [41], Ren et al. presented an ACO-based energy-aware routing for ad hoc networks (so-called ABEAR). Their proposed scheme introduces congestion metric and uses it along with a combination of reactive route setup procedure and proactive neighbor maintenance procedure in its routing phase to find suitable paths for transferring the data from source to destination. In this process, the link quality, remaining energy at each node, and pheromone values are integrated as design variables in the ACO approach when performing the routing computation, with the goal to reduce the network lifetime.

In [46], Dhurandher et al. proposed an ACO-based energy-aware routing protocol (so-called A-CMMBCR) for mobile ad hoc networks, which is an enhancement to the Conditional Max-Min Battery Capacity Routing (CMMBCR) protocol [43]. The ACO design involves a combination of two routing schemes, the Minimum Transmission Power Routing (MTPR) and the MMBCR schemes. This combination results in an optimized scheme that contributes greatly in minimizing the total energy consumed in the network (as compared with those benchmarking schemes). It also ensures that routes are not overloaded due to traffic and backup paths are available in case the routes break due to the mobility of the nodes. However, the fault tolerance aspect was not addressed in this design.

In [44], Roy et al. presented a routing algorithm with a combination of reactive as well as proactive approach. The author talks about using QoS strategy for routing to maximize the utilization of network resources. The hybrid approach consisting of reactive and proactive

approach is taken to cater to the changing network topology in a MANET. The algorithm is based off of Ant Colony Optimization (ACO) algorithm. Although the algorithm is well explained and seems promising, it would have been desirable to also discuss multicast routing, as multicast requires more bandwidth and a proactive approach.

In [45], Verma et al. discusses the foraging behaviour of ants in ACO. A variation of the ACO algorithm, called AntNet, is used to carry out simulation and analyse performance metrics and compare against AODV. The results show improved packet delivery ratio and lesser time delay. However, this algorithm could have held more significance if other variations of ACO were also discussed.

In [36], Dhurandher et al. proposed an energy-efficient ad hoc on-demand routing protocol (so-called EEAO DR), that can be used to balance the energy load among nodes so that a minimum energy level is maintained among nodes and the network life is increased. This is realized by implementing a mechanism for distributing the energy consumption in the network. In this scheme, several workload and node requirement parameters were introduced, which yielded the optimal network performance, but the way these parameters were maintained or updated was not disclosed.

In this thesis, an ACO-based approach to the EEAO DR solution is investigated in-depth, with the goal that the obtained performance will be better than that obtained using EEAO DR.

# Chapter 3

## Methodologies

This chapter covers the main contributions of the thesis. It includes the description of the Energy Efficient Ad Hoc On Demand Routing (EEAODR) [36] scheme as well as our proposed ACO-EEAODR scheme.

### *3.1. EEAODR Algorithm Design*

In EEAODR, the route selection is performed by using control packets that carry information about the needed route. The algorithm takes into account the battery powers of each device in the network and tries to find the path that utilizes the minimum battery power of each node. There are different factors that affect the battery power utilized during the network transmission, few of which are: the packet size, the hop count, the length to the link between two nodes. The final selected routing path (so-called optimal path) i.e. the path that is most energy-efficient, is calculated based on these parameters.

Conventional algorithms tend to focus more on shorter paths and, hence, keep selecting same nodes over a period of time. This causes some nodes to lose their entire battery power, which may lead to network failure. However, EEAODR takes an energy-efficient approach. Instead of considering a shorter path, the algorithm considers the path which is more energy-efficient i.e. leaves each node in the network with maximum possible energy, whether that particular path is time-efficient or short. The number of hops is the second most important criterion for path selection. The reason for using the number of hops is that less hops will ensure a lesser consumption of energy in overall data transfer. Finally, time is also another important parameter. There have been algorithms proposed that save energy by minimizing the size of control packets. However, these types of algorithms require more time for path selection. Therefore, there are different factors to which a network administrator can assign weights based

on preference or a desired throughput. Equation 1 depicts all the above-mentioned parameters in a mathematical relation [9]:

$$\text{Cost} = \sigma \times \text{time} + \mu \times \frac{1}{\text{minimum battery power of node in route}} + \tau \times \frac{1}{\text{number of hops}} \quad (\text{Eq. 1})$$

The EEAODR process starts by a node requesting data transfer. All previously saved valid paths (i.e. paths for which all nodes are still alive) are searched to check if there is a route to the desired destination. If no valid path is found, a Route REQuest (RREQ) is sent out to all neighbouring nodes. Each node that receives an RREQ packet checks whether it is a genuine packet or a packet that has already been received in the past (i.e. duplicate). The node, then, checks its own route table to search for a valid path to the desired destination. In the case where a valid path is found, the same RREQ is routed to that valid path. Otherwise, a new RREQ is formed and broadcasted again. Finally, when the destination receives an RREQ, it waits for a certain period of time, known as RT, before it analyzes all the paths received through RREQ, select the optimal path and send a Route REPLY (RREP) packet to that path. In addition, it also stores other paths that were less optimal than the most optimal path. This is done for future use so that energy is not consumed in path selection every single time. Finally, data is transmitted using that path through which the destination sent RREP.

The pseudo-code of the EEAODR algorithm was described in [9] as follows:

```

A node wants to transmit data packet:
If (Path to destination is found in Route Table and it has not expired)
    Jump to L4.
Else
    // Find the path
    L1: Broadcast RREQ to neighbor nodes
    Valid RREQs are received by nodes.
    If RREQ receiving node is the destination
        If RREP has been sent
            It is a duplicate RREQ, reject it.
        Else
            If it is the first RREQ from the source
                Setup the Timer for RT duration
            Endif
            Do the corresponding entry in Seen Table
            L2: Wait for new valid RREQ until RT times out.
        Endif

```



```

    Else node is intermediate one
        Do the corresponding Entry in Seen Table.
        Jump to L1
    Endif

At destination node:
If Time Interval RT passes
    Calculate best path with lowest cost.
    Take some backup paths.
    Send RREP
Endif

At source node:
If source receives RREP within time out interval
    Source updates its routing table with those
    selected paths
    L4: Source transmits data through the selected path
Else
    If some valid back-up path which has not timed
    out
        Select that path and jump to L4
    Else
        Send RREQ again or Logout
    Endif
Endif

```

Figure 1: EEAODR pseudo-code

### 3.2 ACO-EEAODR Algorithm Design

#### 3.2.1 Steps for Solving a Problem Using ACO Approach

The steps for solving a problem using the ACO paradigm are as follows:

- Problem representation: the problem has to be represented as a graph, the solution being a minimum cost path
- Evaluation function: this is a problem-specific function that evaluates the solutions.
- Local heuristic: this is the information that guides the ants' search process e.g. the gain in the value of the evaluation function.
- Pheromone update rule(s): these rules are meant to determine how much pheromone content should be deposited and on what events.

- Probabilistic transition rule: this rule utilizes both the heuristic value and the pheromone amounts so that the ants decide their transitions from one node of the graph to the next. We have used the AS algorithm to implement our ACO-based EEAODR. In doing so, the Probabilistic Transition Rule was implemented based on the following formula [28]:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k \text{ is in allowed}_k} [\tau_{ik}(t)]^\alpha \cdot [\eta_{ik}]^\beta} & \text{if } j \text{ is in allowed}_k \\ 0 & \text{otherwise} \end{cases} \quad (\text{Eq. 2})$$

where  $allowed_k$  represents the valid paths that are stored in the node's routing table,  $\alpha$  and  $\beta$  are parameters that control the relative importance of pheromone versus distance,  $\tau_{ij}$  is the pheromone content between link (i,j) and  $\eta_{ij}$  is the distance between nodes i and j

### 3.2.2 ACO-Based EEAODR

We follow the above-mentioned steps to design our ACO-based EEAODR algorithm as follows:

**3.2.2.1 Problem presentation:** Since the algorithm deals with finding an optimal route for packet routing in Mobile Ad-hoc Networks, our graph is the entire terrain of the network. The network contains various mobile devices (nodes) which can be considered as the vertices in the graph. Each link between any two nodes of the network is considered as an edge. Therefore, a path from node A to node B will be denoted as  $P_{AB} = (A,B)$ . Since the amount of energy consumed by transmitting data from node A to node B is the same the amount of energy consumed while transmitting data from node B to node A,  $P_{AB} = P_{BA}$ . This indicates that the graph is an un-directed graph and all edges are bi-directional. The desired solution of the graph is a set of links (e.g. (S,a), (a,b), (b,x),.....(y,D)) for source-destination pair (S,D), where S is the source and D is the destination. The solution must be such that it leaves each vertex with an amount of energy that is sufficient to stay in the network. The solution should also contain the lowest HopCount.

