OPTIMIZING HYBRID WIRELESS SENSOR NETWORK PERFORMANCE USING MOBILE NODES

MICHAEL PORTNOY

A THESIS SUBMITED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN COMPUTER SCIENCE

GRADUATE PROGRAM IN COMPUTER SCIENCE
YORK UNIVERSITY, ONTARIO, CANADA
MARCH, 2008

OPTIMIZING HYBRID WIRELESS SENSOR NETWORK PERFORMANCE USING MOBILE NODES

BY MICHAEL PORTNOY

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES OF YORK UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN COMPUTER SCIENCE

Statement of Authenticity

Abstract

Wireless Sensor Networks (WSNs) are comprised of multiple, small, inexpensive, static devices capable of forming ad-hoc networks for the purpose of environmental monitoring. Due to environmental factors and randomness of deployment, WSNs are prone to the formation of routing anomalies (a.k.a. *routing holes*), which can adversely affect network performance.

In this thesis, we examine the underlying problems associated with routing holes and propose a new remedial technique in which mobile nodes are used to bridge the communication gaps created by the routing holes. We introduce a new routing anomaly which we term the *microhole* – a small imperfection in the routing path that is not classified as a traditional routing hole – and demonstrate the applicability of our technique to even such basic routing imperfection. Expanding on the work done in conjunction with our colleagues, we apply our technique to a scenario where a routing hole is impacted by multiple, variable-rate, data streams. Specifically, we propose **OPlaMoN**-2 – "**O**ptimal **Pla**cement of a **Mo**bile **N**ode – 2" distributed algorithm, which is aimed at finding the most effective deployment location (with respect to a combined, weighted, energy-delay performance metric) of a mobile-bridge inside a routing hole.

Finally, we simulate our theoretical findings as further proof of correctness and appropriateness of our technique and present our conclusions and some of the directions for possible future work.

Acknowledgements

I would like to extend my greatest gratitude to my Thesis Supervisor – Dr. Natalija Vlajic. She provided me with exceptional mentoring and support, far surpassing her duties. Her adamant desire to have me succeed and infallible belief in my abilities were some of the determining factors for the successful completion of my Masters degree and this thesis report.

Secondly, I would like to thank Nelson Moniz who was never hesitant to sacrifice his time in order to help and advise me. Thank you Nelson! It was a pleasure knowing and working with you! Also, I would like to thank everyone in the Signal Processing and Communications lab for harboring such a positive work environment. Especially, I would like to thank Anton Belov for taking the time to review my theoretical conclusions and Dr. Jeff Edmonds for his eagerness to help.

Also, I would like to thank the people from the Computer Science Department Technical and Administrative staff who were always helpful and caring.

Last, but not least, I would like to thank my Laura for her continual support and encouragement. You brought me light when I felt darkest, as well as my family who are always there for me no matter what.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Large Scale WSNs

Wireless Sensor Networks (WSNs) are comprised of a large number of small and inexpensive, yet resource constrained electronic devices. These devices are generally capable of sensing the environment using one or more sensor elements (e.g. thermal, seismic, sonic, motion and etc.), while at the same time performing simple computations and exchanging critical information with each other via built-in wireless transceivers (see Figure 1.1). Once a group of sensors is scattered over an area, the sensors distributively self-organize into a multi-hop network in which the transmission and delivery of data is usually accomplished by means of geographic routing (also known as greedy or compass routing). In geographic routing, nodes forward packets to

other nodes that are found to be closer to the destination than themselves. Through such simple forwarding strategy, geographic routing eliminates the need for nodes to store state information tables, exchange state updates, or carry large packet overheads. This implies considerable savings in energy, memory and processing time of individual sensor nodes. The 'statelessness' of the geographic routing approach also provides for excellent scalability which is highly critical in large-scale WSNs [1].

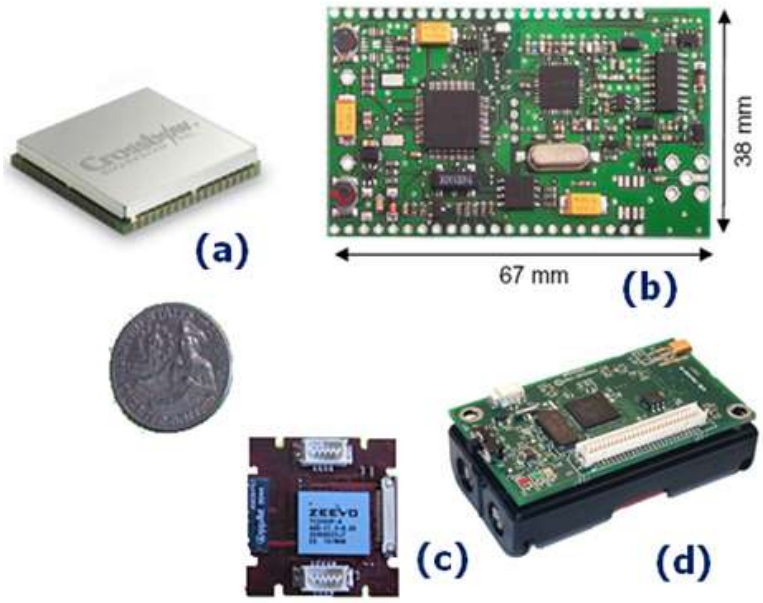


Figure 1.1 A variety of commercial wireless sensor nodes: **(a)** Crossbow IRIS OEM 2.4GHz Module – designed for deeply embedded sensor networks **[2] (b)** ScatterWeb ScatterNode **[3] (c)** Intel Mote prototype **[4] (d)** MICA2 868 916MHz – wireless platform for low-power sensor networks with multi year battery life capabilities **[2]**.

WSNs are most fitting for deployment in areas where an event needs to be promptly monitored, tracked or controlled using sensors; yet, the setup of a conventional, wired

infrastructure is not possible or justifiable. For instance, in the case of a forest fire, the levels of toxic gases or heat radiation can be safely monitored using aerially deployed WSN. As the fire spreads, the sensors locally measure (i.e. capture) and then forward this information to the main processing center, ultimately enabling the event to be accurately mapped and its progress monitored (see Figure 1.2).

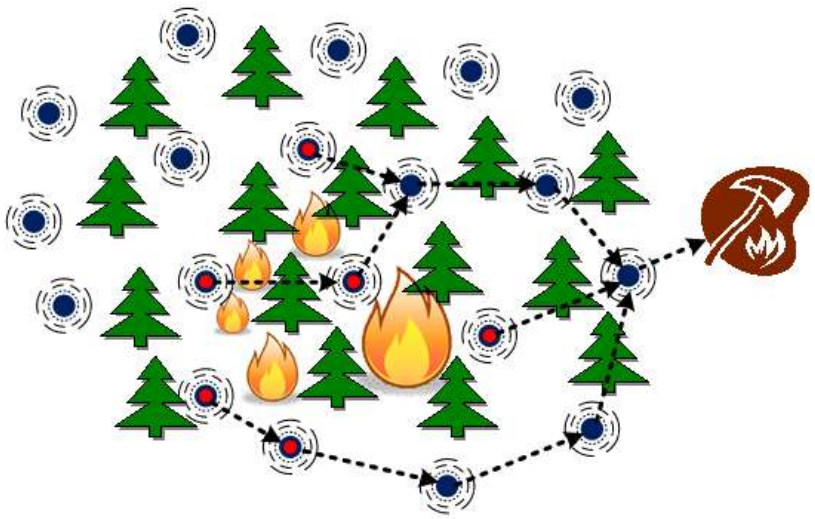


Figure 1.2 Using a WSN, a forest fire can quickly be detected and its progress monitored. In the above, nodes detecting a fire, quickly form an ad hoc network with their neighbors and transmit useful data (e.g. temperature or CO levels) to the fire (base) station.

Similarly, WSNs equipped with seismic or sonic sensors can be used for safe and/or unintrusive tracking of objects, people or animals, throughout areas that are inaccessible to humans (see Figure 1.3). WSNs can also be used in industrial applications such as controlling of a nuclear reactor, where high density distribution of wireless

sensors around the reactor may provide for rapid and appropriate response in case of critical temperature or vibration changes [5].

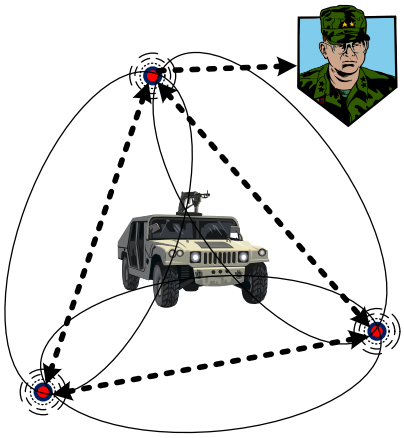


Figure 1.3 As the above vehicle moves, its precise movement vector is constantly monitored by WSN nodes through signal triangulation.

The design and operation of WSNs involves many challenges due to the nature of these networks. Node localization is one such problem because nodes are often dispersed using methods that are inherently imprecise (e.g. dropping nodes from a plane). Since geographic routing requires each node to be aware of its global coordinates, a great deal of effort has been put towards finding ways to localize nodes without the use or with partial use of GPS capabilities[1] [6]. However, even if nodes are localized, there is still a good chance that their positions are less than optimal from the perspective of minimum-cost routing.

[1] Installing a Global Positioning System (GPS) function on each node is not always feasible due to high cost or unavailability of a line of sight (LOS) communication between the nodes and GPS satellites (i.e. satellite signals are blocked by forest foliage).

Besides localization and positioning problems, the operation of WSNs is also challenged by constrained internal resources of sensor nodes, including: limited energy supply, limited storage and limited processing power. Out of the three, the first constrain – limited energy supply – is particularly severe, as it implies indirect restrictions on a number of other parameters, such as node's radio range, amount of transmitted, received and processed data, time spent in wake-up/idle/sleep mode, etc. In order to gain better understanding of this particular problem and possible ways of dealing with it, in the next section we provide a detailed energy consumption model of a typical sensor node [7].

1.2. Energy Model of Sensor Nodes: What drains the energy most?!

In WSNs, energy consumption can be attributed to the following three types of nodal activity: 1) sensing, 2) computation, and 3) communication. Energy consumption due to sensing activity is directly related to the frequency of sensing and volume of sensed information, as well as the amount of energy needed to sense a single bit [8]. Such parameters are usually readily available, and therefore modeling or controlling a node's sensing energy consumption presents little difficulty. Modeling or controlling the energy consumption due to computational and communication activity is a more challenging task. Namely, computational energy consumption is closely coupled with the overall

level of activity occurring in a WSN, which may greatly vary or be completely unpredictable at times. Energy consumption due to communication also depends on parameters that are variable and unpredictable, such as network topology and background noise levels [8; 9; 10]. Please note, it is generally considered that energy consumption due to computation represents only a small fraction of the total WSN energy consumption, with the majority of energy spent on communication [8].

The overall energy used by a sensor node for communication activity depends on the amount of time that the node spends in each of the following possible radio states: transmitting, receiving, overhearing, idle listening, and sleeping.

a) *Transmitting* is usually the most energy expensive operation and can further be decomposed into operations such as: digital sequence generation (encoding), wave form modulation, RF modulation and amplification. Due to the nature of signal propagation (specifically - path loss), the energy required for transmission of a signal increases exponentially with relation to the distance between the sender and its respective receiver. The actual order of exponential increase – aka *path loss exponent* – depends on the nature of propagation environment; though, its value generally ranges from about 2 to 6 [8; 11; 12] (see Table and Table 2.2in Chapter 2).

b) *Receiving* can also be a very energy consuming operation – sometimes as or even more expensive than transmitting. To elaborate, both receiving and transmitting have a similar fixed cost in electronics (e.g. signal de/modulation, d/encoding), with a variable

cost for transmitting depending on the distance to the receiver. However, if the distance to the receiver is relatively small, the constant cost of the receiver electronics may become comparable to the variable cost of transmission and therefore a significant part of the total cost [13]. Nonetheless, it is important to emphasise that it is still a constant expense that is only dependent on the bit rate of the incoming information [8].

c) *Overhearing* packets from other nodes and sensing the medium in order to ensure that it is safe to transmit adds further to energy use and can sometimes become a significant factor in the overall network energy consumption scheme. For example, this is true for dense networks where each node has a large number of other nodes (i.e. neighbours) in its communication range. In such networks, unless some form of scheduling is applied, each node needs to frequently check the packets that it overhears in order to ensure that they are not destined for itself. Also, if a node is required to send information, it may have to sense the medium many times (due to high levels of network congestion) before it is able transmit.

d) In the *idle listening* state, nodes listen to an inactive medium (i.e. a medium without transmissions destined for the listening node) for information to be received. Therefore, the idle state energy consumption is typically considered to be comparable to the energy required for receiving since the node's radio electronics are in an active state. Several studies have shown that in the cases when internodal communication occurs over fairly sparse time intervals, idling can on average consume 50 to 100% of

the energy required for receiving data [10; 14]. In other words, more than 50% of energy is 'wasted' due to a node waiting to have data sent to it. As such, it is in the best interest of the WSN nodes to stay in the sleep state (where most of the electronics, such as the radio, are turned off) as much as possible since it consumes substantially less energy than idling [8].

We focus on the radio states of receiving and transmitting, as these are generally the most energy expensive operations per bit of information transmitted. Based on the work presented in [13; 15] and [16], the expressions (1.1) and (1.2) represent a realistic way of modeling (i.e. characterizing) the actual energy consumption in the states of transmitting and receiving (see Figure 1.4), with realistic energy consumption characteristics shown in Table 1.1.

Transmitting

$$E_{Tx} = e \cdot l + u \cdot l \cdot d^k \tag{1.1}$$

Receiving

$$E_{Rx} = e \cdot l \tag{1.2}$$

where,

E_{Tx} = energy required to transmit l bits
E_{Rx} = energy required to receive l bits
l = number of bits to send/receive
e = energy the radio dissipates per bit in order to run transceiver electronics
u = energy spent by the amplifier per bit to counter signal propagation loss
k = propagation loss exponent ($k = 2$ in our case)
d = distance to the receiver

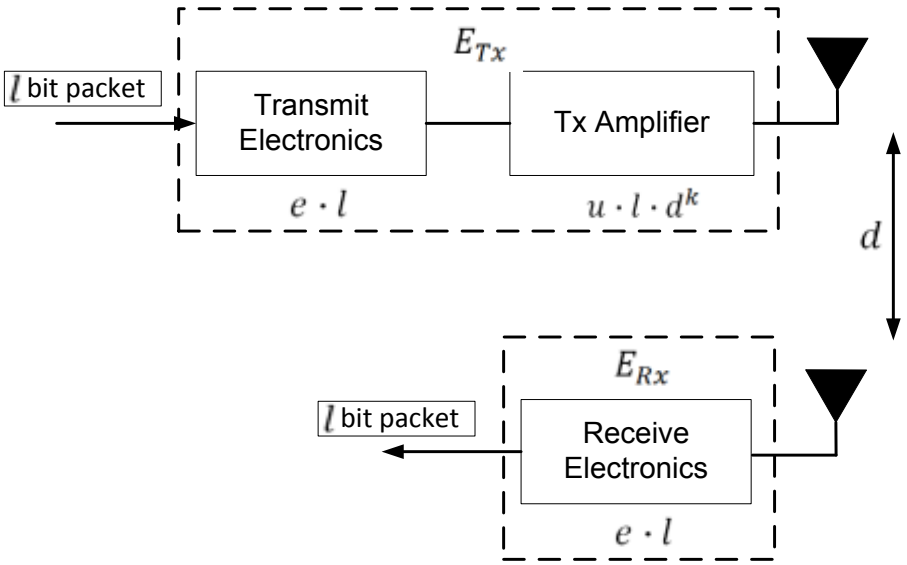


Figure 1.4 Generic radio model [15].

Operation	Energy Dissipated
Transmitter/Receiver Electronics	50 nJ/bit
Tx Amplifier	100 pJ/bit/m^2

Table 1.1 Possible radio characteristics [15].

1.3. Topological Anomalies in WSNs

In an ideal WSN deployment scenario, sensor nodes would be placed at precise (globally optimal) locations, and they would operate without any interruptions throughout the intended lifetime of the network. However, in reality, sensor nodes are often deployed into hostile environments by imprecise means, such as aerial drops. Moreover, during their operation, nodes are likely to experience interruptions and/or complete failure due to one or more of the following:

- component malfunctioning;
- battery depletion;
- environmental factors: extreme heat, flooding, freezing, etc.;
- man-caused factors: interference, accidental damage, explosion, etc..

The malfunction or complete failure of one or multiple sensors inevitably leads to the formation of the so-called *holes* – areas devoid of functional nodes – which have the potential to cause serious impediment to the normal operation of the network.

WSN holes come in a number of varieties. For example, a *coverage hole* comprises a group of nodes unable to sense their environment. A *routing hole* comprises a group of nodes incapable of communicating with their neighbours. Other known types of holes include: *jamming*, *sink*, *black*, w*orm holes*. In contrast to the earlier two types, these holes are rarer and generally appear as a result of deliberate denial of service (DoS) attacks on the network by outside intruder nodes [17].

In our work, we have concentrated on the second types of holes – routing holes, due to their particularly severe impact on the performance of WSNs, as discussed below.

1) Local minimum phenomenon. In WSNs that employ greedy geographic routing, a routing hole may facilitate the so-called *local minimum phenomenon,* where none of a node's neighbors are closer to the destination than the forwarding node itself [18] (see Figure 1.5). Packets that arrive to such a node will get 'stuck', unless one of the improved versions of geographic routing is employed.

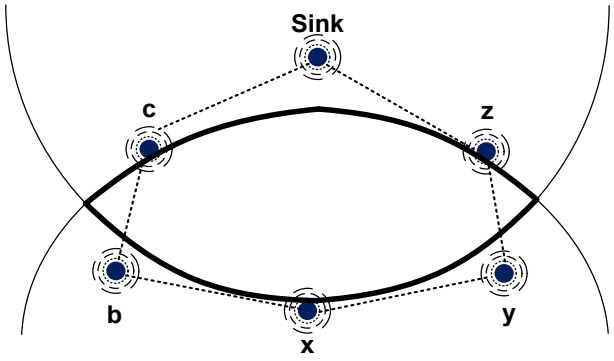


Figure 1.5 If geographic routing is used, node *x* will not know to which node to forward data destined for the Sink since none of its neighbors (i.e. *b* and *y*) are closer to the Sink than itself. In such case, node *x* is considered to be a local minimum node or stuck node, with nodes *x*, *b*, *c*, *Sink*, *z* and *y* forming the boundary of a routing hole with respect to *x*.