**3.2.2.2 Evaluation function:** Evaluation function is the formula that the node applies to calculate the most energy-efficient path. It calculates the cost of each path by considering the pheromone deposition (to be described later in this section) on each node in that path.

Quite clearly, the path having the least cost is selected as the best path. Equation 2 is the mathematical formula for the evaluation function used in the ACO-EEAODR:

$$COST = [\alpha * (hopCount)] + \left[ (1 - \alpha) * \frac{1}{\sum(node.power)} \right] \quad (Eq. 3)$$

where  $\alpha$  is in the interval ]0,1[. For our simulations, we have used  $\alpha = 0.3$

In the above formula (Equation 2), 70% significance is given to the remaining battery powers of all the nodes in a certain path, and 30% significance is given to the number of hops involved in that path. The more the sum of battery powers  $[S(node.power)]$  the smaller will be its reciprocal, resulting in a lower cost. On the other hand, the lower the *hopCount*, the lower will be the cost of that path.

**3.2.2.3 Local heuristic:** This is the information that forms the basis of the search process. Local heuristic is usually the parameter that needs to be optimized. Since this algorithm is an energy-efficient algorithm, local heuristic is base on the remaining battery power of a node. This means that when a control packet transitions from one node to another node, the factor that governs the decision of which node to jumps to, is battery power. Based on the ACO-EEAODR, a node will prefer hopping to a node with a higher battery power, rather than a node with a shorter path length or a node that involves less time consumption in transfer.

**3.2.2.4 Pheromone update rule(s):** The heart of any ACO-based algorithm is pheromone deposition which guides the later coming ants (nodes in this case). In ACO-EEAODR, the source broadcasts RREQs (control packets) through the network, and waits for RREPs. Once all RREQs are received at the destination (or RT time is up), evaluation function is used to perform path selection and RREP is sent through the best path. This is the time when pheromone deposition takes place, indicating the ‘goodness’ of the path. The rule that is used to update pheromone values is as follows:

$$Pher_{new} = Pher_{old} + \left[ \frac{1}{maxPower - node.power} \right] \quad (Eq. 4)$$

where  $Pher_{new}$  is the new pheromone value to be assigned to the node,  $Pher_{old}$  is the current pheromone value of the node,  $maxPower$  is the maximum battery level a mobile device can acquire and  $node.power$  is the current battery level of the mobile device.

As it can be seen from the above equation, the Pheromone Update Rule is dependent upon the remaining battery power ( $node.power$ ) of the node at which pheromone content needs to be updated. The higher the remaining battery power will be, the greater will be the result of the fraction, resulting in a bigger number being added to the existing pheromone value. Due to this method of pheromone update, preference is always given to the nodes that contain more battery power. The node, whose battery power is low, continues to possess lower pheromone value than the other nodes. Just as the case is with real ants, where pheromone content keeps evaporating slowly with time, the battery power of a node also keeps deprecating with time due to several other operations taking place in the node (as they are electronic mobile devices). Therefore, ACO-EEAODR accounts for this by including pheromone evaporating as part of Pheromone Update Rule. In order to avoid added complexity, pheromone evaporating has been kept uniform. On a one-minute timeout, the pheromone value of each node is decreased by 2%. This is shown in Equation 4 in the form of a mathematical formula:

$$Pher_{new} = Pher_{old} - [(0.02) * Pher_{old}] \quad (\text{Eq. 5})$$

**3.2.2.5 Probabilistic transition rule:** As mentioned above, the Probabilistic Transition Rule uses the local heuristic and pheromone to decide the next hop in the network. Since ACO-EEAODR determines the entire route before data transfer, the next hop is usually known, and is not probabilistic. However, there are 2 scenarios where the Probabilistic Transition Rule has been applied in this algorithm. First, when source broadcasts RREQ packets, it broadcasts to its neighbors only. These neighbors are defined through a Probabilistic Transition Rule. Any node that is in close vicinity to the node from where transition is to be made, and contains a valid link to the node from where transition needs to be made, is considered as a neighbor. Hence, the state of RREQ packet can be transitioned to that node. This is the mechanism by which RREQs reach the destination. Second, when an RREP is on its way to the source and a node that is part of the route, dies. In this

scenario, RREP looks for all the nodes that are in neighborhood of the node where RREP resides. The Probabilistic Transition Rule here is based on the following 2 parts:

- Node must contain a valid path to source in its route table
- Node must have the highest pheromone value out of all nodes satisfying part 1)

Based on the above Transition Rule, the RREP is forwarded to the selected node, and finally reaches the source node.

### ***3.2.3 Main Operations of the ACO-EEAODR***

Since ACO-EEAODR is an on-demand protocol, the routing process starts when a data transmission request is initiated. The process ends when either data has been successfully transmitted or there is no possible route available from source to destination. During the idle time, only listening happens, i.e. the protocol listens to all the nodes for any data transmission requests.

The ACO-EEAODR algorithm has been broken down into different steps below, so as to provide a basic understanding of the mechanism of this algorithm. This will be followed by detailed pseudo-code of all these steps and other necessary information which the algorithm uses.

If looked at from a broad perspective, the algorithm could be broken down into the following chunks of processes:

#### ***1. Request for data transmission from source to destination***

Any node can initiate a request to transmit data to any other node in the ad-hoc network. Since all nodes are pre-programmed with the ACO-EEAODR algorithm, whenever a node needs to send data to another node, route selection starts, after which the data could be sent. Since ACO-EEAODR allows each node to store valid back-up paths that have been used in the past, one of these paths can possibly be used to send data.

## ***2. Decision of using either an available path or searching for one***

Decision on using an available path or searching for a new path has to be made by the source node. The algorithm gives precedence to previously stored paths, if they are valid. Valid backup path save a lot of energy since they avoid the need of sending request packets, waiting for reply packets and then sending reply packets, It saves time as well since data transfer can take place without route discovery. If the node does not find any valid paths in its routing table for the intended destination, it sends a request packet, called RREQ (Route REQuest) packet to its immediate neighbour nodes requesting data transmission to the destination node. RREQs are supposed to return back in the form of RREP (Route REPly) with the most optimal path stored in them, hence giving the source the path to send data.

## ***3. Broadcasting of request packets 'RREQ' to neighbors of ad-hoc network***

Broadcasting of request packets 'RREQ' to neighbors of ad-hoc network. If there are no valid paths in the routing table, request packets i.e. RREQ packets are sent requesting data transfer to destination. RREQs are in a specified format explained in the paper and contain the IP address of the source, the destination and the IP address of the node they are being sent to (the intermediate node). Each node that receives a particular RREQ appends its address in an array called Path[]. Therefore, when RREQ reaches the destination, it has the entire path through which it has traversed from source to the destination, in the array Path[].

## ***4. Reaching of RREQ at destination and processing of each RREQ.***

The destination node is programmed to perform in a manner different than in usual routing algorithms. As soon as the destination node receives the first RREQ packet, it does not perform any action. Rather, it initializes a timer called RT. This is the time the destination waits for other RREQ packets that might arrive travelling through a different

path. The destination stores all the RREQs it receives within RT for later processing. Once RT has elapsed, the destination rejects any RREQs after that and starts processing the RREQs received within RT.

##### **5. *Calculation of optimal path after time-out has occurred.***

###### **a) Calculating the path that is most optimal**

Now that the destination has all RREQs, and is ready to process them, it will calculate the best path. Each RREQ holds the path through which it has traversed. However, since we are dealing with ACO-EEAODR, which is an Energy-Efficient algorithm, shortest path is not our priority. This is where energy-efficiency comes into play. Using the Evaluation Function, the algorithm calculates the cost of each path. The algorithm will also determine what will be the amount of energies left over with all the intermediate nodes after data has been transmitted. Hence, it will select a path that keeps the entire network in the most stable state; although this may take up slightly more energy overall or it may not be the shortest path. The key here is that energy required to transmit data is evenly distributed among the nodes of the network, not putting extensive load on any single node.

###### **b) Calculating backup paths**

Once the optimal path has been found, the destination also calculates the energy values for other paths (other RREQ packets) so that back-up paths could be suggested, in case the optimal path goes down (due to a node dieing) or a node leaves the network.