2) Longer routes and packet delays. Even if local minimums are overcome through the use of advanced types of geographic routing, routing holes can still be detrimental as they may interfere with the optimal (i.e. shortest-path and/or minimum-delay) flow of

traffic, e.g. by forcing the traffic to flow around the perimeter of the hole (see Figure 1.6).

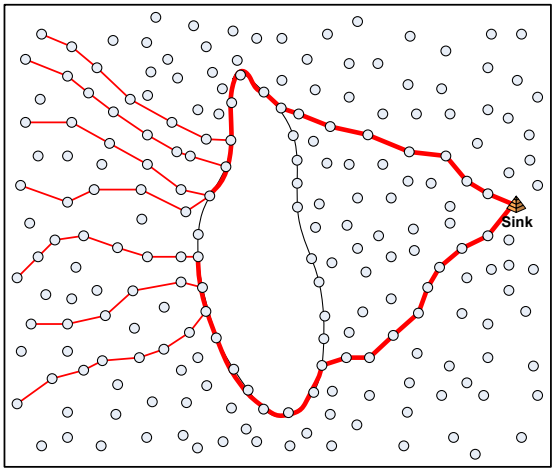


Figure 1.6 In the above, multiple data streams from multiple sources are impeded by a routing hole on their way to the Sink. As the data is routed around the hole, it converges into two large streams, which strain the boundary nodes' resources disproportionately to the rest of the network, threatening to partition it.

3) Hole expansion phenomenon. In addition to longer routes and packet delays, the so-called *hole expansion phenomenon* is another challenge arising from the existence of the routing hole in a scenario depicted in Figure 1.6. Namely, the nodes along the perimeter of the hole suffer an increased energy burden, as multiple traffic streams are now forced to converge into one single stream and be routed through the same set of nodes. (Note, in the absence of the routing hole in Figure 1.6, the optimal routing paths of different traffic streams would likely be disjoint. Consequently, the average energy consumption would be distributed more evenly across the network.) The ultimate

outcome of this phenomenon is premature battery exhaustion of the peripheral nodes and further expansion of the hole.

A number of approaches exist that attempt to mitigate the impact of routing holes in WSNs. Some of these approaches simply find ways for preventing packets from getting stuck at a local minimum, while ignoring the existence and consequence of the hole expansion phenomenon [18; 19; 20; 21; 22]. Another group of approaches propose ways to combat the hole expansion phenomenon, e.g. by moving the traffic away from routing holes [23; 24; 25], but at the expense of higher energy consumption and longer packet delays. Asides from their very specific and narrow scope, one limitation common to all of the above approaches is that they are rather 'defensive' as they focus on 'working around' instead of 'working against' routing holes, e.g. by introducing physical alterations to the network topology. In the subsequent section, we will look into such, more proactive ways of dealing with routing holes by means of topological change(s).

1.4. Hybrid Large Scale WSNs: Mixture of Static and Mobile Nodes

Adding mobile nodes to a wireless sensor network is a rather intuitive way of optimizing network performance and/or combating network anomalies (including routing holes), as mobile nodes enable rapid and effective alterations in the network topology (see Figure 1.7).

The previous research on the use of mobile nodes in WSNs falls into two broad categories:

- works that consider purely mobile WSNs (with all nodes being mobile);
- works that consider a mixture of mobile and static nodes.

In the reminder of this section, we will provide a brief overview of the most significant works from each the two categories.

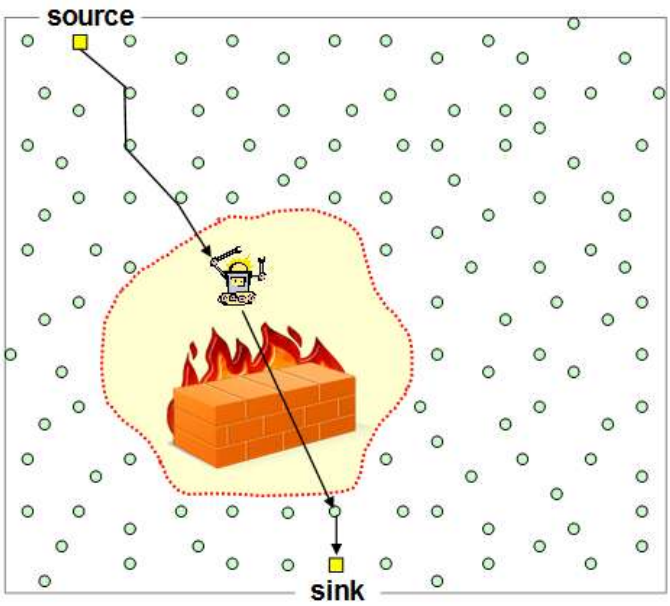


Figure 1.7 Mobile node is used to bridge a routing hole.

1.4.1. Purely Mobile WSNs

Much of the ongoing research on purely mobile WSNs[2] has been dedicated to finding ways to improve WSN coverage by relying on nodes' mobility. (To our knowledge, no work from this category has dealt with the issue of routing holes.) Current Mobile WSNs coverage optimization research can generally be broken down into three main sub-categories: *computational geometry*, *virtual forces* and *sequential node distribution* [17].

The use of computational geometry for maximizing network coverage was first suggested by Wang et al. in [26]. The proposed solution makes use of Voronoi diagrams in order to discover areas that do not fall into a node's sensing disk. More specifically, each WSN node firstly generates its Voronoi cell based on the location coordinates of its neighbours. If some points of a node's Voronoi cell are not covered by its sensing disk, these areas can be assumed to be coverage holes. Three closely related distributed, self-deployment algorithms are proposed by the authors of [26]: VEC, VOR, and Minmax. VEC (standing for VECtor) calculates the average distance between the node and all its neighbours and then subsequently tries to move the node to the position where the distance to its neighbours would be as close as possible to the earlier calculated average. VOR (standing for Voronoi) moves the node towards the farthest vertex of its Voronoi cell in order to cover its largest local coverage hole. Minmax also moves the

[2] In purely mobile WSNs, all sensor nodes are capable of locomotion and therefore are able to modify the WSN topology in order to better meet network goals.

node towards the farthest Voronoi cell vertex, but it also considers the distances to other vertices while attempting to minimize the distance to the farthest vertex.

Virtual forces are another way to autonomously distribute a WSN so that it provides maximal coverage while maintaining connectivity. For this approach, nodes and obstacles are treated as particles exerting forces on each other. For instance, in [27] Howard et al. assumed nodes and obstacles to have potential fields that cause them to repel one another when closer than a certain threshold. A similar approach was suggested by Shucker et al. in [28], though instead of potential fields virtual springs were used as the acting force between the nodes, causing the nodes to repel when they were too close and to attract when they were too far with respect to a particular threshold. The virtual forces approach is proved to be quite successful in maximizing the coverage area and had the advantage of not needing centralized control, localization or communication between nodes, thus scaling well to very large networks.

Yet another proposed technique that attempts to deploy mobile nodes in a way that maximizes coverage while preventing the formation of coverage holes is the sequential approach. In [29] Howard et al. suggested to deploy nodes one-by-one while maintaining a line-of-sight constraint (i.e. each node is capable of communicating with at least one other neighbour at all time), and using the already deployed nodes as landmarks in order to make optimal placement decisions in the unexplored/uncovered areas. The approach is centralized as it relies on a base station for coordination and to

maintain a constant, multihop, bidirectional communication link to every node in the network. In contrast to distributed approaches, the sequential distribution scheme is likely to be slower as nodes are deployed one at a time.

1.4.2. Hybrid WSNs

Purely mobile networks are not always advantageous or possible as the costs of mobile nodes runs substantially higher than that of the static nodes. For example, the cost of a fully functional single static node is around $100 [30], while mobile nodes can cost as much as ten or even a hundred times more (depending on the mobile node capabilities). For instance, the cost of a Pioneer 3-DX robot is around $5000 [31]. This is a relatively large, two wheel drive robot, occupying an area of about 0.25m^2, and having a wide array of advanced sensors and a substantial continuous run time of about 18-24 hours [32]. A Pioneer 3-AT is similar in its dimensions and capabilities, except it is a four wheeled robot that is suitable for off-road conditions, with a higher ground clearance and a wider wheel base. Nonetheless, the off-road capability comes at a price as the AT is heavier and has a continuous run time of only about 4 to 8 hours [33]. Smaller (about 0.01m^2), lighter robots such as the Khepera III [34] or the tank-style treaded Surveyor SRV-1 [35] are cheaper (around $1000) [31], however have shorter continuous run-time, and are not suitable for very rugged terrains. An even smaller, cheaper mobile sensor node is the Robomote, designed by the Robotics Research Laboratory at the University of South California. It occupies an area of about only 0.000047m^3 (comparable to a

match box) and costs around $150 in raw components. Its functionality is comparable to its larger counterparts, with its 3 1.5V alkaline AAA batteries lasting for about 3.5 hours at 100% duty cycle [31]. However, the main disadvantage of this mobile sensor is that its small size makes it inappropriate for rough terrains.

To summarize the above, robust mobile sensors (i.e. robots) that are well equipped for locomotion in harsh environments are rather expensive; hence, in reality, it is often justifiable to deploy only a few such nodes in combination with a large number of static sensors (see Figure 1.8).

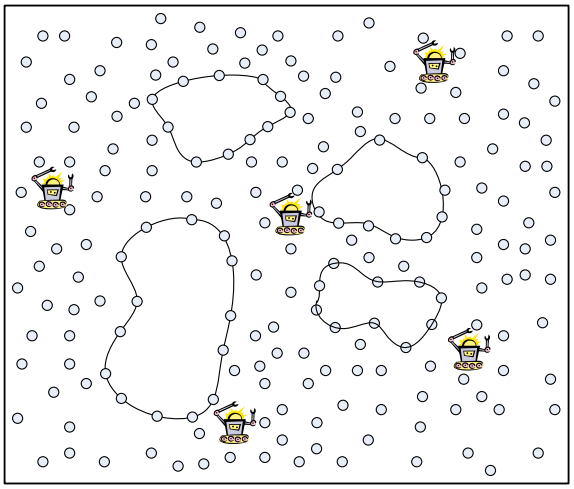


Figure 1.8 How should the mobile nodes be used to optimize coverage?

The existing research works on the topic of hybrid WSNs fall into two general categories:

1) ***WSNs with a Single Mobile Node.*** This hybrid-WSN configuration usually involves a highly capable mobile sensor node roaming the network environment and deploying

static sensors in insufficiently covered areas. A good example of this type of hybrid WSNs can be found in [36], in which Croke et al. introduced the idea of using an Unmanned Aerial Vehicle (UAV) to distribute static sensor nodes throughout the network field. Once a network is deployed, the nodes transmit their actual locations to the UAV, which compares these locations to its predefined network topology. If based on this comparison the UAV decides that there are communication or coverage holes in the network, it deploys additional static nodes to that region until the holes are fixed.

In [37], Batalin et al. present an approach in which a highly capable mobile robot explores an environment while deploying static nodes. Least Recently Visited (LRV) algorithm is used by the robot to emplace static nodes within the environment, which in turn self-organize into a network and emit navigational instructions to the robot as it goes by them at a later time. The navigational instructions are calculated based on the frequency of robot's visitations to, and past directions taken from, that particular node. Such an approach does not require the use of a Global Positioning System (GPS) and is decentralized with respect to static nodes. Also, if a static node fails, the robot will eventually take the route that includes this node, and once failing to receive a navigational instruction, will deploy another static node to replace the failed node, hence preventing the formation of coverage holes/communication holes.

2) ***WSNs with Multiple Mobile Nodes.*** In this type of hybrid-WSN configuration, mobile nodes are typically deployed with the aim to 'fill in' a coverage hole, e.g. by taking the

role of one or more of the missing/malfunctioning static nodes. Examples of works from this group are [38; 39].

In [38], Wang et al. have described a bidding protocol for which the initialization phase involves network nodes applying a technique similar to the one described in [26] in order to calculate Voronoi cells which are used to detect coverage holes. The bidding algorithm consists of three stages that run round by round: *service advertisement*, *bidding* and *serving*. In the service advertisement stage mobile sensors broadcast their location and base price – the negative cost incurred if the mobile moves and therefore creates a coverage hole of a certain size. In the bidding stage, the static nodes use their Voronoi cells in order to detect any **coverage** holes. If such holes exist, the information about these holes (e.g. size and location) is used to calculate the bids and the target locations for the mobiles. Since static nodes are aware of the locations and the base prices of the mobiles, they can attempt to find a mobile that is closest to the coverage hole and has a base price that is lower than the calculated bid. If such mobile is found, a bidding message is sent to this mobile, which in the serving stage can decide on the highest received bid and deploy itself to the corresponding hole, while adjusting its base price to the bidding price received from the static node.

In [39], Du et al. also considered using multiple mobile sensors in a WSN in order to improve coverage and routing. No specifics attempts at detecting routing or coverage holes were described; however, the authors did propose two centralized techniques for

deploying mobile nodes to areas that have relatively fewer nodes than the rest of the network. The two types of proposed mobile sensor placements approaches are the following: 1) cell-density based where mobiles are directed towards poorly covered areas of the network that have relatively low nodal densities; 2) longest inter-sensor distance where mobiles move to the center of the longest lines connecting Voronoi region neighbours. The second method is done one mobile at a time so that the relocation of the mobiles is taken into account. This of course comes at the price of a significantly slower deployment and greater processing time.

1.4.3. Our work

After a close examination of the existing literature on the utilization of mobile nodes in large scale WSNs (as described in the preceding section), we have concluded that the use of mobile nodes for the specific purpose of fixing/reducing the detrimental effects of routing holes remains a greatly understudied research area. Namely, most of the published WSN works on the use of mobile nodes have exclusively focused on the issue of suboptimal coverage and connectivity while ignoring any routing inefficiencies that may be present in the network. Furthermore, these studies appear somewhat limited in terms of the employed assumptions as they tend to ignore network scenarios where information is actively sensed only by a relatively small percentage of the network nodes. Consequently, global network optimization solutions suggested in these works

are likely to be inapplicable to a wide range of WSN deployment scenarios – target-tracking being one such example.

In our research we have concentrated on using one (or a small number of) mobile sensors in a Hybrid WSN in order to reduce the adverse effects associated with the existence of routing holes. Specifically, we propose to have the mobile nodes bridge the gaps created by the routing holes, therefore, shortening the average hop distance of routing paths.

In Chapter 2, as a starting point for the theoretical analysis and proof of concept for our solution, we introduce a new routing anomaly which we term the *microhole*. Microholes are very small imperfections in routing paths and are not classified as routing holes by the traditional routing hole definition. We evaluate our approach with respect to microholes while using energy as the primary metric and show that it can provide a reduction in the overall energy consumption in certain WSNs deployment scenarios. We also introduce three criterions (*feasibility*, *justifiability*, and *effectiveness*) that need to be met in order for the mobile node to deploy to a microhole, or even to be considered for deployment as part of the network. We find that the above concepts are not limited to microholes, but serve as the groundbreaking work for 'traditional' routing holes.

In Chapter 3 we expand on the work done in conjunction with our colleagues, and apply our solution to a scenario where a routing hole is impacted by multiple, variable-rate, data streams. We determine the optimal node to which a mobile node needs to forward

data across the routing hole, and also find a way to optimally place a mobile node inside a routing hole with the help of a metric that combines energy and delay network performance. These theoretical components are integrated into the "**O**ptimal **Pla**cement of a **Mo**bile **N**ode – 2" algorithm which uses mobile nodes in order to bridge the gaps created by routing holes, shortening the average hop distance of routing paths, and thereby leading to reduced delays and energy consumption.

In Chapter 4 we simulate our theoretical findings as further proof of correctness and appropriateness of our technique and present. Finally, we conclude our work in Chapter 5 with a discussion and possible future directions.

Chapter 2

WSN Routing Microholes

2.1. Taxonomy of Routing Anomalies in WSNs

In the previous chapter we have looked into the scenarios and causes leading to the formation of topological/routing anomalies (i.e. routing holes) in large-scale WSNs. As was already pointed out, routing holes create sub-optimal, non min-distance routing paths, which consequently lead to prolonged transmission delays, increased energy consumption, reduced network lifetime and possible network partitioning due to failure of critical nodes. We have also indicated that the causes for routing holes can generally be classified into three main categories: a) *random node deployment*; b) *node failure*; c) *environmental and human-caused factors*.

In random node deployment, routing path inefficiencies occur simply because nodes are unlikely to be deployed in a way that completely excludes the possibility of local routing minimums. Furthermore, even if by chance no local routing minimums are present in the network, in order for the routing paths to be optimal, nodes would have to be perfectly lined up and equidistant from the source to the destination. The probability of the previously described scenario being true is extremely low, even in high density WSNs. Therefore, due to a degree of randomness involved in the deployment of WSNs, there will always be some inefficiency with respect to the routing paths.

Node failure is another possible cause for the creation or routing holes. Nodes may fail for a variety of reasons such as hardware manufacturing defects or bugs in the software. Nevertheless, a more likely reason for nodal failure is premature energy (i.e. battery) depletion. It should be noted that in most real world scenarios, traffic load distributions are non-uniform both in time and space; hence, different nodes experience different battery depletion times.

Environment and human factors can also contribute to the creation of routing holes. For example, terrain features (e.g. hills, lakes, buildings, etc.) can create undesirable nodal distributions, especially in the case of aerial network deployments. Once a WSN is deployed, environmental conditions such as storms, floods or fires may cause physical damage to some of the sensors, further contributing to the creation of routing holes. Not all damage done to WSNs is however unintentional or accidental. Nodes of a WSN

deployed in a war zone are likely to be attacked at every chance by enemy combatants in hopes of destroying the flow of useful information to their enemy's base. Shelling or air strikes are also likely to cause damage to a WSN infrastructure. Even in peace time, humans – due to their inquisitive or mischievous nature – may remove or destroy WSN nodes.