##### **6. *Sending reply packet 'RREP' back through the optimal path.***

Now, as the destination has finished with all the calculations and is ready to reply back, it constructs an RREP packet. The RREP contains the address of the destination itself, address of

the intended source (the sender of RREQ packet), and an array of paths  $Path [ ] [ ]$ . Each element of this array is a valid path from source to destination; the first one being the optimal path, and the rest being back-up paths. While RREPs traverse through the optimal path, they update the pheromone value of each node in the path based on the formula in Pheromone Update Rule. If the RREP reaches the source before timeout, it is accepted and data is transmitted using the path specified by RREP. Otherwise, the entire process of route selection starts again.

## **7. *Data transmission***

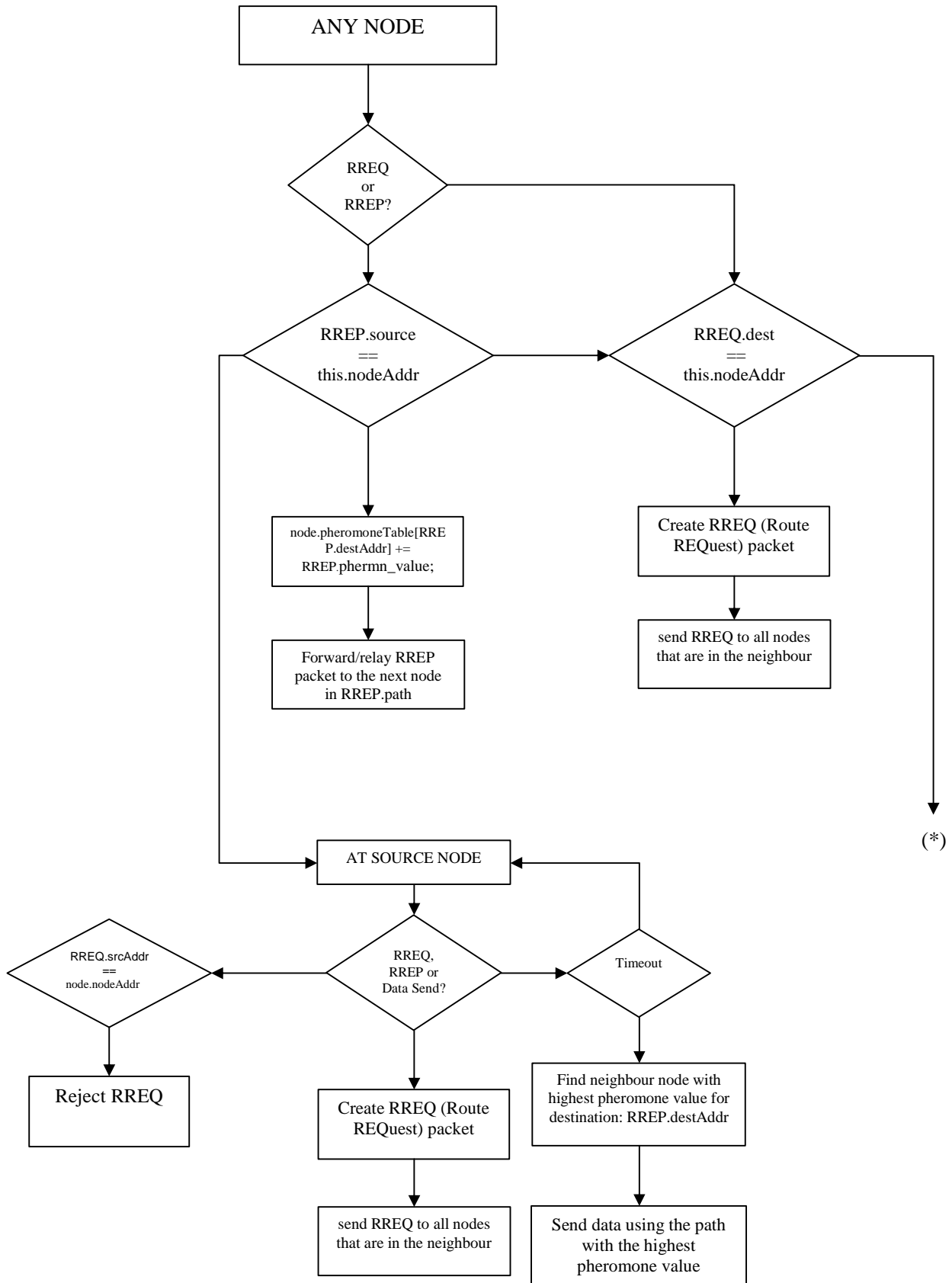
### **a) Storing optimal path and backup paths in local route tables**

As soon as the RREP is routed and reaches the source, the source updates its routing table with the optimal and back-up paths for future use. This consists of a timer as well. Each path in the local routing table of a particular node stays valid for a certain time period, monitored by a timer. Once the timer for a path has elapsed, that path is no longer a valid path, and is discarded. The concept of a timer is introduced here due to the reason that we are working with ad-hoc networks. Not all nodes are present in the network at all times. Therefore, it is important to notice whether a path is still valid or not (i.e. all its intermediate nodes are still alive or not).

### **b) Using the optimal path to send actual Data Packets**

Last step in this process would be the actual data transmission using the optimal path. This is the point where the reason for storing back-up paths becomes visible. If the source tries to send data through the optimal path, but finds that this path is no longer valid due to a node die-out or any other reason, the source may not have to `startover` sending RREQs again. It can just pick the first back-up path stored in its local routing table, verify its validity and start sending data. The next two pages show ACO-EEAODR flowchart in Fig. 2





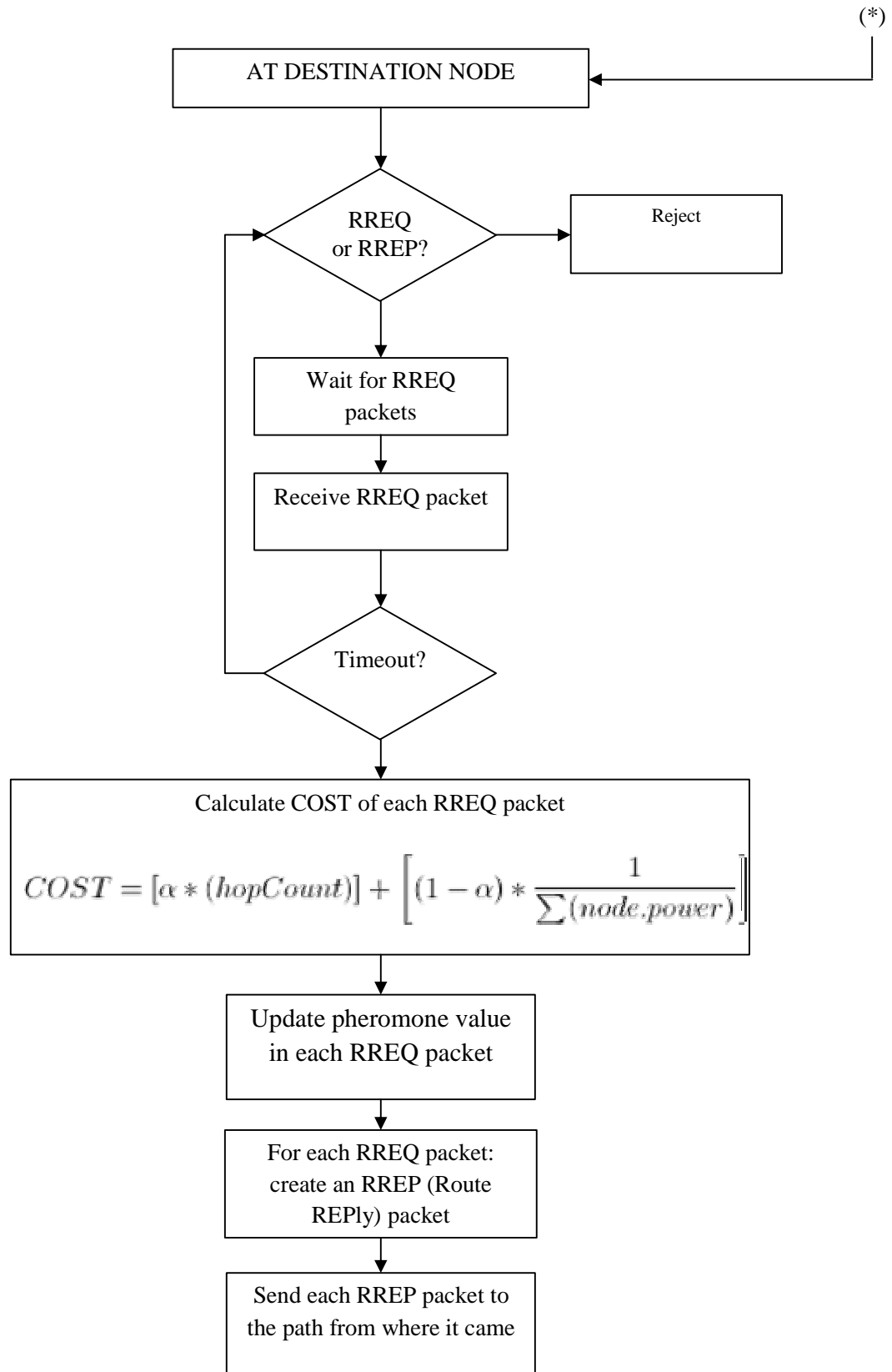


Figure 2: The ACO-EEAODR flowchart.