Some of the techniques for routing hole discovery and mitigation were described in Chapter 1. These techniques generally rely on the existence of local minimum(s), in the sense of geographic routing, to detect and map routing holes. While the reliance on local minimum(s) appears effective in detecting large-scale routing holes, it is not recommended when dealing with smaller-scale routing holes due to a high probability of 'false negatives'.

In the remainder of of this chapter we take a closer look at such smaller-scale routing inefficiencies that cannot be detected by the existing/known hole-discovering techniques. Specifically, we examine the overall effects of these holes on their respective WSNs, and propose new ways to detect and eliminate them with the use of mobile nodes.

2.2. Routing Microholes: When are they worth combating?

2.2.1. Definition of Microholes

In WSNs employing geographic routing, routing holes are generally manifested through the so-called ' local routing minima'. Algorithms such as TENT and BOUNDHOLE [40] may be used to distributively discover and map routing holes by identifying the points of local routing minima. Nevertheless, we argue that even in the absence of local minima in the sense of TENT or BOUNDHOLE algorithms, the routing paths of a WSN could still be riddled with inefficiencies. We call these inefficiencies *microholes* and define them subsequently. Please note, by the TENT rule, a node is considered a local routing minimum (i.e. a node on a boundary of a routing hole where a packet could possibly get stuck) if it forms an angle $\geq 120°$ with its neighbors. Also, for the clarity of our subsequent discussion, note that we assume the maximum possible communication range of all the nodes to be r. However, for the purpose of energy conservation, the nodes are able to adjust/reduce their radio range so that it perfectly matches the actual physical distance to the receiving node.

Definition 2.1 Microholes are areas on the routing paths that do not prevent correct geographic routing (i.e. do not contain local minima with respect to the main routing directions in the sense of TENT rule) but can still be optimized by augmenting the topology of the area in question.

We illustrate the concept of the microhole by referring to the network configurations shown in Figure 2.1. We assume that the maximum communication range of the nodes is r, and node S is the source (or a simple inflow/relay) node sending packets to destination D. The distance between SD is fixed and slightly greater than r ($r + \varepsilon$, where $\varepsilon > 0$), thus preventing their direct communication and forcing the information to travel through either one of the intermediate nodes, U or V.

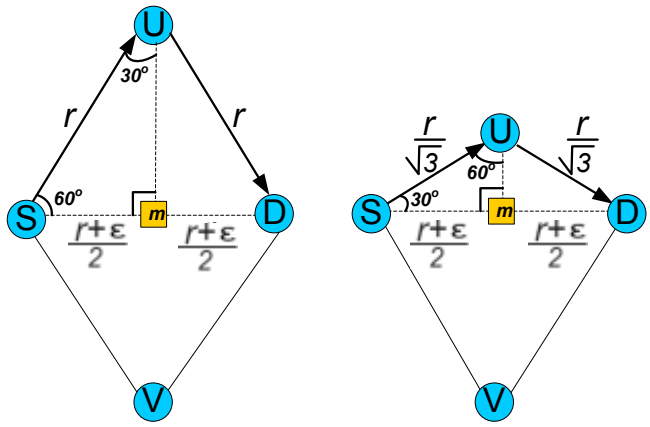


Figure 2.1 **(a)** Boundary case of a microhole, as nodes VSU form an angle of exactly 120°. A microhole of this form will benefit most from the deployment of a mobile node M (e.g. at the midpoint between S and D) due to the large difference between the pre and post deployment average hop lengths. **(b**) Nodal configuration in which the microhole (again) becomes a hole, as SUD now forms 120°

Now, referring to Figure 2.1.a), it can be seen that this topology corresponds to a 'boundary' case of a hole as the VSU angle is exactly 120°. It should be obvious from the presented figure that any inward displacement of node U (or upward displacement of node *V*) – up until the location outlined in Figure 2.1.b) – would imply a topology that

would not qualify as a 'hole', as none of the four observed nodes would form a 120° angle with its neighbours. Or, put another way, any intermediate nodal configuration between the two limiting cases presented in Figure 2.1.a) and 2.1.b) would fail to be identified as a network anomaly, according to the TENT rule. Still, we argue that any of these intermediate configurations would still be suboptimal in the routing (i.e. energy consumption) sense, as the average radio range of the two sending/forwarding nodes would be greater than the ultimately acceptable minimum: r/2. Clearly, to achieve the given minimum, the intermediate node (node *U* or a mobile node) would have to lie somewhere along the *SD* segment.

Theorem 2.1 In the case of a routing microhole as illustrated in Figure 2.1, placing a mobile node *m* at the midpoint of the *SD* segment would be optimal in terms of the overall energy consumption.

Proof: Examining Figure 2.2, in order to find the optimal position of the mobile *m* which minimizes energy consumption we let the distance from the source node S to the mobile *m* be equal x and the distance between the mobile node and the destination and the destination D be equal $(r - x)$. Assuming the energy model described in Section 1.2, we can express the energy consumption of this network configuration as follows:

$$E = e \cdot l + u \cdot l \cdot x^k + e \cdot l + u \cdot l \cdot (r - x)^k \quad (2.1)$$

Differentiating (2.1) and solving the resulting equation with respect to x, we find that the only real root result is:

$$x = \frac{r}{2} \tag{2.2}$$

Therefore, positioning the mobile bridge on the bisector of of the *SD* segment is the optimal course of action with respect to minimizing the nodal energy consumption.

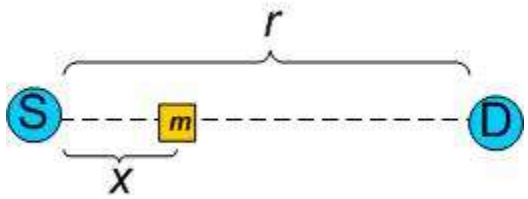


Figure 2.2 What is the energy optimal position of the mobile bridge *m?*

In the following section we will demonstrate that reduced hop distance can provide substantial energy savings, provided the transmission power can be adjusted at the sender.

2.2.2. Combating Microholes by means of Mobile Nodes: Energy Benefits

As was discussed in Chapter 1, the most energy expensive operation in data intensive WSNs is communication. Due to the nature of signal propagation (i.e. path loss), communication energy costs increase exponentially with relation to the distance

between the sender and the receiver. Therefore, we hypothesized that by reducing the hop distance with the help of mobile nodes we can achieve significant savings in terms of energy consumption throughout the lifetime of a WSN.

We demonstrate the potential benefits of the mobile node deployment by firstly examining the required transmission power of a sender with respect to its distance to the receiver under variable environmental conditions. For the purpose of this analysis we consider typical commercial sensor system parameters (Crossbow MPR500CA: 900 Mhz (868/916 MHz); Tx power: -20 to +5 dBm; Rx sensitivity: -98 dBm; Range: 152.4 m) [2]. Also, we use the standard Log-distance Path Loss model [41] in order to estimate the sender's *critical transmission power* (i.e. the minimum required transmission signal power of the sender given a particular sensitivity of the receiver), as shown below.

$$P_t(d)[dBm] \;=\; P_r[dBm] - G_t \;+\; \overline{PL}(d)[dB] - G_r \tag{2.3}$$

where,

P_t = transmitter power;
P_r = required signal power at the receiver;
G_r = receiver antenna gain;
G_t = transmitter antenna gain;
$\overline{PL}(d)$ = average path loss at a distance *d* from the transmitter, given by the following:

The actual expression for $\overline{PL}(d)$ is given in (2.4).

$$\overline{PL}(d)[dB] \;=\; \overline{PL_o}(d_o) \;+\; 10nlog_{10}\left(\frac{d}{d_o}\right) \tag{2.4}$$

where,

$$\overline{PL_o}(d_o) = 20\, log_{10}\left(\frac{4\pi d_o}{\lambda}\right) \tag{2.5}$$

In (2.5),

$\overline{PL_o}(d_o)$ = free space path loss at a reference distance d_o;
λ = wavelength of the carrier;
n = path loss exponent (dependent on the signal propagation environment). Sample values of n, as found in different mobile radio environments, are provided in Table 2.1.

Environment	Path Loss Exponent, n
Free space	2
Urban area cellular radio	2.7 to 3.5
Shadowed urban cellular radio	3 to 5
In building line-of-sight	1.6 to 1.8
Obstructed in building	4 to 6
Obstructed in factories	2 to 3

Table 2.1 Typical path loss exponents obtained in various mobile radio environments **[41]**.

Environment	Path Loss Exponent, n
Open	3.41
Wooded	2.35
Wooded & Hilly	2.90

Table 2.2 Averge path loss exponent results based on field measurement data of 915 MHz near ground wireless sensor nodes' radios randomly distributed in three naturally occurring environments **[42]**.

The results in Figure 2.3 are based on transmitter and receiver gains (G_t and Gr respectively) of 0 dBi, receiver sensitivity of -98 dBm, operating frequency of 900 Mhz and a number of different typical path loss exponents obtained in various mobile radio environments (see Table 2.1 and Table 2.2). We ignore shadowing since it is highly dependent on the particular environmental conditions.

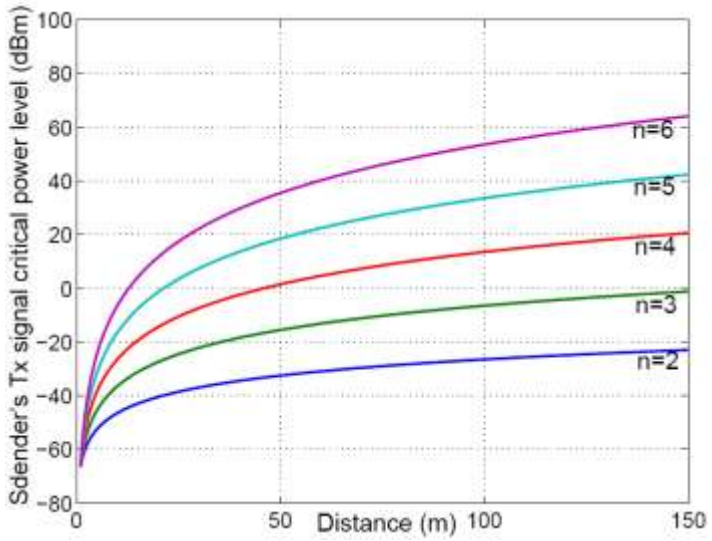


Figure 2.3 The Tx power required for signal detection in various environments by a receiver at a distance *d*, assuming log-distance path loss model without shadowing, constant Rx sensitivity of *Pr* = -98 dBm and path loss exponent *n*.

Figure 2.3 highlights the significant effects of distance and the environment on the critical transmission power. Namely, depending on the value of the path loss exponent, increasing the distance between the transmitter and receiver only twice requires an increase in the transmission power of approximately of 1 to 2 orders of magnitude[3]. We

[3] Since transmission power is represented in decibels, an increase of 20dBm would imply an increase by a factor of 100 for the power consumption when represented in milliwatts (mW).

consequently conclude that shortening the distance between the transmitters and the receivers could provide significant energy gains, especially in transmission intensive wireless networks.

The actual energy gains that may be achieved over the lifetime of a WSN by placing a mobile node in a microhole are illustrated in Figure 2.4. (Note, energy gain is calculated as the energy that is used by the sender and the forwarding node when the mobile node is not deployed, less the energy that is used by the sender and the relay mobile node once the mobile node is deployed.) For this particular case we assumed a microhole in which the hop distance will be reduced by 31.7%[4] when the mobile node is deployed. The parameters for our calculations are based on the Great Duck Island (GDI) bird habitat monitoring experiment, which is one of the most widely publicized and successful deployments of WSNs [43]. The GDI parameters used in our analysis include: the network lifetime, transmission rate and types of nodes used (i.e. node energy consumption was comparable).

[4] The 31.7% reduction in hop distance is the result of placing a mobile node in an "average" sized microhole. Referring to Figure 2.1, in a worst case microhole, angle DSU will be approaching 60°, and in 0° for the least severe case. Choosing a DSU angle of 30° will result in a hop distance reduction of 31.7%.

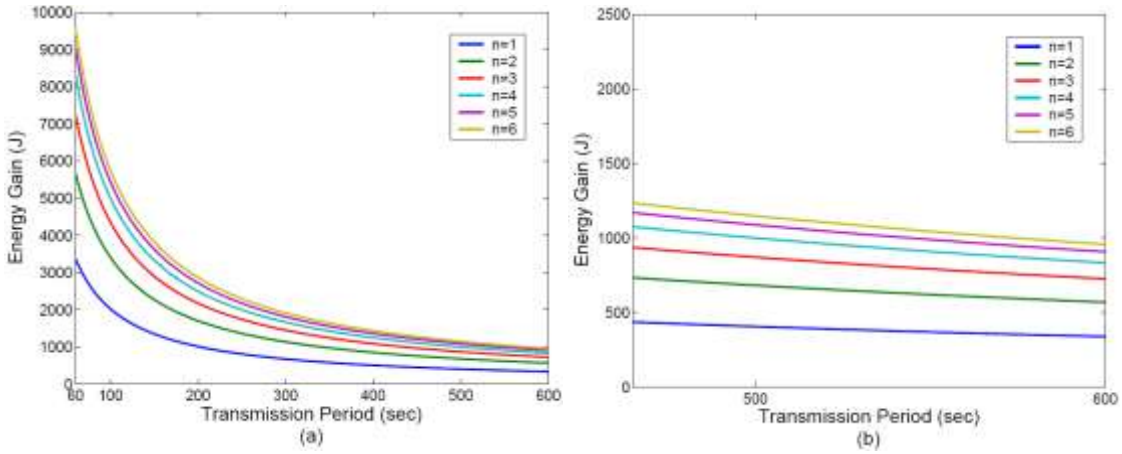


Figure 2.4 **(a)** Energy gained by deploying a mobile node at the optimal location inside a microhole over half a year lifetime. **(b)** Zoom in on (a).

We set the network lifetime to be half a year, the transmission rate 34.2 baud and message size 30 kB. As can be seen from Figure 2.4, we calculated the energy gains for a range of transmission periods and for various values of path loss exponent. We have also assumed our network to be more transmission intensive than in the GDI experiment by setting the packet size to 30 kB as opposed to 36 bytes.

There are two important observations about Figure 2.4:

1) As transmission period decreases (i.e. data is transmitted more frequently), energy gains increase, and vice versa. This is to be expected as a smaller transmission period implies that a greater amount of data is being forwarded by the mobile node during the lifetime of the network, leading to greater energy savings.

2) As n increases, greater energy savings are achieved. This leads to the following important point: due to the properties of signal propagation, shortening the hop distance by the same relative amount for networks communicating in environments with greater values of n produces larger energy savings than for networks communicating in environment with smaller values of n.

From the practical point of view, the savings outlined in Figure 2.4 appear to be quite substantial. For example, looking at the n=3 curve, and given that an average energy capacity of an AA battery is about 15 kJ [44], a node powered by only one such battery with a transmission period of 200 seconds will achieve a 17% lifetime extension over a six-month interval. Furthermore, if the transmission period is assumed to be 60 seconds, the node's lifetime is extended by nearly 50%!

2.3. Feasibility, Justifiability and Effectiveness of Combating Routing Microholes by Means of Mobile Nodes

If one is to contemplate the use of a mobile bridge in a real-world (micro)hole[5] scenario (as suggested in the previous section), the deployment would have to be *feasible*, *justifiable* and *effective*. By feasible we imply that a mobile node possesses the physical

[5] Please note, the discussion of this section is rather general, and it applies not only to the deployment of a mobile bridge inside routing microholes, but also to the deployment of a mobile bridge inside regular (large-scale) routing holes.

capacity (e.g. sufficient energy) needed to be moved to and be operating inside a routing hole. By justifiable we mean that the network benefits from the mobile's deployment (e.g. energy is conserved and as a consequence the network lifetime is extended). Finally, by effective we denote the probability with which a mobile node will be successful in meeting its mission goals. In our case, a mobile node will be considered effective given that it is deployed within a feasible and/or justifiable distance from a routing hole with a relatively high probability. In the reminder of this section, we discuss each of these three critical criteria in more detail.

2.3.1. Feasibility Evaluation

For the deployment of a mobile bridge inside a routing (micro)hole to be feasible, the following condition (2.6) has to be satisfied. To explain, condition (2.6) states that the total energy available to a mobile node $E_{battery}$ cannot be less than the energy required for a mobile to locomote (i.e. move) to the hole location (E_{move}) and plus the energy required for the mobile to operate inside the hole over a period of time P ($E_{op}(P)$), (see Figure 2.5).

$$E_{battery} \geq E_{move} + E_{op}(P') \quad (2.6)$$

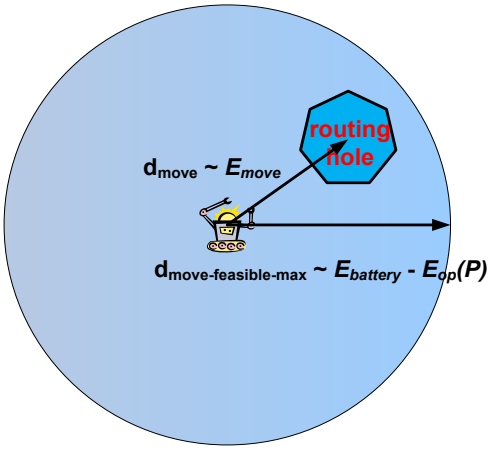


Figure 2.5 Two distances are shown: d_{move} is the distance the mobile node has to travel to a routing hole - E_{move} is directly proportional to this distance. $d_{move\text{-}feasible\text{-}max}$ is the maximum distance the mobile node can travel in order to be deployed while still retaining enough energy to operate at the routing hole for a required period of time ***P***.

E_{move} can be expressed as:

$$E_{move} = d_{move} \times e_{move} \tag{2.7}$$

where,

d_{move} = locomotion distance of a mobile node;
e_{move} = energy a mobile node consumes for locomotion per unit distance.

Similarly, $E_{op}(P)$ can be expressed as:

$$E_{op}(P) = P_{Rx} \times R^n \times P'(P) + a(P) \tag{2.8}$$

where

$P_{Rx} \times R^n = P_{Tx}$ represent the mobile's transmission power, assuming a fixed receiving power (P_{Rx}) and the mobile-to-receiver distance of R (for more see (2.3));

$P^{'}(P)$ is the fraction of time (i.e. fraction of observed interval P) that the mobile spends transmitting (clearly, $P^{'}$ a function of the total operational time P);

$a(P)$ accounts for the energy expenditure in states of idling, sleeping, sensing or processing over the period of time P.

By substituting (2.7) and (2.8) into (2.6), (2.6) gets transformed into (2.9).

$$E_{battery} \geq d_{move} \times e_{move} + P_{Rx} \times R^n \times P'(P) + a(P) \quad (2.9)$$

By rearranging (2.9) we obtain the following:

$$d_{move} \leq d_{move-feasible-max} = \frac{E_{battery} - P_{Rx} \times R^n \times P'(P) - a(P)}{e_{move}} \quad (2.10)$$

From (2.10), one can notice that the maximal feasible locomotion distance ($d_{move-feasible-max}$) of a mobile bridge is strongly dependent on two network/environment parameters: *R* – a parameter directly related to the distance between the mobile node and its respective receiving node and *n* - path loss exponent. Specifically, higher values of either of the two parameters would result in increased power consumption of the mobile node upon its deployment in the microhole – $E_{op}(P)$ (see (2.8)), leaving less energy for locomotion. Or, put another way, with a higher $E_{op}(P)$ a stricter bound would be placed on $E_{move-feasible-max}$ in order to satisfy (2.4),

ultimately resulting in shorter maximal feasible locomotion distance ($d_{move-feasible-max}$).

To put the above results/observations in the context of real-world WSN applications, let us assume that the mobile platform (i.e. robot carrying the mobile sensor) deployed is tank-style treaded Surveyor SRV-1 [35] robot (see Figure 2.6), as its dimensions are well suited for carrying a sensor node such as the Crossbow MPR500CA sensor.

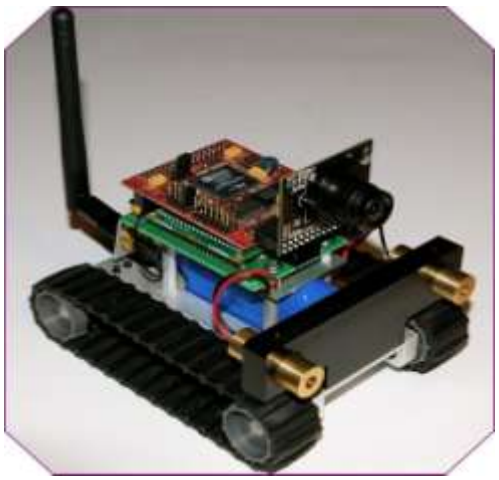


Figure 2.6 Surveyor SRV-1 Robot. Open Source Wireless Mobile Robot with Video for Telepresence, Autonomous and Swarm Operation **[35]**.

Based on the Surveyor SRV-1 specifications, we are able to estimate that energy capacity ($E_{battery}$) of its 7.2V 2Ah Lithium-ion battery pack is about 50,000 Joules and its energy consumption for movement (e_{move}) is about 10 Joules/meter. We also assume that the mobile sensor node needs to operate at the destination for half a year ($P = 15552000$ seconds), is required to send a 36 byte packet every hour after its deployment at the destination, and is deployed in an environment with a path loss

exponent *n*=2.5. Other parameters that we use in our calculation are the distance beween the mobile bridge and its respective receiver (i.e. node on the exiting side of the hole – see Figure 2.2) R=50m, $P^{'}(P)$= 36.4 seconds (calculated based on sensor node T_x data rate of 34.2 Kbaud), $P_{Rx} \cong 0.03$ Watt (based on MPR500CA specifications), and $a(P)$ = 746.5 Joules (based on MPR500CA processor and radio sleep energy consumption parameters).

Using the above parameters with (2.10) we get the following:

$$d_{move} \leq d_{move-feasible-max} = \frac{E_{battery} - P_{Rx} \times R^n \times P^{'}(P) - a(P)}{e_{move}}$$

$$= \frac{50{,}000 - 0.03 \times 50^{2.5} \times 36.4 - 746.5}{10} = 2005.1\ m$$

In other words, given the task this particular mobile node is required to perform at the microhole, its feasible travel distance to the destination at the microhole should not exceed ~2005.1 meters. This results suggests that from the practical point of view, the use of a mobile bridge for the purpose of combating routing (micro)holes can generally be considered feasible, as the diameter of the deployment field in most real-world WSNs is of the order of only few hundred meters to a few kilometers.

2.3.2. Justifiability Evaluation

Assuming that the feasibility constraint (2.6) is satisfied, it is still possible that moving a mobile into the microhole is not justified. Specifically, if the energy consumed for the

mobile's locomotion well exceeds the energy gain $G_e(P)$ obtained by the mobile's deployment (where $G_e(P) =$ energy consumed if the mobile is not deployed – energy consumed with the mobile deployed) obtained by the mobile's deployment, then the justifiability of the entire operation may become questionable. For example, imagine the scenario in which the mobile consumes 99% of its energy to move to a particular location; yet, the savings it provides to the network upon its deployment are minimal if not negligible.

We formalize gain $G_e(P)$ as follows:

$$G_e(P) = E'(P) - E(P) \quad (2.11)$$

where,

$E'(P)$ = total energy expanded by nodes comprising a microhole in a period of time *P*;

$E(P)$ = total energy expanded in a period of time *P* after a mobile node is deployed to a microhole.

$$E'(P) = 2 \times \left((P_{Rx} \times (R')^n) \times P'(P) + a(P)\right) \quad (2.12)$$

$$E(P) = 2 \times \left((P_{Rx} \times R^n) \times P'(P) + a(P)\right) \quad (2.13)$$

The above (2.12) and (2.13) are calculated based on the microhole definition from Section 2.2. R' and R are microhole inter-hop communication distances prior and after the mobile node mobile node deployment accordingly. By substituting (2.12) and (2.13) into (2.11),

$$G_e(P) = 2 \times P'(P) \times P_{Rx} \times ((R')^n - R^n) \quad (2.14)$$

We quantify the 'justifiability criteria' in (2.15), by requiring that the energy consumed for the locomotion of the mobile be 'relatively comparable' to the gain obtained by the mobile's deployment at a particular location. The value of parameter k in (2.15) could take different values in different application environments; though, $k = 10$ seems rather reasonable for a wide range of scenarios. (With $k = 10$, we ask for the gain obtained by the mobile's deployment be at least 10% of the energy consumed by the mobile's locomotion.)

$$E_{move-justified} \leq k \cdot G_e(P) \tag{2.15}$$

In the above expression, $E_{move-justified} = d_{move-justified} \times e_{move}$. Based on this we can calculate maximum justifiable locomotion distance $d_{move-justified-max}$ as follows:

$$d_{move-justified-max} = \frac{G_e(P)}{e_{move}} \tag{2.16}$$

By employing expression (2.14), we expand (2.16) into (2.17).

By substituting (2.14) into (2.16),

$$d_{move-justified-max} = \frac{2 \times P'(P) \times P_{Rx} \times ((R')^n - R^n)}{e_{move}} \tag{2.17}$$

From (2.17), $d_{move-justified-max}$ appears to be sensitive to changes in $P'(P)$, R, R' and n. Specifically, a greater transmission period $P'(P)$, larger difference between R'

and R or larger n, would all result in more significant energy gains arising from the mobile's deployment, and would provide for a large $d_{move\ -justified\ -max}$.

In general, it would be desirable to have $d_{move\ -justified\ -max} \leq d_{move\ -feasible\ -max}$ (see Figure 2.7), as in that case any practically feasible deployment of the mobile bridge would also (i.e. automatically) be justifiable.

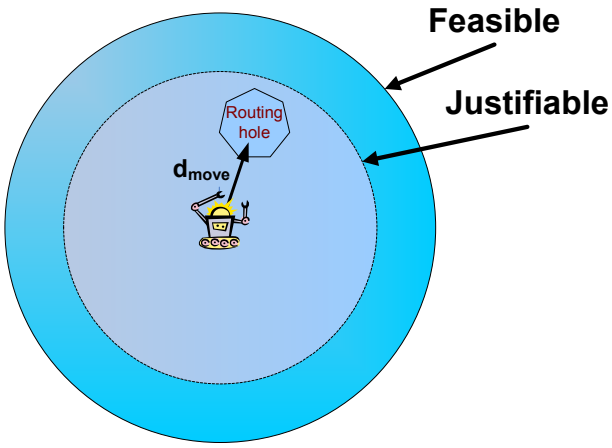


Figure 2.7 Shown is a routing hole that is located within the justifiable and feasible locomotion ranges of the mobile.

For example, using parameters from the Feasibility Evaluation section, and assuming that $\frac{R'}{2} = R = \frac{50}{2}$ (since mobile node is placed on the bisector between the sender and the receiver in order to maximize energy gain), we can calculate $G_e(P)$ as follows:

$G_e(P) = 2 \times P'(P) \times P_{Rx} \times ((R')^n - R^n) = 2 \times 36.4 \times 0.03 \times (50^{2.5} - 25^{2.5}) =$ 31.8 kJ.

Knowing $G_e(P)$ we can calculate $d_{move\ -justified\ -max}$:

$$d_{move\ -justified\ -max} = \frac{G_e(P)}{e_{move}} = \frac{31.8\ \text{kJ}}{10\ \text{J/m}} = 3180\ \text{meters}$$

In other words, the mobile is justified to travel a maximum of 3180 meters before the energy it uses during its travel to the routing hole exceed the gains attained from having it deployed. It is also worth noting that in this case

$d_{move\ -justified\ -max} \geq d_{move\ -feasible\ -max}$ [6] , which contradicts the desideratum that $d_{move\ -justified\ -max}$ should be $\leq d_{move\ -feasible\ -max}$. In practical terms this implies that the energy capacity of the mobile node should be expanded since the gains outweigh the costs.

2.3.3. Effectiveness Evaluation

To determine whether deploying a mobile node to a particular microhole will be feasible and justifiable is only possible if both - the state of mobile's battery supply and its precise distance from the given microhole - are known. In some cases, however, it may be important to evaluate the general effectiveness of a mobile node deployment without focusing on any particular microhole (see Figure 2.8). Specifically, it may be useful to estimate the probability with which a given (single) mobile is capable of repairing all/any of the possibly occurring topological inefficiencies, i.e. microholes, in an observed network.

[6] Previously, in the Feasibility Evaluation section, $d_{move\ -feasible\ -max}$ was calculated to be 2005.1 m

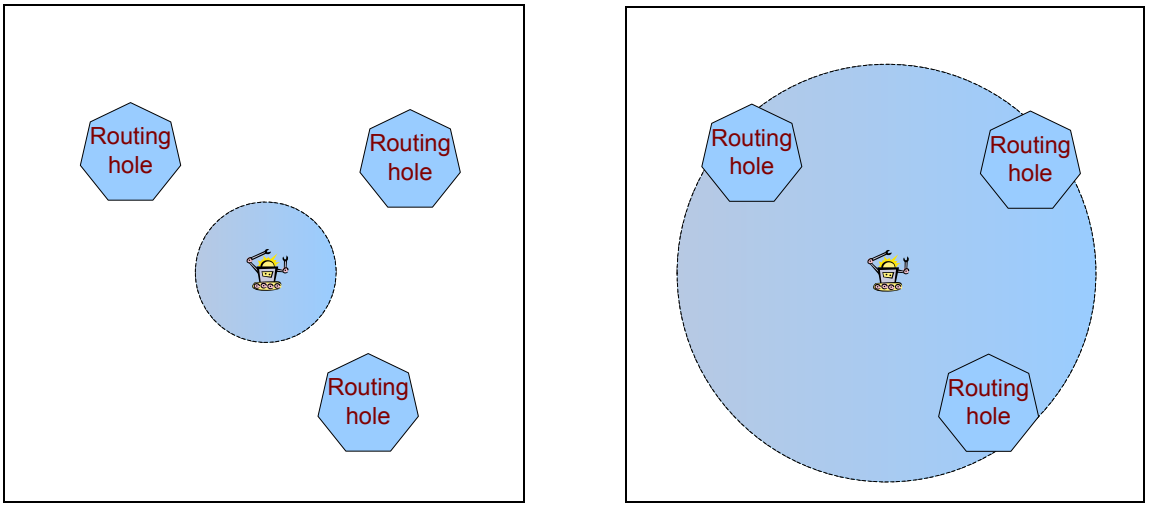


Figure 2.8 **(a)** Mobile node deployed in a WSN is unable to reach any of the routing (micro)holes since its feasible range is too small in relation to the WSN routing hole density. **(b)** In this case, there are a number of routing (micro)holes that are within the mobile node's feasible range.

In order to determine this probability, we make the following assumptions: The observed WSN consists of $N \times N$ static nodes uniformly distributed over an area of size $A_{WSN} = (N \times R)^2$(see Figure 2.9). The radio range of individual sensor nodes is R. A single mobile node exists in the network.

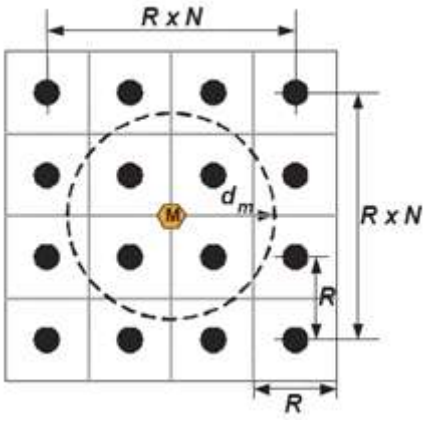


Figure 2.9 WSN deployment with $N \times N$ static nodes having a node density of $= \frac{1}{R^2}$. The mobile node M is in the interior of the network, so its maximum locomotion area $\boldsymbol{\pi} \times \boldsymbol{d_m^2}$ is entirely contained within the borders of the network.

Under the given assumptions, the probability with which a given (single) mobile is capable of repairing all of the possibly occurring network microholes corresponds to: 1) the probability with which only a single static node fails (i.e. a single microhole is created), AND 2) the probability that this microhole is within the mobile's feasible locomotion distance. Note, 1) is needed because if more than one microhole gets created in the network, regardless of their location, the mobile could be deployed to at most one of them, leaving the other microholes unrepaired. As for 2), even if only one microhole is found in the network, but it happens to be at a distance greater than $d_{move-feasible-max}$ from the mobile, the given distance will preclude the mobile from being deployed, again leaving the given michrohole unrepaired.

We summarize the above with the following expression:

$$P_{mobile-effective} = P_{one-node-fails} \times P_{node-near-mobile} \quad (2.18)$$

where,

$P_{one-node-fails}$ = probability that exactly one static node fails in the network (i.e. exactly one microhole is created);

$P_{node-near-mobile}$ = probability that the single and the respective microhole is within the locomotion distance from the mobile. (Here, the locomotion distance represents $d_{move-feasible-max}$.)

The actual expressions for $P_{one-node-fails}$ and $P_{node-near-mobile}$ are given in (2.19) and (2.20).

$$P_{one-node-fails} = N^2 \times \left(p_{fail} \times \left(1 - p_{fail}\right)^{N^2-1}\right) \quad (2.19)$$

$$p_{node-near-mobile} = \frac{d_{move}^2 \times \pi}{\left((N-1) \times R\right)^2} \quad (2.20)$$

In (2.19), p_{fail} is the probability of a node failure. (We assume this probability to be the same for all the nodes in the network.) As such, p_{fail} also represents the probability that a microhole gets formed in the network.

In (2.20), $P_{node-near-mobile}$ is obtained simply as the ratio of the mobile's feasible locomotion area and the total area covered by the network.

By substituting (2.19) and (2.20) into (2.18), (2.18) becomes:

$$P_{mobile-effective} = N^2 \times \left(p_{fail} \times \left(1 - p_{fail}\right)^{N^2-1}\right) \times \frac{d_{move}^2 \times \pi}{\left((N-1) \times R\right)^2} \quad (2.21)$$

Based on the above expression, we may draw several conclusions regarding the effectiveness of a mobile node/bridge deployment.

1) Examining (2.21) it becomes apparent that any increase in the distance that a mobile can move (d_{move}), would cause the probability of the mobile node effectiveness to increase as well. From the practical point of view, this effect could be achieved by supplying the mobile node with batteries of greater capacity.

2) An increase in the distance between network nodes (R) will clearly cause $P_{mobile\ -effective}$ to decrease. This can be easily explained by the fact that an increase in distance between network nodes implies a larger overall network field - of course, assuming all other parameters, including the number of nodes, remain the same.

3) The number of nodes in the network is represented by parameter N. Given the highly non-linear dependence between N and $P_{mobile\ -effective}$, it is difficult to intuitively evaluate the effects of this parameter on $P_{mobile\ -effective}$. Instead, we resort to evaluating the given dependence graphically – see Figure 2.10.

Figure 2.10 clearly demonstrates that increasing the number of nodes decreases the effectiveness of a mobile node. This is reasonable since increasing the number of nodes in the network field will increase the probability that several of these nodes will fail

within the justifiable distance of the mobile node, and as it is only able to fix one such failed node, the overall mobile node effectiveness will go down.