The pseudo-code of the route selection in ACO-EEAODR is depicted in Fig. 3:

**At the source node:**

```
Data_transmit = true // Need for data transmission occurred.
For i=0 to route_table.length
    if pair(source, destination) = pair[i]
        isPresent = true
    else
        isPresent = false

if isPresent
    Path[i].Load path(Path[]) // load the path from route table into the array Path[]
    Send data using Path[]
    Update route_table
    Break; // Break the loop

else
    For i=0 to neighbour_list.length
        sendPacket(RREQ, neighbor[i])

waitTimer.start() // wait for a certain time for RREP
on waitTimer.timeOut(goto("At Source"), data_transmit = false)
```

**At intermediate nodes (including Destination Node):**

```
Event(RREQ_RECIEVED) triggered
if RREQ.DESTINATION_IP == THIS.IP_ADDRESS
    RREQ.Path[Path.length] = THIS.IP_ADDRESS
    rreqList.add(RREQ)
    entryTimeTimer.start()

Event(RREQ_RECIEVED) triggered
if RREQ.DESTINATION_IP == THIS.IP_ADDRESS

    if entryTimeTimer.timeOut = false
        RREQ.Path[Path.length] = THIS.IP_ADDRESS
        rreqList.add(RREQ)
    else
        RREQ.discard() // discard if timeout occurred

// if this is an intermediate node
Else
    RREQ.Path[Path.length] = THIS.IP_ADDRESS
    For i=0 to route_table.length
        if pair(source, destination) = pair[i]
            isPresent = true
        else
```

```

    isPresent = false

    if(isPresent)
        create new RREP
        RREP.Path = path[i]
        RREP.Source = RREQ.Source
        RREP.Destination = RREQ.Destination
        RREP.hopCount = path[i].hopCount
        sendPacket(RREP,source)

    else
        For i=0 to neighbour_list.length
            sendPacket(RREQ,neighbor[i])
        waitTimer.start()

At the destination node:

    For i=0 to rreqList.length
        rreqList[i].Path[rreqList[i].Path.length] = THIS.IP_ADDRESS

    // customEntryTime,  $\mu$ ,  $\gamma$  are values set by Network Administrator
    bestPath = calculateOptimalPath(rreqList, customEntryTime,  $\mu$ ,  $\gamma$ )
    backupPaths[] = calculateBackupPaths(rreqList,customEntryTime)

    create new RREP
    RREP.source = rreqList[0].source
    RREP.destination = THIS.IP_ADDRESS
    RREP.path = bestPath
    RREP.backups = backupPaths
    sendPacket(RREP,source)

    calculateOptimalPath(rreqList, customEntryTime,  $\mu$ ,  $\gamma$ )
    {
        For i=0 to rreqList.length
            Cost[i] = _
            (customEntryTime*rreqList[i].time) + _
            ( $\mu$  * rreqList[i].minimumPower) + _
            ( $\gamma$ * rreqList[i].Path.length)

            optimumIndex = Cost.least()
            return rreqList[optimumIndex]
    }

```

Figure 3: ACO-EEAODR pseudo-code

# Chapter 4

## Performance Evaluation

This Chapter describes the performance evaluation of the EEAODR and ACO-EEAODR algorithms using various performance metrics and simulation settings. The goal is to compare ACO-EEAODR against EEAODR.

### 4.1 Simulation Tool

Global Mobile Information System Simulator (GloMoSim) is a scalable network protocol simulation software that simulates wireless network systems. GloMoSim uses a parallel discrete-event simulation capability provided by Parsec [15]. This simulation tool is designed to be extensible. It provides a high fidelity simulation results for wireless communication with detailed result sets for each layer in the network [15]. We use the GloMoSim 2.03 version to run our simulations using Parsec with RedHat-7.2 configuration as a compiler based on a 32-bit Linux operating system.

### 4.2 Performance Metrics

The following performance metrics are used for the evaluation of the above-mentioned algorithms.

- *Energy consumed in path selection:* The total energy consumed is the battery power that was utilised for each node for a particular data transfer. This can be found out by taking a difference of the initial and final battery powers of each node. For ‘n’ nodes, following is the formula used to calculate the total energy consumed:

$$E_{consumed} = \sum_{x=1}^n (E_{ix} - E_{fx}) \quad (\text{Eq. 6})$$

where  $E_{ix}$  = energy of node x before transmitting data and

$E_{fx}$  = energy of node x after transmitting data

The energy consumed has been graphed against varying number of nodes as well as against varying simulation time, to study its impact.

- *Hop count*: Hop count is the total number of nodes, except for source and destination that were part of a particular route. A route with a smaller hop count means it traversed through fewer nodes (although that does not guarantee a shorter path, since the distances between the nodes could be high). Hop count helps gauge which path could consume more energy. If the hop count is low, then fewer nodes had to transmit data, resulting in a lower consumption of energy, and vice versa.
- *Network lifetime*: Network life time is described as the time when the first node in a Mobile Ad-hoc Network dies i.e. its battery power is exhausted. Network lifetime is a very important performance metric, as it measures the reliability of a network. Even if one node in a network dies, it could cause a significant damage to the overall network performance. The reason being, that node could be part of multiple routes that are stored in different route tables. In our simulation we have graphed network lifetime against simulation time, to see at what point the first node in the network dies out.
- *Simulation time*: Although our main focus was on energy-efficiency, it is important to analyse how fast ACO-EEAODR algorithm is able to transfer data. Therefore, keeping a fixed number of data packets, and the same source-destination pair, time taken to transfer one set of data, was plotted against the total number of nodes.

The results of all above performance metrics are discussed in the next section.

## 4.3 Simulation Parameters

Table 1 outlines the simulation parameters used throughout the simulations:

Parameter	Value
Terrain dimension	2000 x 2000
Node placement	Uniform
MAC Protocol	IEEE 802.11
Traffic type	CBR
Packet size	512 bytes
Network Layer routing	AODV

Table 1: Simulation Parameters

## 4.4 Simulation Results

In this simulation, we vary the simulation time, and we analyze the energy consumed in path selection when the ACO-EEAODR scheme is used compared to when the EEAODR is deployed. The results are captured in Table 2 and depicted in Fig.4.

Simulation Time (secs)	Energy Consumed (units)	
	ACO-based solution	EEAODR
30 seconds	486	586
60 seconds	684	982
90 seconds	882	1378
120 seconds	1080	1774
150 seconds	1278	2170
180 seconds	1476	2556
210 seconds	1674	2962
240 seconds	1872.5	3358

Table 2: Energy consumed in path selection vs. simulation time when using ACO-EEAODR and EEAODR