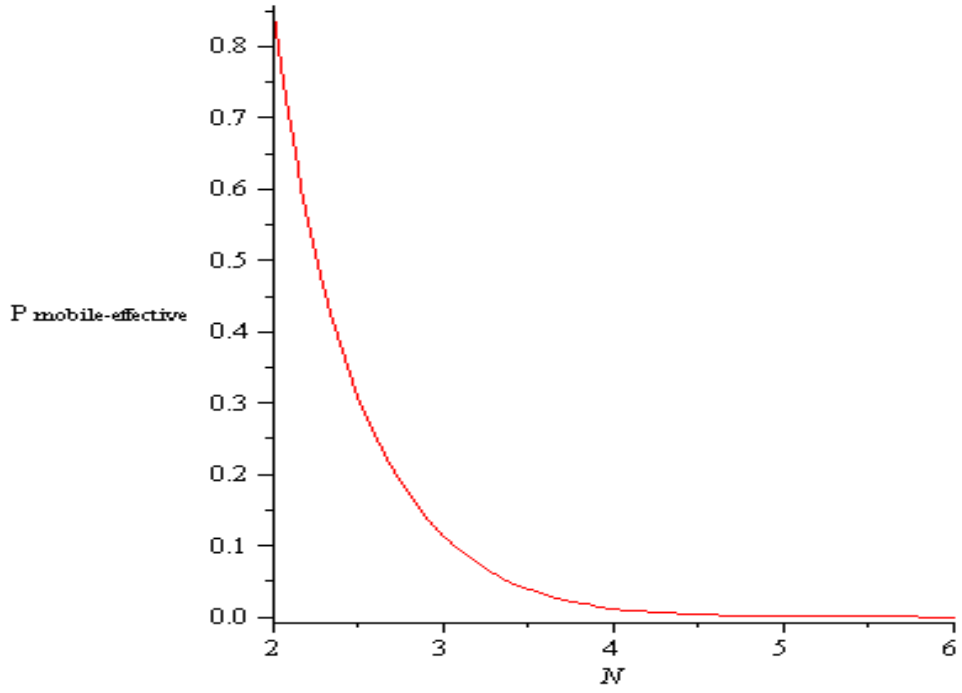


Figure 2.10 Mobile node effectiveness versus the number of nodes in the network.

4) Similarly to N, the effects of node failure on $P_{mobile\ -effective}$ do not lend themselves to simple analysis as p_{fail} appears in several places in the (2.21). Hence, we again resort to a graphical analysis. Figure 2.11 is obtained by plotting $P_{mobile\ -effective}$ as a function of p_{fail} while keeping other parameters fixed. Figure 2.11 shows that initial increase in p_{fail} is followed by increase in $P_{mobile\ -effective}$. However, after the peak $P_{mobile\ -effective} = 0.95$ (at $p_{fail} = 0.25$) is reached, mobile node effectiveness

begins to decline. The supporting argument for this result can be seen as similar to increasing the number of nodes in the network area. At first, when p_{fail} is zero, no nodes will fail within the justifiable distance of the mobile node. As p_{fail} begins to increase it becomes more likely that a single node will fail within the justifiable distance of the mobile node, and therefore the mobile node's effectiveness will increase as well. As p_{fail} continues to grow, so does the mobile node effectiveness, until it reaches a peak. The peak signifies the point at which it is most likely that exactly one node will fail within a justifiable distance of the mobile node. Still, as p_{fail} continues to increase, it becomes more likely that more than exactly one node will fail within the justifiable distance of the mobile, and therefore its effectiveness takes a sharp downturn from this point on.

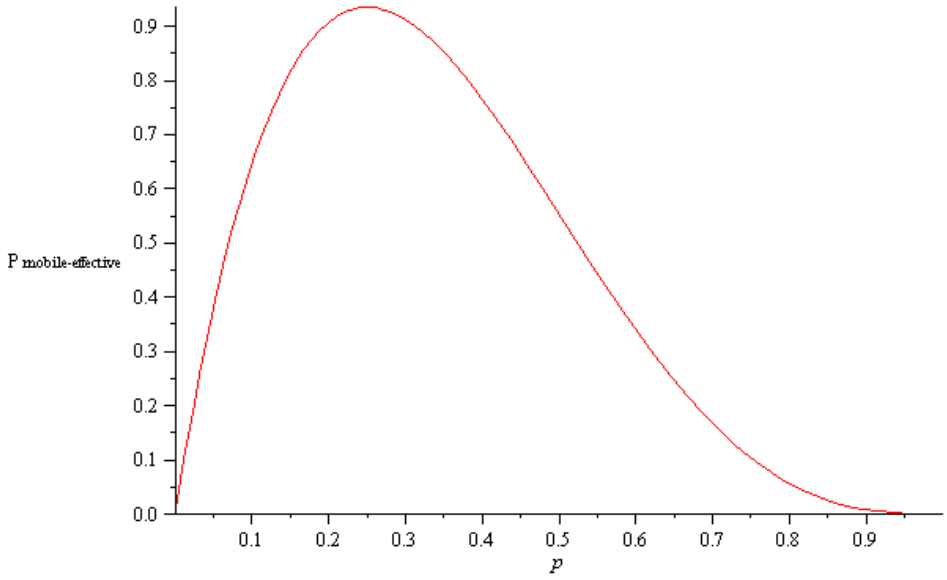


Figure 2.11 Mobile node effectiveness versus the probability of a node failure.

2.4. Summary

The goal of this chapter was to introduce the concept of microhole – smaller-scale topological inefficiencies possibly occurring in large-scale WSNs – and to evaluate the benefits and challenges associated with combating these inefficiencies by means of mobile nodes. Our analysis has shown that microholes found along data routes are wasteful in terms of network's energy consumption and can be optimized through the deployment of a mobile bridges – assuming the radio ranges of all involve nodes (the static sensors and the mobile bridge) are adjustable.

The concepts of feasibility, justifiability and effectiveness can be used for making decisions on whether a mobile node should deploy to a microhole or even deployed at all. It should be emphasized that these concepts, as defined in the preceding sections, are not limited to microholes but can also be applied (i.e. extended) to routing holes of different sizes and geometry. In the next chapter we take a closer look at such larger-scale routing holes, and extend our routing optimization technique to include these holes.

Chapter 3

WSN Routing Holes

3.1. Overview

In Chapter 2 we have introduced the concept of *routing microhole* and examined the cases in which these routing imperfections can be alleviated with the help of mobile nodes. In this Chapter, as a natural continuation of the work presented in Chapter 2 and [45], we move to analyzing the use of mobile nodes for the purpose of combating traditional large-scale routing holes (i.e. routing holes with multiple boundary nodes).

3.2. Summary of Work Presented in "Self-Healing Wireless Sensor Networks" [45]

In the analysis of Chapter 2, we used energy as the sole criterion for evaluating the feasibility and justifiability of a mobile node, i.e. bridge, deployment. We chose energy as the sole metric, since it is readily quantifiable, has a direct impact on the WSN efficacy and, more importantly, on the overall network lifetime. While energy seemed as a sufficient criterion when considering the deployment of mobile nodes to routing microholes[7], the question remained whether energy would be an appropriate criterion in an analogous discussion involving traditional routing holes. This matter was investigated in [45][8], resulting in some interesting conclusions. Namely, through formal and simulation-based analysis, it was shown that in certain cases, even when the deployment of a mobile bridge inside a routing hole seemed very intuitive and beneficial (e.g. in terms of transmission delays and hole confinement), the use of energy as the sole criterion completely failed to justify the mobile node deployment.

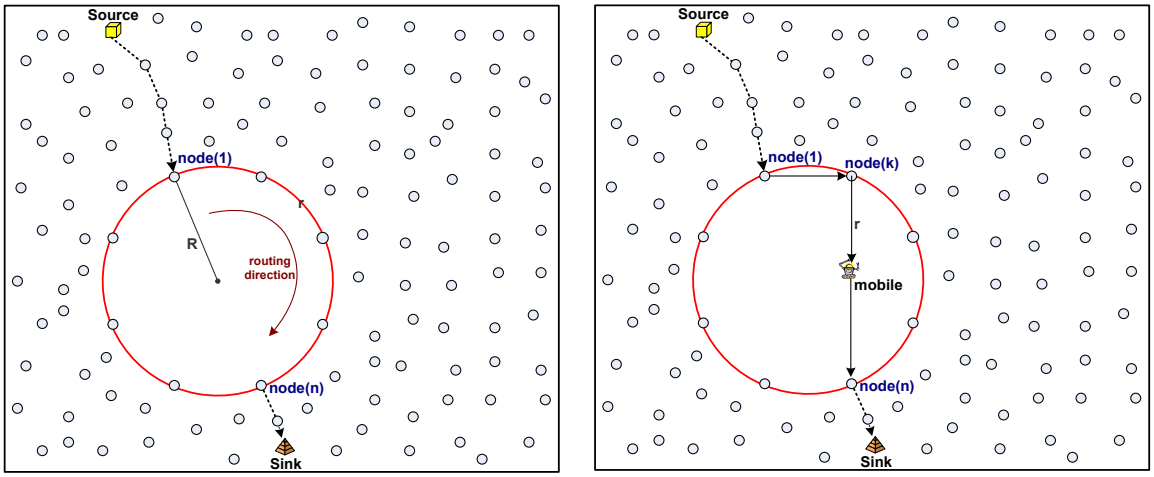
A more detailed overview of the most important contributions of [45] is provided below:

In the study of circular-shaped routing holes, an analytical expression for the energy

[7] Note: the deployment of a mobile node inside a microhole does not provide any real advantage in terms of the overall transmission delay, as the number of hops remains unchanged before and after the mobile's deployment – two hops in both cases.

[8] The author of this thesis was involved in the work presented in [52] by reviewing and assisting the principal investigators – N. Moniz and N. Vlajic.

consumption when a mobile bridge is not used, and data is forced to travel around the boundary of the hole (see Figure 3.1a) has been provided. Similarly, an analytical expression for the energy consumption when a mobile is used to bridge the hole (see Figure 3.1a) has also been provided. Similarly, we have formalized the energy consumption when a mobile is used to bridge the hole by connecting one of the boundary nodes routing the traffic and an arbitrary exit node (see Figure 3.1b). By comparing the energy consumption of the two scenarios, it has been proven that the use of a mobile node for the purpose of bridging a circular routing hole will never be energy justifiable[9], regardless of the actual size of the hole and/or the number of boundary nodes actively involved in routing.



(a) **(b)**

Figure 3.1 **(a)** Circular shaped routing hole **(b)** Circular hole bridged by a mobile node

[9] Specifically, the energy of routing data through a mobile bridge (and across a circular routing hole) will always exceed the energy of routing the same amount of data around the given hole.

Next, in [45], a similar evaluation of rectangular-shaped routing holes has been performed. a similar evaluation for routing holes of rectangular shape. Based on the obtained results it has been proven that for routing holes of nearly square shape (i.e. rectangle with equal sides), using a mobile bridge would not be energy justifiable, just as was the case with circular shaped routing holes.

Through extensive simulation, the theoretical result from [45] have been verified and further extended. Namely, it has been proven that deploying a mobile bridge to a wide range of uniform-like shaped[10] routing holes would never be justifiable if energy is to be used as the sole deployment criteria. Nevertheless, the simulations have shown promise in using a mobile bridge for the purpose of energy conservation in routing holes of elongated shape.

Another important contribution of [45] is the introduction of a new algorithm for **Op**timal **Pla**cement of a **Mo**bile **N**ode (OPlaMoN). This algorithm is aimed at finding the most effective deployment location (with respect to both energy and delay) of a mobile-bridge inside a routing hole of an arbitrary size and shape. OPlaMoN is "fully distributed, and it reaches the final solution through a cooperative decision-making process, assuming a minimum exchange of information among the affected nodes". In particular, according to the algorithm, each boundary node affected by an observed traffic stream calculates a so-called 'bidding value', which represents the overall energy/delay cost

[10] Uniform-like shaped holes are holes whose center of gravity is also its geometric center. Circle and square are two special cases of such holes.

associated with the deployment of the mobile-bridge near this specific node. Subsequently, through an iterative low-cost procedure, the nodes exchange their bidding values, which allows them to cooperatively identify the optimal bid and bidder – i.e. the place of the mobile bridge deployment that guarantees the lowest overall energy/delay cost.

In summary, the work described in [45] is the first of its kind, as it provides a thorough study and introduces unique solutions concerning the use of mobile nodes for the purpose of combating routing hole. Still, one of the major limitations of this work is the fact that it considers only scenarios in which a single routing hole impedes a single traffic stream. Most real-world WSNs, however, assume the existence of multiple data sources, thus implying the possibility that a single routing hole impedes multiple traffic streams, as shown in Figure 3.2.

To overcome this major limitation of [45], in the subsequent sections we investigate the use of mobile bridges in routing holes that obstruct multiple traffic streams of variable data rates (we name these *1-hole/1-mobile/n-variable-rate-traffic-streams* scenarios). Consequently, we introduce an improved version of OPlaMoN algorithm named OPlaMoN-2.

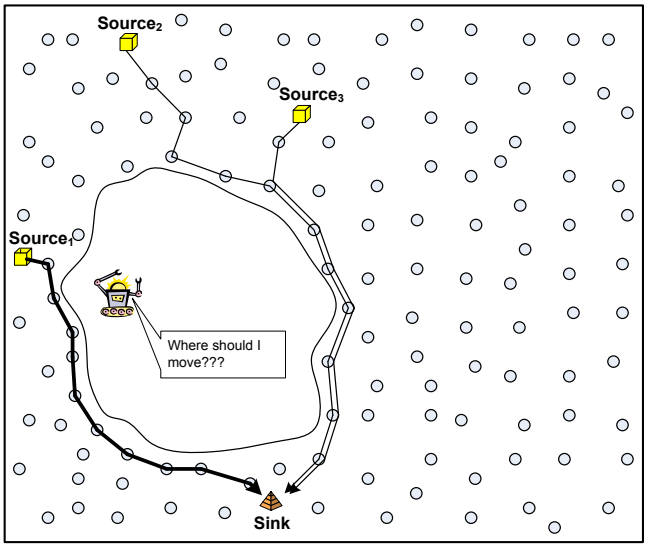


Figure 3.2 Multiple, multi-rate data streams are impeded by a routing hole. What is the optimal placement for the mobile node?

3.3. Combating Routing Holes: 1-Hole/1-Mobile/n-Variable-Rate-Traffic-Streams Scenario (OPlaMoN-2)

As already mentioned, the goal of this chapter is to extend the original OPlaMoN algorithm from [45] to deal with the possibility that multiple variable-rate data streams are impeded by a single routing hole. In order to achieve this goal, we introduce a number of modifications to the original algorithm, including:

- an improved calculation of nodes' bidding values (see Section 3.3.3) and,
- an improved search for the optimal exit node (see Section 3.3.2).

Before discussing each of the improvements in more detail, and in order to facilitate the reader's understanding of the material, we here provide a brief overview of the main phases of the new algorithm (OPlaMoN-2), when applied to a routing hole of an arbitrary shape and size.

1) **Routing hole identification phase:** Using the TENT rule and BOUNDHOLE algorithm [40], the boundary nodes of a given routing hole identify themselves to each other.
2) **Traffic stream estimation and dissemination phase:** During this phase, the inflow boundary node $nbs_k(1)$ [11] of each stream s_k (see Figure 3.3) estimates the data rate of its respective traffic stream(s) by monitoring the stream's intensity over a period of time T.[12] Subsequently, the estimated data rate and the ID of the associated inflow node are disseminated to all other boundary nodes.
3) **Bidding value calculation phase:** Each boundary node uses the information received from the inflow nodes in order to calculate its own bidding value. For more on the bid-calculation function employed by boundary nodes see Section 3.3.3.
4) **Bidding value exchange phase:** Through a distributed process involving the local (i.e. nearest neighbour only) exchange of bidding values, the boundary nodes cooperatively identify the highest bid node. The coordinates of this node are,

[11] The boundary node which is affected/impacted first by a traffic stream is termed *boundary inflow node.* More details on network assumptions and annotations can be found in Section 3.3.1.
[12] The optimal value of T will depend on the actual application scenario. The estimation of this parameter is part of our future work.

subsequently, forwarded to the mobile node, as they represent the optimal location of the mobile's deployment (refer to Section 3.3.3).

5) **Routing activation phase:** At this final stage, the mobile node moves to the optimal location of its deployment, and lets every inflow node know that it is ready to begin relaying data across the routing hole (see Figure 3.3). Each inflow node, then, decides whether it will be more advantageous to send its respective data stream through the mobile than to route it around the hole; and if so, the inflow node redirects its traffic stream towards the mobile.

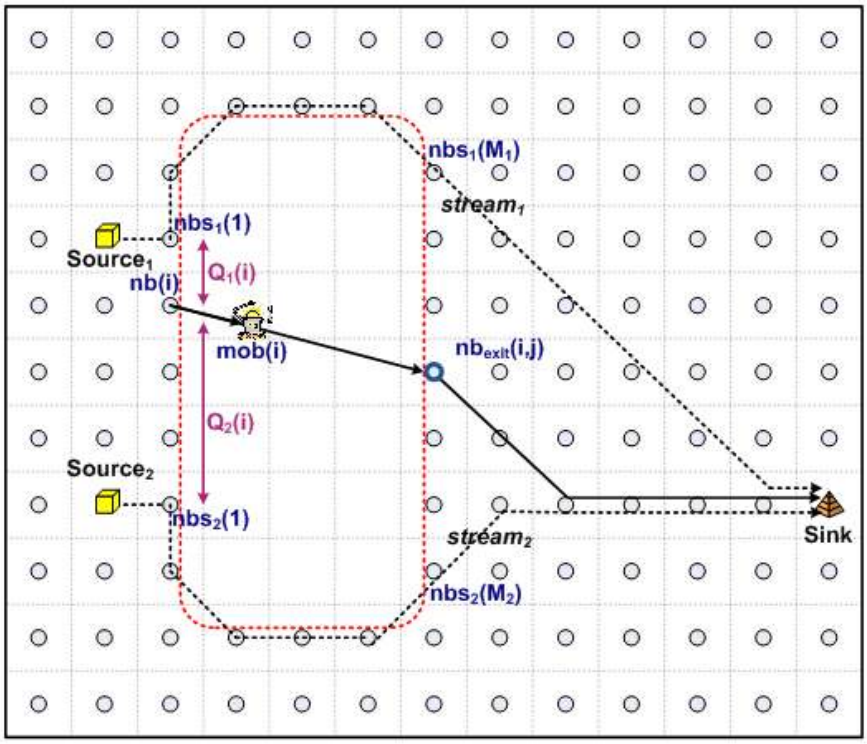


Figure 3.3 Example: Two data streams impeded by a routing hole. Streams may take the route through the mobile or flow around the boundary of the routing hole. To reach the mobile, the streams might need to traverse at least some of the boundary nodes.