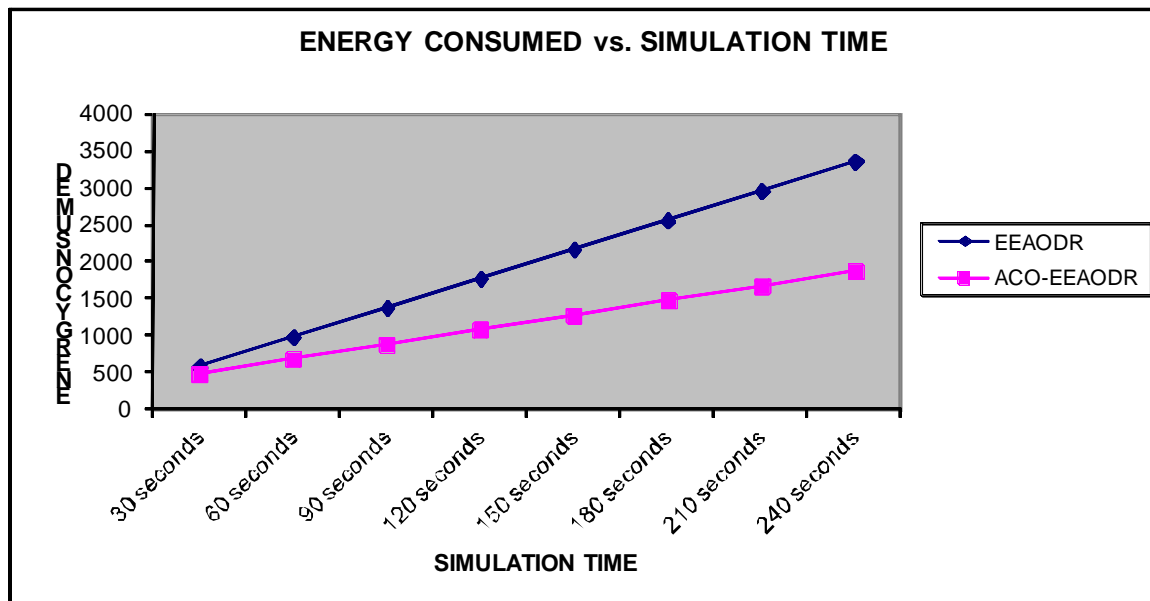


Figure 4: Energy consumed in path selection vs. simulation time when using ACO-EEAODR and EEAODR.

In Fig. 4, it can be observed that the ACO-based solution uses much less energy than the original EEAODR. This happens because ACO continuously learns paths based on energy. Therefore, a path that leaves the nodes in the network with the most amount of energy (battery level) is selected each time a data transfer request is made. This results in less energy consumption overall, and continues to increase the life of the network.

Now, we examine the energy consumed in path selection when the number of nodes varies. The results are captured in Table 3, and depicted in Fig. 5.

Number of nodes	Energy Consumed (units)	
	ACO-based solution	EEAODR
20 nodes	1550.5	966
60 nodes	275.5	2153.5
100 nodes	2480	2763.5

Table 3: Energy consumed in path selection for ACO-EEAODR vs. EEAODR



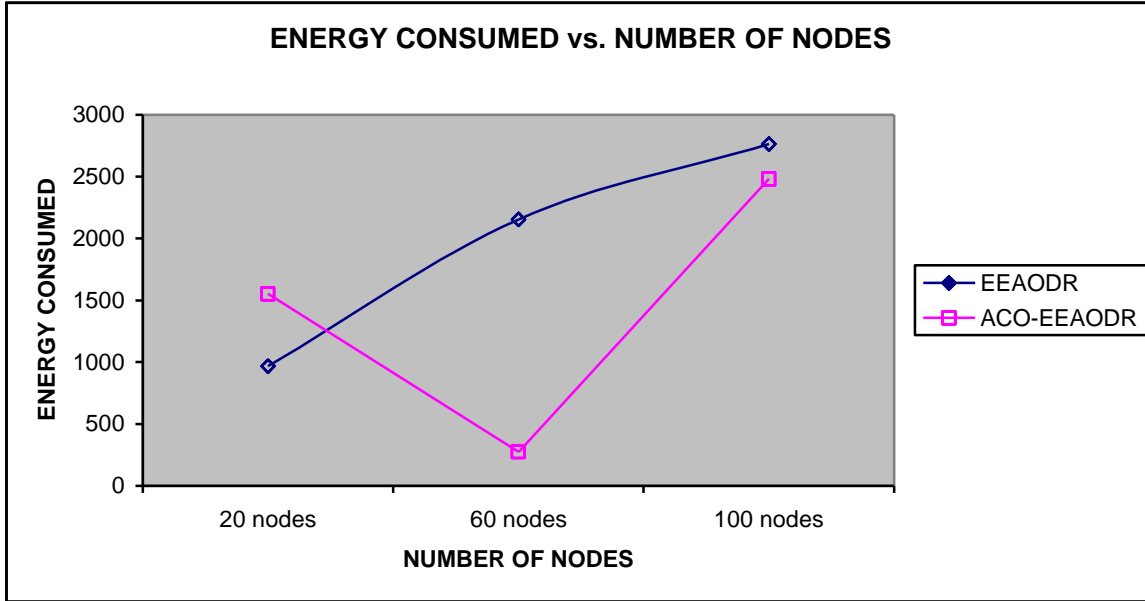


Figure 5: Energy consumed in path selection for ACO-EEAODR vs. EEAODR

In Fig. 5, it can be observed that when the number of nodes increases, the route selection becomes more complicated and more possibilities come into existence for a single source-destination pair. In such a scenario, it is important that the consumption of energy be monitored with increasing network size. The above graph shows how the energy consumed is affected as the number of nodes increase. In the beginning, ACO based algorithm seems to use more energy than EEAODR, but as the number of nodes increase, more and more paths have been stored in route tables, and are being used to transmit data. As a result, energy is saved since lesser route discovery operations need to be performed. High energy consumption is explained by the overhead involved in the ACO based solution in the initial phases. Since this is a learning algorithm, time overhead like Response Time and energy overhead like sending and receiving control packets is involved. These packets are much smaller in size than the data packets but, they do contain data. Therefore, they use up some battery power of the nodes in a MANET and cause more energy consumption than EEAODR. But as the algorithm continues to run in the network, learning gets better, more paths are memorized and stored and therefore, less energy is consumed in the long run. Source nodes continue to have more and more valid paths stored in the local tables and the overhead involved with control packets is cut down.

Next, we analyze the simulation time taken by changing the number of nodes in the network. The results are captured in Table 4 and depicted in Figure 6.

Number of nodes	Simulation Time (sec)	
	ACO-based solution	EEAODR
<b>20 nodes</b>	10200	4440
<b>40 nodes</b>	7040	3710
<b>60 nodes</b>	6300	2590
<b>80 nodes</b>	5550	1760
<b>100 nodes</b>	4390	1660
<b>120 nodes</b>	2530	1490
<b>140 nodes</b>	2280	1480
<b>160 nodes</b>	2010	1258

Table 4: Simulation Time vs. Number of Nodes for EEAODR & ACO-EEAODR

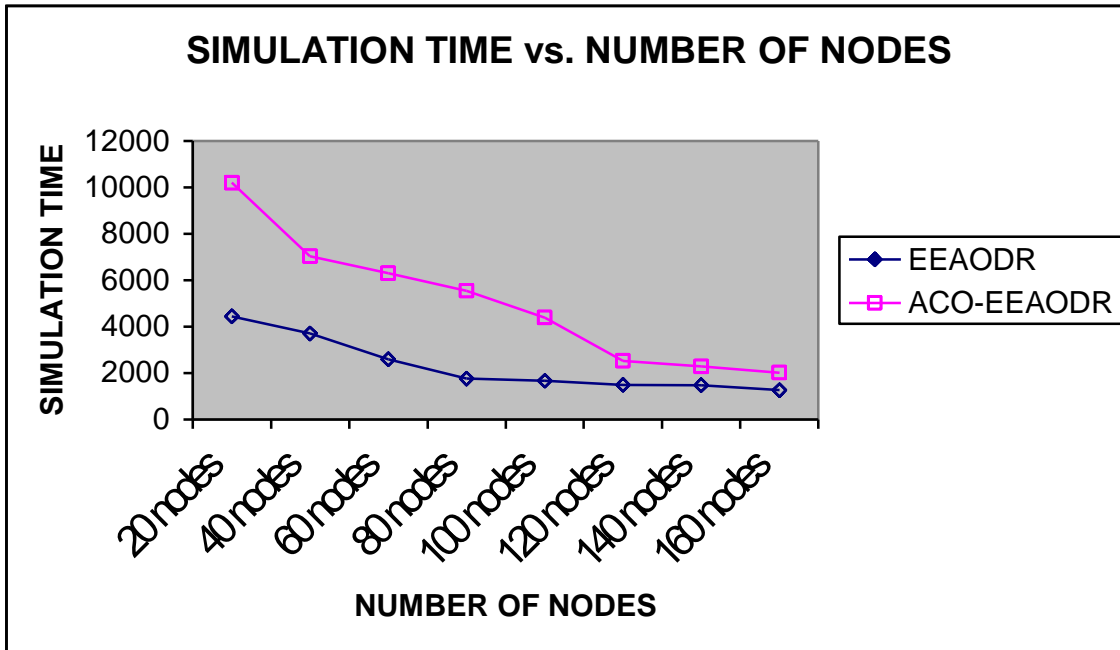


Figure 6: Simulation Time vs. Number of Nodes for EEAODR & ACO-EEAODR

Although the ACO based algorithm is an energy efficient algorithm, it is also important to note the time it takes for data transfer along with energy consumption. We do not want an algorithm that utilizes less energy but trades off time for energy.

As it can be seen in Figure 6, ACO based solution is not as time efficient as EEAODR. Due to the fashion in which the ACO algorithm works, it is bound to consume more time than usual. There are certain wait times involved, such as the time destination node waits for more and more request packets, the time that source waits before starting to calculate the best path. These overheads cause this algorithm to consume more time. However, what is interesting to note in the graph above, is that the difference in time consumption keeps decreasing between the ACO based solution and EEAODR. This shows a good sign for larger networks, that although the time consumption is more in ACO algorithm, it is not as worse as expected in the beginning, and depending upon the size of the network, ACO may be made as the choice.

Next, we analyse the effect of Response Time on the performance of the algorithm by plotting it against varying Simulation Time. When a destination receives the first RREQ, it waits for more RREQs to arrive before sending an RREP. This time, for which the destination waits after receiving the first RREQ, is called Response Time, denoted by RT. The results of simulation are captured in Table 5 and depicted in Figure 7

Simulation Time	Response Time (ms)	
	ACO-based solution	EEAODR
<b>30 seconds</b>	500	586
<b>60 seconds</b>	1100	982
<b>90 seconds</b>	1270	1378
<b>120 seconds</b>	1599	1774
<b>150 seconds</b>	1897	2170
<b>180 seconds</b>	2360	2556
<b>210 seconds</b>	2600	2962
<b>240 seconds</b>	2900	3358

Table 5: Response Time vs. Simulation Time for EEAODR & ACO-EEAODR

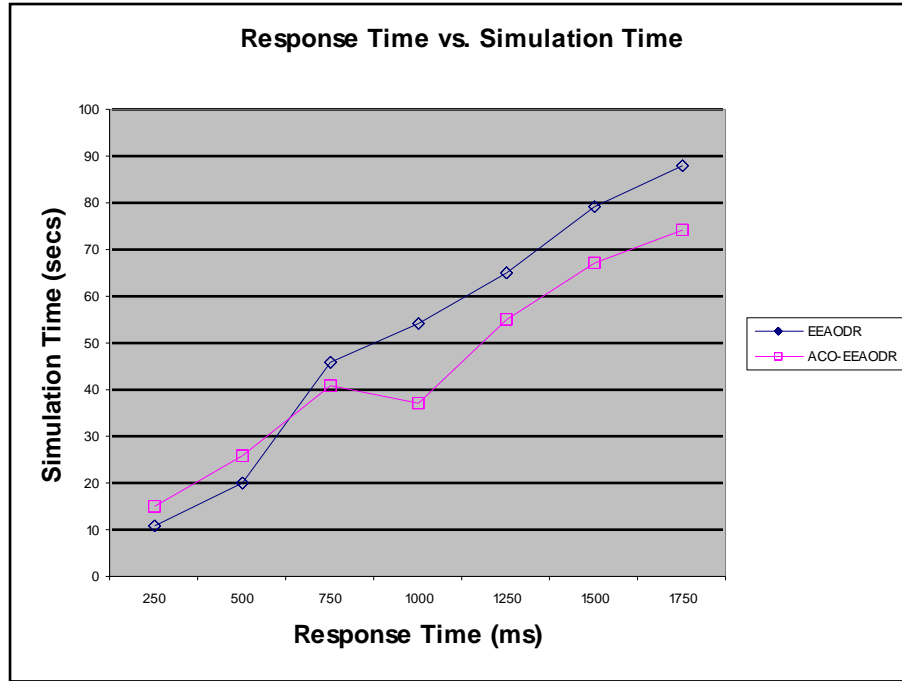


Figure 7: Response Time vs. Simulation Time for EEAODR & ACO-EEAODR

The Response Time can be calibrated and adjusted by analyzing the statistics, to see which number best suits a particular network. The above graph shows the response times that were found to be optimal for achieving a certain throughput for data.

From Table 5 and Figure 7, it is clear that with the original EEAODR, the response time had to be increased in a linear fashion. Whereas in the ACO-EEAODR, Response Time is less as the simulation time increases. This happens because everytime a route selection operation is carried out, that particular route is saved in the Route Table (for later use). Not only that, during route selection, other paths that are not as energy-efficient as the best path, but still consume less energy, are saved as 'backup paths' in the respective node's Route Table. This eliminates the need for route selection to be performed with each and every data transfer request. One of the valid backup paths could easily be used, if it is available. If a backup path is used for data transfer, and no route selection operation is performed, very clearly there will be no Response Time involved (as that is only part of route selection operation). Hence, we will encounter fewer instances where the destination has to go through a Response Time wait.

We now study the effect of Hop Counts on ACO-EEAODR performance while changing the number of nodes. The results are captured in Table 6 and depicted in Figure 8.

Number of nodes	Hop Counts	
	ACO-based solution	EEAODR
20 nodes	3	2
40 nodes	2	2
60 nodes	5	5
80 nodes	8	6
100 nodes	8	5
120 nodes	11	7
140 nodes	9	4
160 nodes	15	12

Table 6: Hop Counts vs. Number of Nodes for EEAODR & ACO-EEAODR

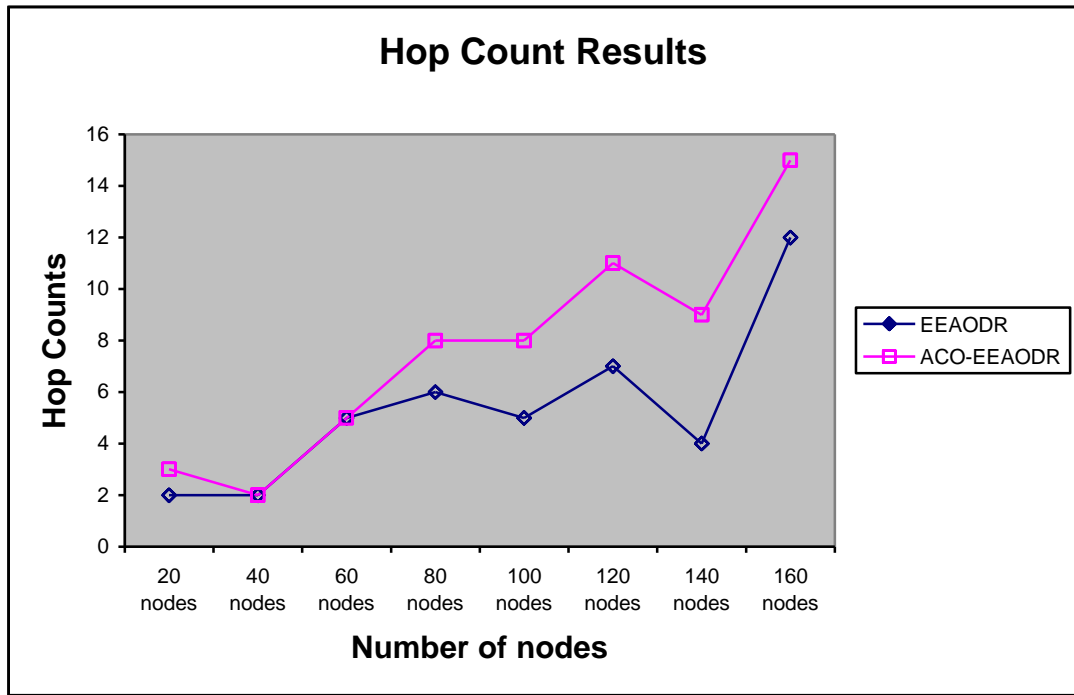


Figure 8: Hop Counts vs. Number of Nodes for EEAODR & ACO-EEAODR

As it can be seen from Figure 8, initially ACO-EEAODR take more hops to transfer data than EEAODR. However, this number slowly comes down and becomes lower than EEAODR's. This happens because nodes that have low battery power and are close to dying out, gradually stop becoming part of the best paths. Only nodes that contain high battery powers are being used to transmit data, even though it might increase the length of the path. This causes low-energy nodes to live longer (as they are not being used often), increasing the life of network.