3.3.1. Network Assumptions and Notation

Here we introduce some of the major assumptions concerning the network topology and operation, as used in our analysis:

- The observed wireless sensor network is organized into a communication grid, with every grid-cell containing a single randomly placed sensor node. Each sensor node has a fixed maximum communication range r [meters] which guarantees that the nodes located in the eight surrounding cells can always be reached. (Note, this nodal arrangement is generally known as the *virtual grid topology* [46].)
- All static nodes are identical in terms of their technical features and capabilities.
- k traffic streams (annotated by $s_1, s_2, \ldots, s_k$) are impeded by the same routing hole (see Figure 3.3).
- The routing hole is bordered by N boundary nodes $\{nb(i) \mid i = 1, \ldots, N\}$.
- When discussing the possible deployment of the mobile bridge next to boundary node $nb(i)$, $nb(i)$ is termed *candidate boundary node*. Put another way – node $nb(i)$ is a *candidate node*, if we consider deploying the mobile at most distance r from this node[13] and bridging the hole by passing the traffic between $nb(i)$ and an *exit node* via the mobile (Figure 3.3). [14]

[13] If a mobile node is placed at a distance greater than r from $nb(i)$, node $nb(i)$ will be out of the communication range with the mobile.

[14] Generally, the number of candidate boundary nodes can be reduced if the geometry of the routing hole is known. For instance, for the case of a rectangular routing hole, we may limit the candidate boundary nodes to the side of the hole where the data streams initially impact the routing hole. Positioning the mobile node anywhere on the other side of the hole would defeat the purpose of having it deployed.

- The boundary nodes affected by (i.e. involved in the routing of) traffic stream s_k are annotated by $nbs_k(1), .., nbs_k(M_k)$. Here, clearly, M_k corresponds to the total number of boundary nodes involved in the routing of stream s_k. (Again, see Figure 3.3.)
- The first boundary node coming into contact with a traffic stream is termed the stream's *inflow* node. In other words, the inflow node of stream s_k is the first boundary node which s_k 'touches' on its way towards the destination. Clearly, such a node is annotated by $nbs_k(1)$.
- Analogous to the above, the *outflow* node of a traffic stream is the boundary node at which this traffic stream leaves (i.e. exits) the routing hole on its way to the final destination (sink). The exit node of stream s_k is annotated by $nbs_k(M_k)$.
- In general, it is worth noting that two independent traffic streams impeded by the same routing hole may have completely different outflow nodes – see $nbs_1(M_1)$ and $nbs_2(M_2)$ in Figure 3.3. The actual locations and hop distance between such two outflow nodes will depend on:

1. the hop distance between the sources of s_1 and s_2 (Source$_1$ and Source$_2$ in Figure 3.3 respectively), and their distance from the hole;
2. the shape of the hole;
3. the relative position of the streams (s_1 and s_2) with respect to the hole;
4. the hop distance between the hole and the ultimate data *sink*.

3.3.2. Determining Optimal Exit Node

The ultimate goal of OPlaMoN-2 algorithm is to find the optimal deployment location of a mobile bridge in the interior of a routing hole – one that would minimize the overall energy and/or delay cost function. In general, this goal is accomplished by considering the placement of the mobile near each candidate node $nb(i)$ $(i = 1, \dots, N)$ and choosing the one that guarantees the lowest energy/delay cost. It is important to observe that the actual cost associated with placing the mobile next to node $nb(i)$ may vary depending on the selection of *exit node* $nb_{exit}(i,j)$.[15] (We use term *exit node* to annotate the boundary node to which the mobile bridge forwards data, as shown in Figure 3.4.)

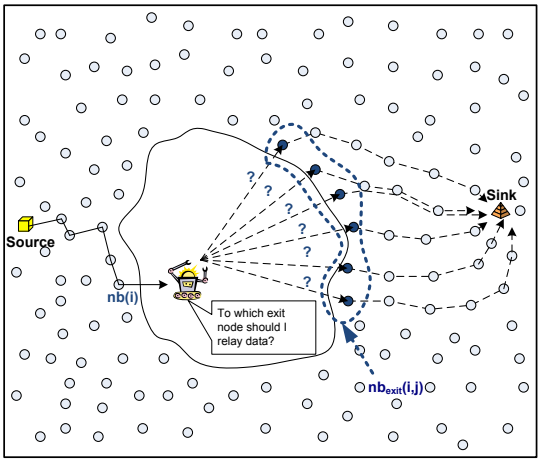


Figure 3.4 When considering mobile node deployment, node $\boldsymbol{nb(i)}$ needs to determine which exit node $\boldsymbol{nb_{exit}(i,j)}$ will minimize the cost transmission in terms of energy and delay.

[15] For simplicity, in [52], the outflow node of the single existing stream in the network was considered to be (the only viable) exit node. However, our analysis shows that such a simplistic assumption may lead to suboptimal results, and the exit node selection requires careful consideration of quite a few factors.

In cases when several potential *exit nodes* are considered, the cost of forwarding data through a particular *exit node* will depend on two key factors: 1) the actual transmission/physical distance between the mobile and the given exit node (this factor is especially important if *energy* is used as the more dominant cost function); 2) the number of hops between the given exit node and the sink (this factor, on the other hand, is highly critical if *delay* is used as the more dominant cost function).

Based on the above, it is clear that finding the optimal exit node emerges as an important and not so trivial sub-problem in the search for the best deployment location of a mobile bridge.

In the proceeding subsections (3.3.2a, 3.3.2b, and 3.3.2c) we take a closer look at the issue of determining the optimal exit node – the one that would optimize a particular cost function. For the purposes of this analysis, the assumptions stated in Section 3.3.1 are extended as follows:

- Assuming a fixed location of the mobile – next to node $nb(i)$ $(i = 1, \dots, n)$, the respectively considered exit nodes will be annotated with: $nb_{exit}(i,j)$, $j = 1, \dots, k$ where $k \leq n$.

- For simplicity of the discussion, the distance between the mobile and the exit node $nb_{exit}(i,j)$ is approximated with the distance between $nb(i)$ and $nb_{exit}(i,j)$, and annotated by $A(i,j)$. More specifically, $d_{Euclid}\left(mob(i), nb_{exit}(i,j)\right) \approx d_{Euclid}\left(nb(i), nb_{exit}(i,j)\right) = A(i,j)$ (see Figure 3.5).

- The distance between $nb_{exit}\,(i,j)$ and the *sink* is annotated by $B(i,j)$, (see Figure 3.5).

At this point we should also emphasize that our subsequent analysis focuses on routing holes of rectangular shape and arbitrary dimensions $p \times q$, were p and q could take any real value in the domain (0,+∞), as illustrated in Figure 3.5. Although this assumption may seem somewhat limiting, we believe that the general principles and results of our analysis could readily be extended to holes of other shapes.

Under the assumption of rectangular-shaped routing holes, we introduce three further (auxiliary) parameters, aimed at facilitating the proceeding analysis:

- **a** corresponds to the perpendicular distance between the side where the candidate node $nb(i)$ is located and the side where the exit node $nb_{exit}\,(i,j)$ is located. Thus, for any such pair of candidate-exit nodes, the perpendicular distance between their respective sides will correspond to the width of the hole (which will be p or q, depending on the alignment of the hole in the network).
- **b** is the perpendicular distance from the sink to the side of the hole on which $nb_{exit}\,(i,j)$ is located.
- **c** is the perpendicular distance from $nb(i)$ to the line used to define **b**.
- α represents the angle between **b** and the line $B(i,j)$ that connects the sink and $nb_{exit}(i,j)$.

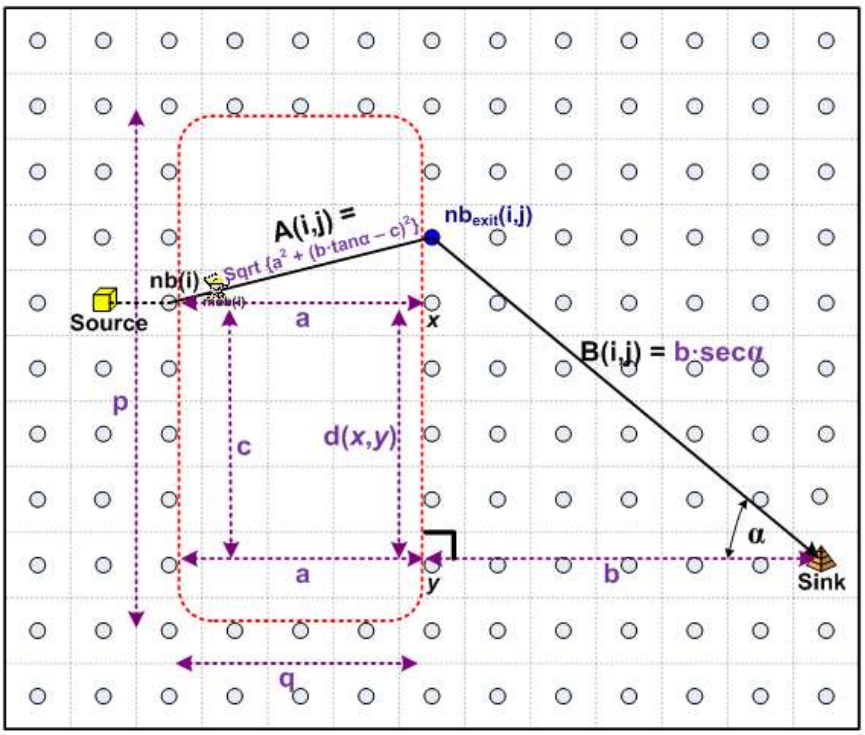


Figure 3.5 The above figure shows the auxiliary parameters introduced and used for the purpose of finding the optimal exit node

3.3.2 a) Calculation of Optimal Exit Node in case of Delay-Cost Function

We define the delay cost function $Delay(i,j)$ as the time it takes to route a unit of data from the mobile node deployed besides the candidate node $nb(i)$ up to the sink, through the exit node $nb_{exit}\,(i,j)$. Under the assumption that all nodes (static and mobile) are identical in terms of their transmission rates and processing speeds, while propagation delays are generally negligible, we can represent the delay cost function as

a simple linear function of the hop distance (i.e. number of hops) between the mobile and the sink[16]:

$$Delay(i,j) \approx HopDist(nb(i), nb_{exit}(i,j), Sink) = 1 + k \cdot \frac{B(i,j)}{r} = 1 + k \cdot \frac{b \cdot \sec\alpha}{r} \quad (3.1)$$

In (3.1), k is an empirically determined constant that accounts for the relative randomness of node distribution along $B(i,j)$.

To find the optimal exit node for a fixed $nb(i)$, we need to find the minimum of $D(i,j\,|i = const)$. Clearly, the minimum of $D(i,j\,|i = const)$ will be obtained by minimizing the respective $B(i,j)$, i.e. by choosing $nb_{exit}(i,j \mid i = const)$[17] physically closest to the *sink*. In Figure 3.5, the node annotated with ***y*** represents such a node.

Lemma 3.1: For a rectangular hole similar to the one represented in Figure 3.5, the delay associated with the routing of packets across the hole (via a mobile bridge) and to the sink will be minimized by choosing $\boldsymbol{nb_{exit}(i,j)}$ to be the boundary node that assumes the shortest distance to the sink.

3.3.2 b) Calculation of Optimal Exit Node in case of Energy-Cost Function

Unlike the delay, the overall energy required to route a packet from the mobile node deployed next to node $nb(i)$ up to the sink via the exit node $nb_{exit}(i,j)$ - which we

[16] (3.1) can easily be converted into proper delay units if just multiplied by the time it takes to send a unit of data over a hop. We omit this step in our analysis, as we are only interested in examining how delays compare to each other, and not in their actual numerical values.

[17] Wherever one specific candidate node $nb_{exit}(i,j)$ is considered, $i = const$ is assumed in respective equations.

annotated with $Energy(i,j)$ - depends not only on the number but also on the length of individual hops found along the route. Using the energy model described in Chapter 1 and basic trigonometry of the rectangular hole illustrated in Figure 3.5, $Energy(i,j)$ can be approximated with the following expression[18]:

$$Energy(i,j) \approx A(i,j)^2 + \left(\frac{k \cdot B(i,j)}{r}\right) \cdot r^2 = A(i,j)^2 + k \cdot r \cdot B(i,j) \tag{3.2}$$

Here, it is important observe: if we assume a known/fixed candidate node $nb(i)$, then for each considered exit node $nb_{exit}(i,j \mid i = const)$ angle α will take a different value. Consequently, we can rewrite (3.2) as

$$\begin{aligned} E(i,j|i = const) = \; & E(i,\alpha|i = const) = \\ & = \sqrt{(a^2 + (b \cdot \tan\alpha - c)^2)}^2 + k \cdot r \cdot b \cdot \sec\alpha = \\ & = b^2 \cdot \tan^2\alpha - 2bc \cdot \tan\alpha + k \cdot r \cdot b \cdot \sec\alpha + a^2 + c^2 \end{aligned} \tag{3.3}$$

Furthermore, the values of parameters **a**, **b**, and **c** in (3.3) remain unchanged, regardless of the actual exit node $nb_{exit}(i,j \mid i = const)$ observed. Consequently, based on (3.3), angle α turns out to be the only optimization variable of the objective function $E(i,\alpha|i = const)$. In order to find α, and its respective $nb_{exit}(i,j \mid i = const)$, that optimizes (i.e. minimizes) $E(i,\alpha|i = const)$, we employ the standard first-derivative rule, as shown in (3.3).

[18] (3.2) can easily be converted into proper energy units if just multiplied by the energy it takes to send a unit of data over a hop. However, this step is omitted from our analysis, as we are only interested in examining how energy consumption costs compare to each other, and not in their actual numerical values.

$$\frac{\partial E(i,\alpha|i=const)}{\partial \alpha} \approx$$
$$(2 \cdot \mathrm{b}^2 \cdot \tan\alpha) \cdot (1+\tan^2\alpha) - 2 \cdot b \cdot c\,(1+\tan^2\alpha) + k \cdot r \cdot b \cdot \frac{\sin\alpha}{\cos^2\alpha} = 0 \quad (3.4)$$

Note, we confirm that $E(i,\alpha|i=const)$ is a convex function and thus can be minimized by plotting (3.3) with **α** as a variable**,** while keeping the other parameters constant – see Figure 3.6.

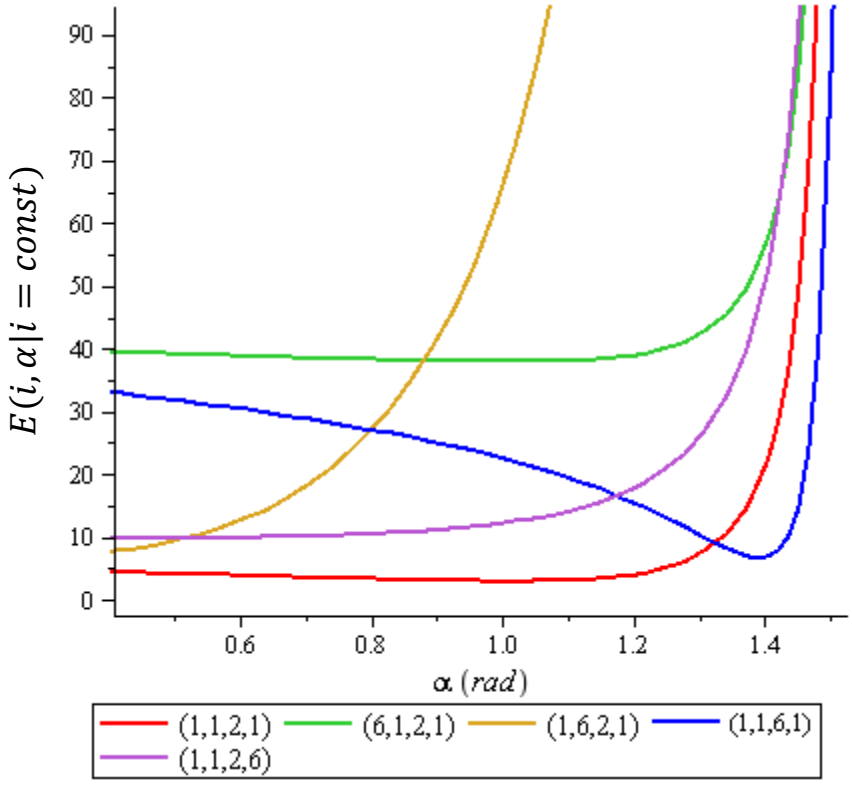


Figure 3.6 The function plots produced from (3.2) all clearly have a single minimum when **α** is within the range of $[0...\frac{\pi}{2})$. The legend values in parenthesis correspond to energy cost function parameters (**a**, **b**, **c**, **r**).

Unfortunately, it turns that that (3.3) is not solvable in a closed analytical form. Still, it may be noticed that expression (3.3) does not have **a** as a parameter, which implies that

the optimal angle **α** is independent of the width of the hole. We summarize this observation in the following lemma:

Lemma 3.2: The location of the exit node that optimizes (i.e. minimizes) the energy cost function does not depend on the width of the observed rectangular hole.

Next, in order to examine how the other parameters (**b**, **c**, and **r**) affect the optimal exit node, we first solve (3.4), numerically using Maple using Maple [47], and then plot the obtained solutions. The generated plots imply a few interesting conclusions.

In Figure 3.7, the position of the optimal exit node $\left(nb_{exit}\,(i,j \mid i = const)\right)$ is plotted as a function of **b** and **c**. The plot is generated using the Maple following command:

$$plot3d\left(\frac{1}{c}(b\cdot\tan(solve(\ (diff\,(e(\text{alpha},5,b,c,1),\text{alpha})=0),\ \text{alpha}\ ))),b=2..10,c=2..10\right);$$

where *e* is,

$$e:=(\alpha,a,b,c,r)\rightarrow b^2\tan(\alpha)^2-2\,b\,c\tan(\alpha)+\frac{r\,b}{\cos(\alpha)}+a^2+c^2$$

To explain, the above finds the optimal **α** from (3.4) for different values of **b** and **c**, assuming fixed **r**=1, **a**=5 and **k**=1. The obtained (optimal) **α** is subsequently used to find

the corresponding (optimal) $nb_{exit}\,(i,j \mid i = const)$ – simply as $(nb_{exit}\,(i,j \mid i = const) = b \cdot \tan(\boldsymbol{\alpha})$ (see Figure 3.5). Finally, in Figure 3.7, such optimal $nb_{exit}\,(i,j) \mid i = const$ is plotted against parameters **b** and **c**. Note, to simplify the discussion, in Figure 3.7 the actual location of optimal $nb_{exit}\,(i,j \mid i = const)$ is shown as a fraction of distance **d(*x*,*y*)** - between point ***x*** (point directly across $nb(i,j)$ on the opposite side of the routing hole) and point ***y*** (the point where line **b** impacts the routing hole), where **d(*x*,*y*)** = **c** as seen in Figure 3.5. 0% **d(*x*,*y*)** would mean the exit node is located at point ***y***, while 100% **d(*x*,*y*)** (1.00 on the graph) would imply the exit node is located at point ***x***.