Now, we examine the Network Lifetime against the number of packets transmitted. The results are captured in Table 7 and are depicted in Figure 9:

Number of packets	Network Lifetime (secs)	
	ACO-based solution	EEAODR
512	312	363
1024	230	341
1536	302	156
2048	287	98
2560	324	43
3072	251	28
3584	240	11
4096	195	0

Table 7: Network Lifetime vs. Number of packets for EEAODR & ACO-EEAODR

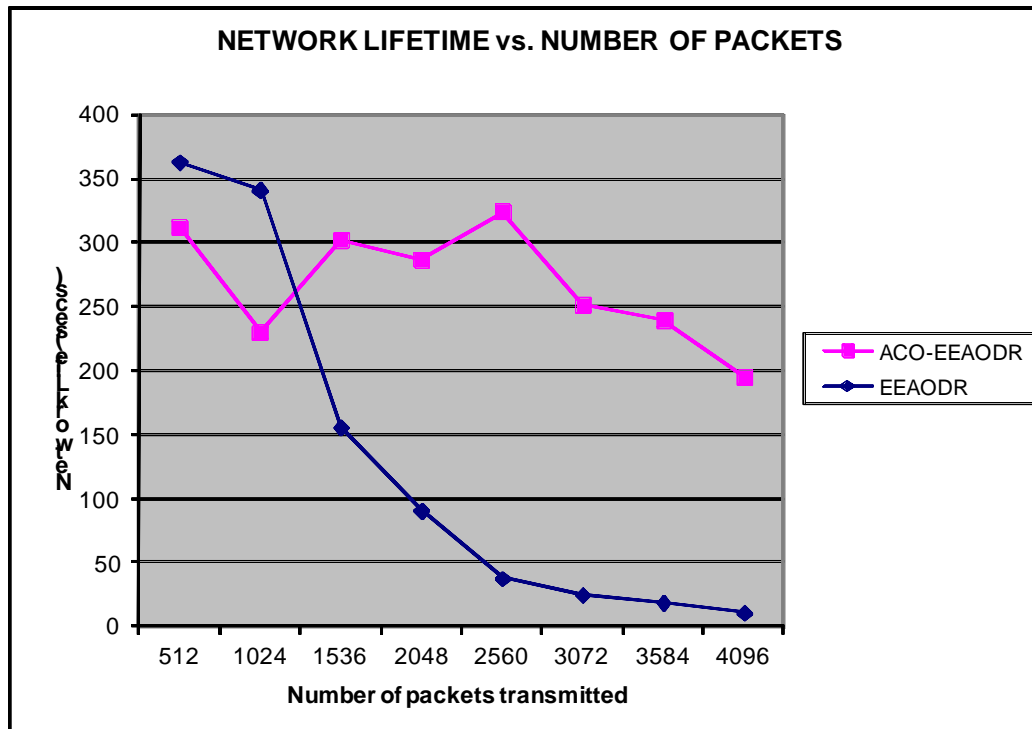


Figure 9: Network Lifetime vs. Number of packets for EEAODR & ACO-EEAODR

Figure 9 demonstrates how the ACO-based routing algorithm and the original EEAODR differ in keeping the network stable. Network Lifetime is considered as the time until which there are no dead nodes in the network. If even one node dies OR loses energy below a certain threshold level, then that node is no more part of the network and the entire network is considered dead (as that particular node may be taking part in many routes).

The above graph shows a better performance for the original EEAODR. This is true with the reason being that initially a significant amount of energy is lost by using ACO based solution in transmitting and receiving control packets and accounting for wait times and the source and destination. (time which the destination waits after receiving the first RREQ and 'Response Time' which the destination waits after receiving the first RREQ). However, as it is clear, the ACO-based solution learns more and more routes, which are stored in tables called 'Route Tables' in each node. This improves route discovery in the long run. Additionally, using the ACO-based solution, different nodes are selected even if there are 2 consecutive requests for the same source-destination pair. This is due to the fact that ACO-based solution gives a higher percentage of weightage to energy than hop count/pathlength. As a result, all nodes in the network are uniformly included to cater to data transmission requests, resulting in overall higher energy level of the entire network. This ensures that whenever a new data transfer request is generated, ACO-EEAODR tries to select a route that contains nodes with higher energy level rather than the nodes that form the shortest path. Even though this approach does not guarantee the shortest path, it guarantees that the network will last longer since low energy nodes will not be kept out of data transfer operations and only high energy nodes will participate. Although mobile devices contain limited power, above simulation results show that ACO-EEAODR increases network lifetime.

# Chapter 5

## Conclusion

In this thesis, we have proposed an enhancement (so-called ACO-EEAODR) to an existing energy efficient routing algorithm for MANETs (called EEAODR) and have compared its performance against the EEAODR, used as benchmark. Our simulation results demonstrate that ACO-EEAODR outperforms EEAODR in terms of Energy Consumed and Network Lifetime chosen as performance metrics. It has been observed that the network lifetime significantly increases as we have implemented a continuous learning technique, where preferred paths are stored in tables called Routing Tables, and are used later. This decreases the need for transmitting control packets (i.e. RREQ and RREP) to find the best path. The total energy that a network consumes for a particular data transfer request has also proven to be far less than the original algorithm.

In the future, more focus could be put on decreasing the waiting times at source and destination, so that ACO-EEAODR provides better results even in the initial stages of simulation. Also, since this algorithm has been implemented by using a particular derivative of ACO i.e. AS (Ant System), there is an opportunity to try and enhance the EEAODR using other derivatives of ACO such as SACO.



# References

- [1] J. Hoebeke, I. Moerman, B. Dhoedt, P. Demeester, "An Overview of Mobile Ad Hoc Networks: Applications and Challenges", *Journal of the Communications Network*, vol. 3, pp. 60-66, July, 2004.
- [2] A. Ephremides, "Energy Concerns in Wireless Networks", *IEEE Wireless Communications*, pp. 48-59, Dec. 15-17, 2002.
- [3] B. Forouzan, "Data Communication and Networking", 4/e, McGraw-Hill, ISBN: 0072967757, 2007.
- [4] W. Stark et al., "Low Energy Wireless Communication Network Design", In *Proc. of the 38<sup>th</sup> Annual Allerton Conference Communication, Control, Comp.*, Urbana, Illinois, Oct., pp. 785-805, 2000.
- [5] V. Rodoplu and T. H. Meng, "Minimum Energy Mobile Wireless Networks", *IEEE Journal on Selected Areas in Communications*, vol. 17, Aug. 1999.
- [6] S. Singh, M. Woo, and C. S. Raghavendra, "Power aware Routing in Mobile Ad Hoc Networks", In *Proc. of ACM/IEEE Intl. Conference on Mobile Computing and Networking (MobiCom'98)*, Dallas, Texas, USA, Oct., pp. 181-190, 1998.
- [7] I. Stojmenovic and X. Lin, "Power-Aware Localized Routing in Wireless Networks", In *Proc. of the IEEE Symposium on Parallel and Distributed Processing Systems*, May, 2000.
- [8] R. Ramanathan and R. Rosales-Hain, "Topology Control of Multihop Wireless Networks Using Transmit Power Adjustment", In *Proc. of the 9<sup>th</sup> IEEE International Conference on Computer Communications (INFOCOM 2000)*, Tel-Aviv, Israel, Mar. 26-30, pp. 4043, 2000.
- [9] K. Scott and N. Bambos, "Routing and channel assignment for low power transmission in PCS", In *Proc. of the 5<sup>th</sup> IEEE International Conference on Universal Personal Communications (ICUPC'96)*, Cambridge, MA, USA, Sep. 29 – Oct. 2, 1996.

- [10] Z. Li, Y. Zhen-Wei, Z. Yang, Z. Chun-Kai, "A Power-aware Adaptive Dynamic Routing Scheme for Wireless Ad Hoc Networks", In Proc. of the IEEE International Conference on Networking, Sensing and Control (ICNSC 2008), Sanya, China, pp. 966-970, Apr. 6-8, 2008.
- [11] S. Singh, M. Woo, and S. Raghavendra, "Power-aware with Routing in Mobile Ad Hoc Networks", In Proc. of ACM/IEEE Intl. Conference on Mobile Computing and Networking (MobiCom'98), Dallas, TX, USA, 1998.
- [12] C.-K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks", IEEE Communications Magazine, June 2001.
- [13] J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad hoc networks", In Proc. of the 9<sup>th</sup> IEEE International Conference on Computer Communications (INFOCOM 2000), Tel-Aviv, Israel, Mar. 26-30, pp. 22-31, 2000.
- [14] S. K. Dhurandher, S. Misra, M. S. Obaidat, P. Gupta, K. Verma, and P. Narula, "An Energy-Aware Routing Protocol for Ad-Hoc Networks Based on the Foraging Behavior in Ant Swarms", In Proc. of the IEEE International Conference on Communications (ICC'09), Dresden, Germany, June 14-18, pp. 1-5, 2009
- [15] M. Dorigo, "Optimization, learning and natural algorithms", Ph.D thesis, Dipartimento di Elettronica, Politecnico di Milano, Italy, 1992.
- [16] M. Dorigo, V. Maniezzo, A. Colorni, "Positive feedback as a search strategy", Technical Report 91- 016, Dipartimento di Elettronica, Politecnico Di Milano, Italy, 1991.
- [17] M. Dorigo and G. Di Caro, "The Ant Colony Optimization meta-heuristic", In D. Corne, M. Dorigo, and F. Glover (Eds.), New Ideas in Optimization, McGraw-Hill, London, U.K, pp. 11–32, 1999.
- [18] A. P. Engelbrecht, "Computational Intelligence: An Introduction", 2<sup>nd</sup> Edition, University of South Africa, Pretoria, John Wiley & Sons Ltd., ISBN 978-0-470-03561-0, 2007.
- [19] K. Socha, "ACO for continuous and mixed variable optimization", In: Dorigo M, et al. (Eds.), 4<sup>th</sup> International Workshop on Ant Colony Optimization and Swarm Intelligence (ANTS 2004), Lecture Notes in Computer Science, vol. 3172, Springer, Heidelberg, pp. 25–36, 2004.

- [20] M. Guntzsch and M. Middendorf, "Applying population-based ACO to dynamic optimization problems", In Dorigo, M., Di Caro, G. A., Samplels, M. (Eds.) ANTS 2002, Lecture Notes in Computer Science, vol. 2463, pp. 97-104, Springer, Heidelberg, 2002.
- [21] C. J. Eyckelhof and M. Snoek, "Ant Systems for a dynamic DSP: Ants Caught in a Traffic Jam", In Proc. of 3<sup>rd</sup> Intl. Workshop on Ant Algorithms (ANTS02), Brussels, Belgium, Sept. 12-14, 2002.
- [22] M. Dorigo, V. Maniezzo, A. Colorni, "The Ant System: an autocatalytic optimizing process", Dipartimento di Elettronica, Politecnico di Milano, Italy, 91-016, Revised, 1991.
- [23] L. M. Gambardella and M. Dorigo, "Solving Symmetric and Asymmetric TSPs by Ant Colonies", In Proc. of IEEE International Conference on Evolutionary Computation, Nagoya , Japan, pp. 622–627, May 20-22, 1996.
- [24] T. Stutzle and H. Hoos, "MAX-MIN Ant System and Local Search for The Traveling Salesman Problem", In Proc. of the IEEE International Conference on Evolutionary Computation, Indianapolis, IN, USA, Apr. 13-16, pp. 309–314, 1997.
- [25] O. Roux, C. Fonlupt, and E-G. Talbi, "ANTabu", Technical Report LIL-98-04, Laboratoire d'Informatique du Littoral, Université du Littoral, Calais, France, 1998.
- [26] V. Maniezzo and A Carbonaro, "Ant Colony Optimization: An Overview", In C. Ribeiro, editor, Essays and Surveys in Metaheuristics, pp. 21-44, Kluwer Academics, Boston, MA, USA, 1999.
- [27] G. Di Caro and M. Dorigo, "Ant colonies for adaptive routing in packet-switched communications networks", In Parallel Problem Solving from Nature - PPSN V, Lecture Notes in Computer Science 1498, pp. 673–682, Sept., 1998.
- [28] G. Di caro, "Ant Colony Optimization and its Application to Adaptive Routing in Telecommunication Networks", PhD thesis in Applied Sciences, Polytechnic School, Université Libre de Bruxelles, Brussels, Belgium, 2004.] G. Di Caro, "A society of ant-like agents for adaptive routing in networks", DEA thesis in Applied Sciences, Polytechnic School, Université Libre de Bruxelles, Brussels, Belgium, 2001.

- [29] G. Di Caro, "A society of ant-like agents for adaptive routing in networks", DEA thesis in Applied Sciences, Polytechnic School, Université Libre de Bruxelles, Brussels, Belgium, 2001.
- [30] M. Saleem, M. Farooq, "BeeSensor: A bee-inspired power aware routing protocol for wireless sensor networks", In Proc. of EvoWorkshops (EvoCOMNET), Valencia, Spain, Lecture Notes in Computer Science 4448, pp. 81-90l, 2007.
- [31] G.A. Di Caro, F. Ducatelle, L.M. Gambardella, "Ant Colony Optimization for Routing in Mobile Ad Hoc Networks in Urban Environments", Technical Report IDSIA-05-08, IDSIA, Lugano, Switzerland, May, 2008.
- [32] V. R. Zanjani, A. T. Haghighat, "Adaptive Routing in Wireless ad hoc Networks Using Ant Colony Optimization", In Proc. of International Conference on Computer Technology and Development (ICCTD'09), Kota Kinabalu, Malaysia, Nov. 13-15, pp. 40-45, 2009.
- [33] H. Al-Zurba, T. Landolsi, M. Hassan, F. Abdelaziz, "On the suitability of using Ant Colony Optimization for Routing in Multimedia Content over Wireless Sensor Networks", International Journal on Applications of Graph Theory in Wireless Ad Hoc Networks Networks (GRAPH-HOC), vol. 3, No. 2, June, 2011.
- [34] F. Celik, A. Zengin and S. Tuncel, "A survey on swarm intelligence based routing protocols in wireless sensor networks", International Journal of the Physical Sciences, Academic Journals, ISSN 1992-1950, vol. 5 (14), pp. 2118-2126, Nov. 4, 2010.
- [35] T. Stutzle, M. Lopez-Ibanez, M. Dorigo, "A concise Overview of Applications of Ant Colony Optimization", In Encyclopedia of Operations Research and Management Science, James J. Cochran (Ed.), John Wiley & Sons Inc., 2010.
- [36] S. K. Dhurandher, S. Misra, M. S. Obaidat, V. Bansal, P. R. Singh, V. Punia, "EEAODR: An energy-efficient ad hoc on-demand routing protocol for mobile ad hoc networks", International Journal Of Communication Systems, pp. 789–817, 2009.
- [37] V. D. Park and M. S. Corson. Temporally-ordered routing algorithm (TORA) version 1 functional specification. Internet Draft, Internet Engineering Task Force, Nov., 2000.

- [38] C. Perkins, E. Belding-Royer, and S. Das, Ad hoc On- Demand Distance Vector (AODV) Routing, RFC 3561, July 2003
- [39] C.E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", *Comp. Comm. Rev.*, pp.234-244 Oct. 1994.
- [40] Stefano Basagni, Marco conti, Silvia Giorano and Ivan stojmenovic, 'Mobile Ad hoc networking, " IEEE press, wiley interscience, 2004.
- [41] J. Ren, Y. Tu, M. Zhang, and Y. Jiang, "An Ant-Based Energy-Aware Routing Protocol for Ad hoc Networks", *Proc. of International Conference on Computer Science and Service System (CSSS)*, Nanjing, China, June 27-29, pp. 3844-3849, 2011.
- [42] Y-M. Kim E-J. Lee H-S. Park, "Ant Colony Optimization Based Energy Saving Routing for Energy-Efficient Networks", *IEEE Communications Letters*, vol. 15, Issue 7, pp. 779-781, July, 2011
- [43] S. K. Dhurandher, M. S. Obaidat, and M. Gupta, "Application of Ant Colony Optimization to Develop Energy Efficient Protocol in Mobile Ad-Hoc Networks", in *Proc. of WINSYS-2011*, Seville, Spain, pp. 12-17, 2011.
- [44] B. Roy, S. Banik, P. Dey, S. Sanyal, N. Chaki, "Ant Colony based Routing for Mobile Ad-Hoc Networks towards Improved Quality of Services", *Journal of Emerging Trends in Computing and Information Sciences*, VOL. 3, NO. 1, January 2012
- [45] A. Verma, "Adhoc Network Routing Optimization and Performance Analysis of ACO Based Routing Protocol", *International Journal of Advanced Research in Computer Engineering & Technology*, Volume 1, Issue 1, March 2012.
- [46] S. K. Dhurandher, M. S. Obaidat, and M. Gupta, "Application of Ant Colony Optimization to Develop Energy Efficient Protocol in Mobile Ad-Hoc Networks", in *Proc. of WINSYS-2011*, Seville, Spain, pp. 12-17, 2011.