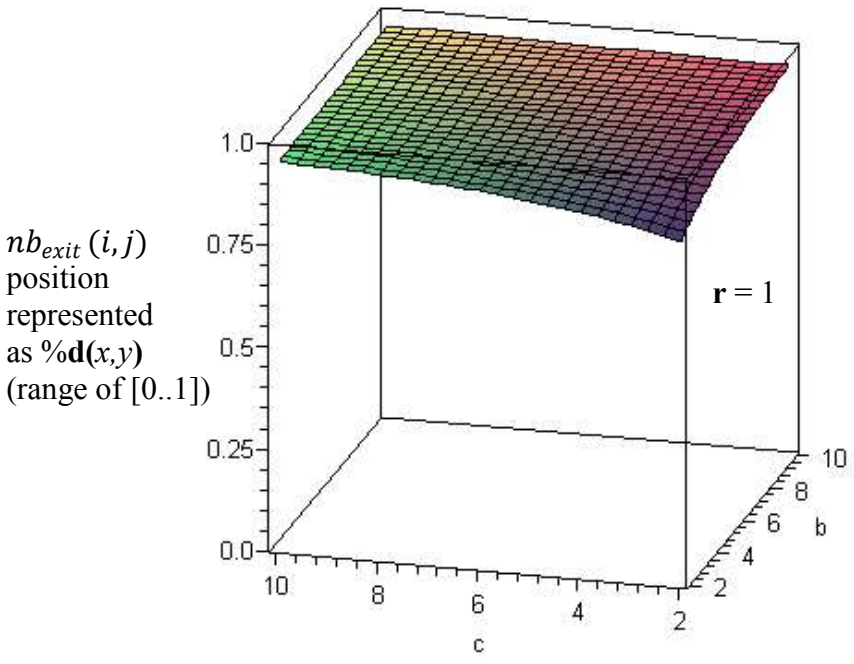


Figure 3.7 Position of optimal exit node for different values of **c** and **b**, assuming **r**=1, **a**=5 and **k**=1.

Figure 3.7 clearly demonstrates that, regardless of the actual values of **c** and **b**, the optimal exit node $nb_{exit}\,(i,j \mid i = const)$ tends to be very close to point ***x***. In other words, for any particular candidate node (i.e. any possibly considered position of the mobile node), the respective optimal exit node generally tends to minimize the distance that the signal has to travel across the routing hole.

Next, we are interested in examining how changes to the value of **r** affect the position of the optimal exit node. The plot in Figure 3.8 shows the position of optimal $nb_{exit}\,(i,j \mid i = const)$, for **r**=6 and other parameters kept at their earlier values. Clearly, based on Figure 3.8, **r** appears to have little to no affect on the final result, except for very small values of **b** and **c.**

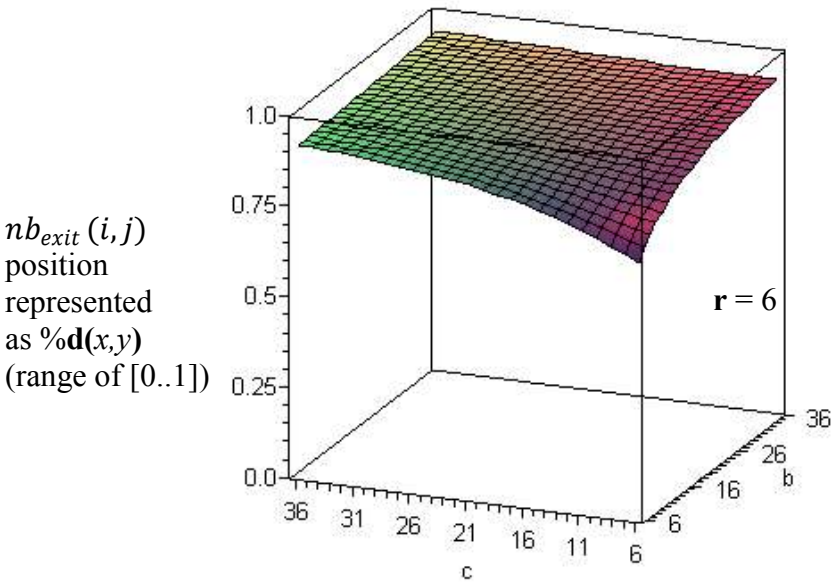


Figure 3.8 Increases in **r** do not appear to have any major effects on the optimal exit node position (relative to $\boldsymbol{nb(i)}$).

To further substantiate our above findings, the following three graphs are generated from (3.4) (see Figure 3.9). In these graphs, the y-axis represents the position of the optimal $nb_{exit}\,(i,j \mid i = const)$ node, expressed as a fraction of **d(*x*,*y*)**. The values of parameters **r**, **b** and **c** are individually varied, while other parameters are kept constant.

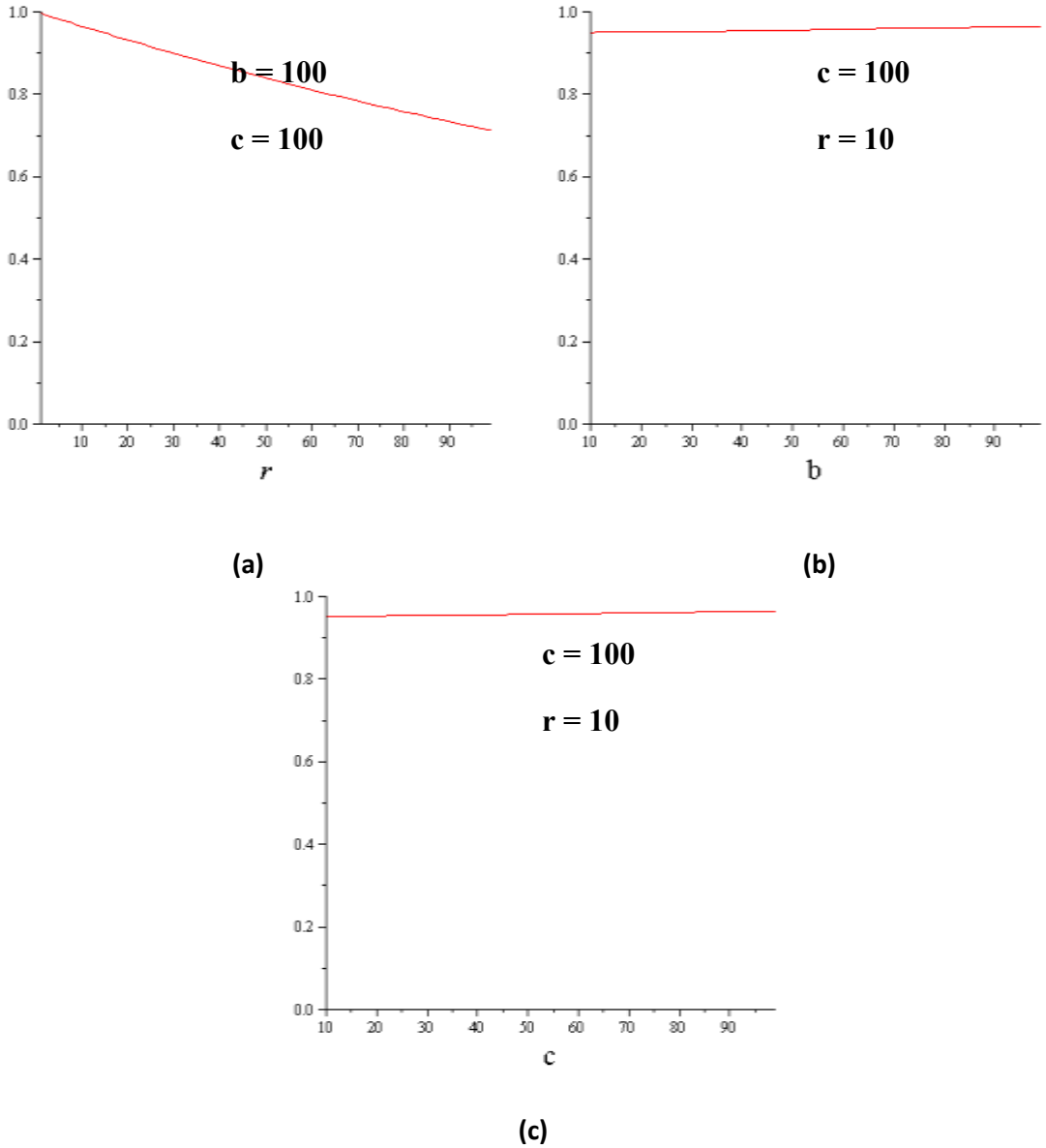


Figure 3.9 The three graphs indicate that increasing **a**, **b** or **r** will not have a significant effect on the optimal exit node position.

Figure 3.9b and Figure 3.9c clearly show that increases in **b** or **c** will only bring the optimal exit node position closer to $nb(i)$ (i.e. point ***x***). Figure 3.9a also supports this theory, but can be misleading at first look. Though it appears that with the increase in **r** the optimal exit node will be found farther and farther from $nb(i)$, (i.e. point ***x***), one must also take into consideration the value at which parameter **c** has been set. (Recall, **c** happens to correspond to the actual value of **d(**x,y**)**.) Hence, for example, under **c** = **d(***x*,*y***)** = 100 and **r**=90 there will be (approximately) only two nodes along **d(**x,y**)** - located at/near the actual points ***x*** and ***y***. Consequently, the optimal location of the exit node at 73% of **d(***x,y***)** from point ***y*** unquestionably falls closer to (and thus must be approximated by the position of) point ***x***.

The above observations regarding the optimal location of the exit node, from the perspective of the overall energy consumption, are summarized in the following theorem.

Lemma 3.3: For a rectangular hole similar to the one represented in Figure 3.5, the energy consumed for routing of packets across the hole (via a mobile bridge) and to the sink will be minimized if the exit boundary node assumes the shortest possible distance from the mobile bridge.

3.3.2 c) Calculation of Optimal Exit Node in case of Energy-Delay Cost Function

In Sections 3.3.2a and 3.3.2b, we have dealt with finding the optimal exit node that minimizes packet transmission delay and nodes' energy consumption independently from each other. In this section we are interested in combining these two objective

functions into a weighted cost metric $Cost(i,j)$ and then finding the optimal exit node that would minimize both – delay and energy consumption, at once. In other words, we are aiming at finding the best exit node in terms of Pareto optimality [48], such that would minimize both objective functions without disadvantaging each other, despite their indirect codependency.

To be able to combine the two objectives into one single cost function, we introduce an alternative representation of the nodes' energy consumption. Namely, by normalizing $Energy(i,j)$ with r^2 – which is assumed to be constant (see Section 3.3.1) – we obtain a function that still accurately quantifies the nodes' overall energy consumption, but now in terms of an equivalent number of transmissions over a hop of distance r. This new function is annotated as $E_{normalized}\ (i,j)$ (see (3.5)) and is guaranteed to be 'unitless', just as was the case with $Delay(i,j)$ (function).

$$E_{normalized}\ (i,j)\ \approx \frac{A^2(i,j)}{r^2} + \frac{k \cdot B(i,j)}{r} \tag{3.5}$$

The cost function that combines $E_{normalized}\ (i,j)$ and $Delay(i,j)$ ($Cost(i,j)$ is defined in (3.6)),

$$Cost(i,j) = w_1 \cdot E_{normalied}\ (i,j) + w_2 \cdot Delay(i,j) \tag{3.6}$$

where w_1, $w_2 \in [0, 1]$, and $w_1 + w_2 = 1$. In practical terms, *Cost(i,j)* as defined in (3.6), represents a weighted average of $E_{normalized}\ (i,j)$ and $Delay(i,j)$. At this point, we do

not assign any particular value to w_1 and w_2. Instead, we assume that these parameters could (i.e. would) be adjusted to suit each particular application scenario. Clearly, in time critical applications w_2 should be set to be greater than w_1. On the other hand, in applications where the longevity of the network is of primary concern, w_1 should be set to be greater than w_2.

By substituting equations (3.1) and (3.5) into (3.6), we get $Cost(i,j)$ in the following extended form:

$$\begin{aligned} Cost(i,j) &= w_1 \cdot \left(\frac{A^2(i,j)}{r^2} + \frac{k \cdot B(i,j)}{r}\right) + w_2 \cdot \left(1 + \frac{k \cdot B(i,j)}{r}\right) \\ &= w_1 \cdot \left(\frac{A^2(i,j)}{r^2} + \left(\frac{k \cdot B(i,j)}{r} + 1\right) - 1\right) + w_2 \cdot \left(1 + \frac{k \cdot B(i,j)}{r}\right) \\ &= w_1 \cdot \left(\frac{A^2(i,j)}{r^2} - 1\right) + (w_1 + w_2) \cdot \left(1 + \frac{k \cdot B(i,j)}{r}\right) \end{aligned} \tag{3.7}$$

Due to the assumption $w_1 + w_2 = 1$, and by adequately substituting $\boldsymbol{A}(i,j)$ and $\boldsymbol{B}(i,j)$ with the auxiliary parameters **a**, **b** and **c**, (3.7) further becomes:

$$\begin{aligned} C(i,j|i = const) &= C(i,\alpha|i = const) = \\ &= 1 + \frac{k \cdot B(i,j)}{r} + w_1 \cdot \left(\frac{A^2(i,j)}{r^2} - 1\right) \\ &= 1 + \frac{k \cdot b}{r \cdot cos\alpha} + w_1 \cdot \left(\frac{a^2 + (b \cdot \tan\alpha - c)^2}{r^2} - 1\right) \end{aligned} \tag{3.8}$$

As already pointed out, for each particular scenario of interest – where hole geometry, network topology, and the position of 'candidate node' (i.e. mobile node) are known – parameters **a**, **b** and **c** remain fixed. Hence, similar to $E(i,j|i = const) =$

$E(i, \alpha | i = const)$ from 3.3.2a, $C(i, j | i = const) = C(i, \alpha | i = const)$ ends up being a function of one single variable: **α**. Consequently, in order to find the value of **α**, i.e. the respective exit node, that minimizes $C(i, \alpha | i = const)$, we employ the standard first-derivative rule:

By minimizing the $C(i, \alpha | i = const)$ in (3.8) with respect to $\boldsymbol{\alpha}$ we get the following:

$$\frac{\partial C(i, \alpha | i = const)}{\partial \alpha} = \\ = \frac{k \cdot b \cdot \sin \alpha}{r \cdot \cos^2 \alpha} + \frac{2 \cdot w_1 \cdot b \cdot (b \cdot \tan \alpha - c) \cdot (1 + \tan^2 \alpha)}{r^2} = 0 \tag{3.9}$$

Note, we confirm that $C(i, \alpha | i = const)$ function has only one minimum and thus can be minimized by plotting (3.8) with **α** as a variable, while keeping the other parameters constant – see Figure 3.10.

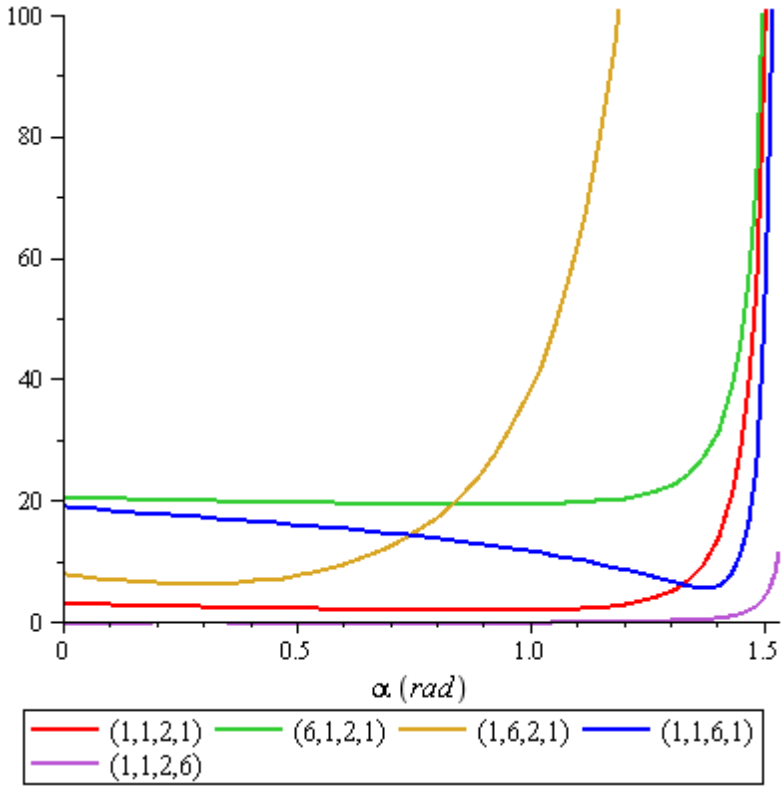


Figure 3.10 The function plots (w_1 = 0.5, k = *1.0*) produced from (3.8) all clearly have a single minimum when **α** is within the range of $[0...\frac{\pi}{2})$. The legend values in parenthesis correspond to energy cost function parameters (**a**, **b**, **c**, **r**).

Just as was the case with (3.3), it turns out that (3.9) is not solvable for **α** in a closed analytical form. Also, similar to what was observed in the earlier section (when we studied energy as the sole optimization criterion), it may again be noticed that expression (3.9) does not have **a** as a parameter. Hence, the optimal angle **α** – the one that minimizes the cost function $C(i, \alpha | i = const)$ – is independent of the width of the hole. Hence, the following lemma:

Lemma 3.4 The location of the exit node that optimizes (i.e. minimizes) the combined energy-delay cost function does not depend on the width of the observed rectangular hole.

Next, in order to examine how the values of w_1 affect the optimal exit node, we first solve (3.9) numerically using Maple [47], and then plot the obtain solutions. The generated plots imply a few interesting conclusions.

In Figure 3.11, the position of the optimal exit node $nb_{exit}\,(i,j \mid i = const)$ is plotted as a function of **b**, **c** and $\boldsymbol{w_1}$. For example, for $w_1 = 0.5$, the plot is generated using the Maple following command:

$$plot3d\left(\frac{1}{c}(b\cdot\tan(solve(\ (diff\,(edp\,(alpha,\ 50,\ b,\ c,\ 2,\ 0.5\,),\mathrm{alpha})\ = 0),\mathrm{alpha}\ ))),\ b = 2\,..10,\ c = 2\,..10\ \right);$$

where *edp* is,

$$edp\ :=\ (\alpha, a, b, c, r, w1\,) \rightarrow \frac{b}{r\,\cos(\alpha)} + w1\left(\frac{a^2 + (b\,\tan(\alpha) - c)^2}{r^2} - 1\right)$$

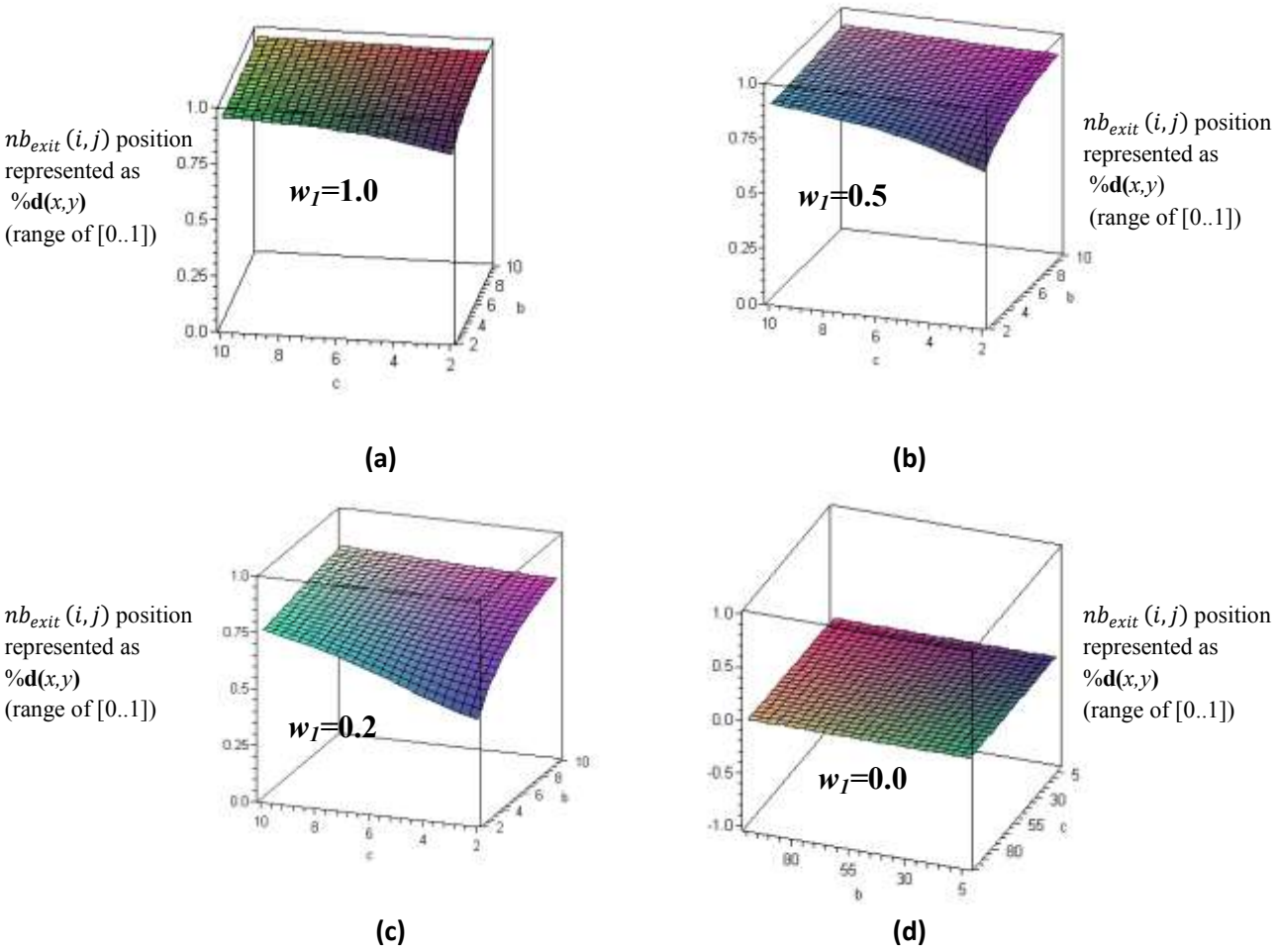


Figure 3.11 As energy becomes less significant and delay more significant in the overall cost function through the decrease of w_1, α approaches 0, i.e. the location of the optimal exit node approaches point ***y*** in Figure 3.5.

By examining Figure 3.11 (a to d) it becomes apparent that as w_1 decreases, the optimal value of angle α approaches 0. Accordingly, the position of the optimal exit node approaches point ***y*** in Figure 3.5. The opposite can be observed for high values of w_1, when energy consumption becomes a more dominant component of

$C(i, \alpha | i = const)$ and the position of optimal exit node approaches point ***x*** in Figure 3.5.

3.3.3. Determining Optimal 'Candidate Node'

In Section 3.3.2, under the assumption of a known/fixed position of the mobile bridge in the interior of a rectangular-shaped routing hole (next to a particular candidate node $nb(i)$), we have resolved the problem of identifying the respective optimal exit node[19] – one that corresponds to the (local) minimum of the energy and/or delay cost function. In this section, in an attempt to the optimal deployment location of the mobile which would match the global minimum of the energy and/or delay cost function, we will relax the earlier assumption. Specifically, we will look at the entire space of possible candidate nodes (i.e. respective mobile node locations), each in combination with its corresponding optimal exit node (as discussed in Section 3.3.2a), and identify the one that truly minimizes the cost function.

In order to find such 'globally optimal' candidate node, we propose a bidding procedure similar to the one from [45]. According to this procedure, all candidate nodes calculate their so-called bidding values[20], which are subsequently compared against each other by means of distributed cooperative computation. The boundary node that calculates (i.e.

[19] Recall, previously we have referred to this problem as one of the two 'sub-problems' of OPlaMoN-2 algorithm.

[20] Recall, bidding value of candidate node $nb(i)$ is an indirect representation of the overall energy/delay cost associated with the deployment of the mobile bridge next to this particular candidate node.

corresponds) to the lowest bid is considered to be the best candidate for the mobile node deployment.

Please recall, in contrast to the original algorithm proposed in [45], which dealt with simple *1-hole/1-mobile/1-traffic-streams* scenarios, here we consider the possibility that a single routing hole impedes multiple traffic streams of different intensities (see Figure 3.12). Hence, in order to calculate its bidding value $Bid\big((s_1, s_2, \dots, s_k \mid mob(i)\big)$, every boundary node $nb(i)$ $(i = 1, \dots, n)$ has to perform the following:

1) receive the information about all traffic streams impacting the hole;
2) for each of the traffic streams determine which of the two possible routes is less costly to take: (a) around the hole, or (b) through the mobile bridge, if the mobile is to be placed next itself ($nb(i)$) – see expression (3.10);
3) sum up the smaller (of the two considered) routing costs over all traffic streams – see expression (3.11).

Note, in order to make our discussion and calculations more general, we consider the 'cost' of routing to be a weighted combination of delay and energy (as already explained in Section 3.3.2).

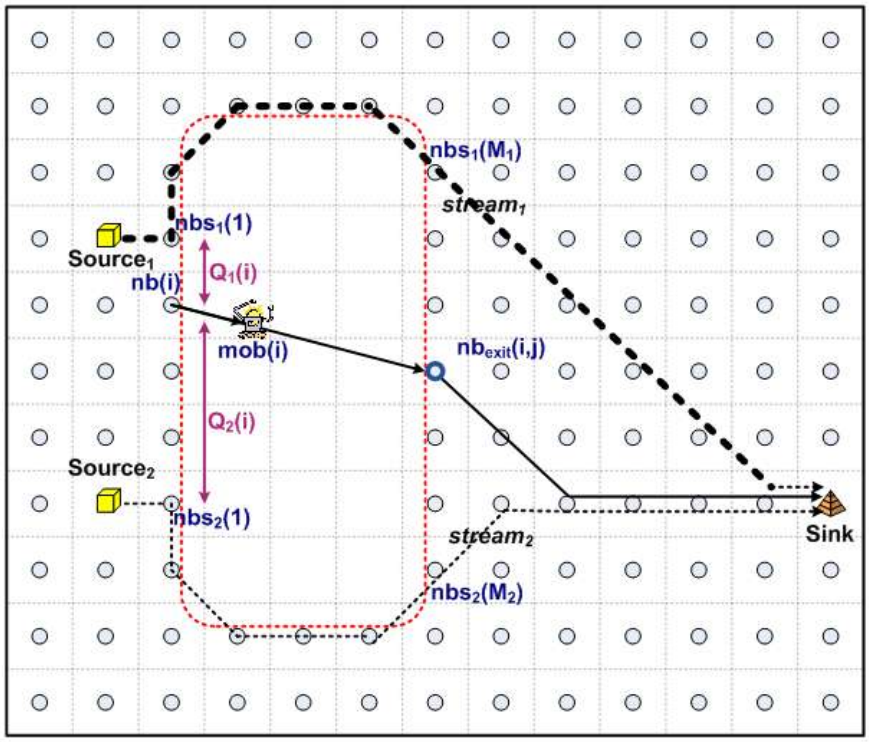


Figure 3.12 Two traffic streams of different data rates impeded by same routing hole. For each stream, two options exist: to be routed around the hole's boundary or to be routed through the mobile placed next to boundary node $\boldsymbol{nb(i)}$ $(\boldsymbol{i = 1, \dots, n})$. To reach the mobile when placed next to particular $\boldsymbol{nb(i)}$, traffic streams may have to be diverged from their 'natural' routing direction.

$$BestEDP(s_k \mid mob(i)) = \min\{EDP_{around-hole}(s_k), EDP_{through-mob}(s_k \mid mob(i))\} \tag{3.10}$$

$$Bid(s_1, s_2, \dots, s_k \mid mob(i)) = \sum_{k=1}^{K} Rate(s_k) \cdot BestEDP(s_k \mid mob(i)) \tag{3.11}$$

A more detailed overview of the four functions appearing (3.10) and (3.11) and defining $Bid(s_1, s_2, \dots, s_k \mid mob(i))(i = 1, \dots, n)$, together with their physical interpretation, are provided below.

$Rate(s_k)$:	Data rate (i.e. intensity) of stream s_k, as measured by inflow node $nbs_k(1)$;
$EDP_{around\ -hole}(s_k)$:	Energy-delay performance, i.e. a weighted sum of transmission delay and consumed energy associated with the routing of a single data bit belonging to stream s_k, if s_k is to be routed around the hole.
$EDP_{through-mob}\left(s_k \middle\| mob(i)\right)$:	Energy-delay performance, i.e. a weighted sum of transmission delay and consumed energy associated with the routing of a single data bit belonging to stream s_k, if s_k is to be routed across the hole via a mobile bridge placed next to boundary node $nb(i)$.
$BestEDP(s_k \| mob(i))$:	Smaller of the above two energy-delay costs (i.e. performances) associated with the routing of stream s_k, if one particular position of the mobile bridge is considered – next to boundary node $nb(i)$:

Based on the above, it is clear that $Bid\left(s_1, s_2, \ldots, s_k \mid mob(i)\right)$, as defined in (3.11), represents the smallest possible energy/delay cost associated with the deployment of the mobile bridge next to boundary node $nb(i)$ – cumulative over all affected traffic streams, in proportion to their actual intensity. It is worth noting that the smallest possible energy/delay cost associated with the deployment of the mobile bridge next to $nb(i)$ allows for the possibility that some streams still get to be routed around the hole, if that turns out to be more cost effective.

In the following subsections, we take a more detailed look at how each of the following three functions is defined and could be calculated: $EDP_{around\ -hole}(s_k)$,

$EDP_{through-mob}(s_k|mob(i))$, and $BestEDP_{through-mob}(s_k|mob(i))$, ultimately enabling the calculation of $Bid(s_1, s_2, \ldots, s_k \mid mob(i))$ by each boundary node.

(a) Calculation of $EDP_{around-hole}(s_k)$

According to its definition, $EDP_{around-hole}(s_k)$ represents a weighted sum of transmission delay and consumed energy associated with the routing of a single data bit belonging to stream s_k, if s_k is to be routed around the hole. The two components of $EDP_{around-hole}(s_k)$ are annotated with $E_{around-hole}(s_k)$ and $D_{around-hole}(s_k)$, as shown in (3.12).

$$EDP_{around-hole}(s_k) = w_1 \cdot E_{around-hole}(s_k) + w_2 \cdot D_{around-hole}(s_k) \quad (3.12)$$

Based on the annotation and discussion from the earlier sections, one can easily derive (i.e. validate) the following expressions for $E_{around-hole}(s_k)$ and $D_{around-hole}(s_k)$,

$$\begin{aligned} E_{around-hole}(s_k) &= E_{per-hop} \cdot NH_{around-hole}(s_k) \\ &= P_{received} \cdot r^2 \cdot \left[(M_k - 1) + k \cdot \frac{d_{Euclid}(nbs_k(M_k), Sink)}{r}\right] \end{aligned} \quad (3.13)$$

$$\begin{aligned} D_{around-hole}(s_k) &= D_{per-hop} \cdot NH_{around-hole}(s_k) \\ &= D_{per-hop} \cdot \left[(M_k - 1) + k \cdot \frac{d_{Euclid}(nbs_k(M_k), Sink)}{r}\right] \end{aligned} \quad (3.14)$$

where:

- $E_{per-hop}$ represents the average energy required to transmit a single bit of data over a single hop of the given network (in units of energy);
- $D_{per-hop}$ represents the average delay experienced by a single bit of data when transmitted over a single hop of the given network (in units of time);
- $NH_{around-hole}(s_k)$ represents the number of hops that a bit of stream s_k traverses from the inflow node $\left(nbs_k(1)\right)$ to the sink, when the mobile node is not used.

Also, please recall, M_k corresponds to the total number of boundary nodes involved in the routing of stream s_k, $nbs_k(M_k)$ represents the last boundary node 'touched by' stream s_k, while d_{Euclid} denotes Euclidean distance between two points in a 2-dimensional space.

A closer inspection of (3.13) and (3.14) reveals that $E_{around-hole}(s_k)$ and $D_{around-hole}(s_k)$, can be represented with the same base function $NH_{around-hole}(s_k)$, but in each case multiplied with a different constant: $E_{per-hop}$ and $D_{per-hop}$, respectively. Given that these two constants are of inherently different unit and scale, and have no significance from the optimization point of view, we remove them from the further analysis, and simply rewrite (3.12) as:

$$EDP_{around-hole}(s_k) = (w_1 + w_2) \cdot NH_{around-hole}(s_k) = \\ = (w_1 + w_2) \cdot \left[(M_k - 1) + k \cdot \frac{d_{Euclid}\left(nbs_k(M_k), Sink\right)}{r}\right] \quad (3.15)$$

Finally, by recalling that $w_1 + w_2 = 1$, we arrive at the following expression for $EDP_{around-hole}(s_k)$.

$$EDP_{around-hole}(s_k) = NH_{around-hole}(s_k) = \\ = (M_k - 1) + k \cdot \frac{d_{Euclid}(nbs_k(M_k), Sink)}{r} \quad (3.16)$$

(b) Calculation of $EDP_{through-mob}(s_k|mob(i))$

According to its definition, $EDP_{through-mob}(s_k|mob(i))$ represents a weighted sum of transmission delay and consumed energy associated with the routing of a single data bit belonging to stream s_k, if s_k is to be routed across the hole via a mobile bridge placed next to boundary node $nb(i)$ – see (3.17).

$$EDP_{through-mob}(s_k \mid mob(i)) = \\ = w_1 \cdot E_{through-mob}(s_k \mid mob(i)) + w_2 \cdot D_{through-mob}(s_k \mid mob(i)) \quad (3.17)$$

We annotate the energy and delay component of $EDP_{through-mob}(s_k \mid mob(i))$ with $E_{through-mob}(s_k \mid mob(i))$ and $D_{through-mob}(s_k \mid mob(i))$, and provide the actual expression for each in (3.18) and (3.19), respectively.

$$E_{through-mob}(s_k \mid mob(i)) = \\ = P_{received} \cdot r^2 \cdot (NH_{through-mob}(s_k|mob(i)) - 1) + \\ + P_{received} \cdot d^2_{Euclid}(mob(i), nb_{exit}(i)) = \\ = E_{per-hop} \cdot (NH_{through-mob}(s_k|mob(i)) - 1) + \\ + P_{received} \cdot d^2_{Euclid}(mob(i), nb_{exit}(i)) = \quad (3.18)$$

$$= E_{per-hop} \cdot \left(Q_k(i) + 1 + k \cdot \frac{d_{Euclid}\,(nb_{exit}\,(i), Sink)}{r} \right) + P_{received} \cdot d^2_{Euclid}\left(mob(i), nb_{exit}\,(i)\right) =$$

$$= E_{per-hop} \cdot \left[Q_k(i) + 1 + k \cdot \frac{d_{Euclid}\,(nb_{exit}\,(i), Sink)}{r} + \frac{d^2{}_{Euclid}\left(mob(i), nb_{exit}(i)\right)}{r^2} \right]$$

$$D_{throug\,h-mob}\left(s_k \mid mob(i)\right) = D_{per-hop} \cdot NH_{throug\,h-mob}\left(s_k \mid mob(i)\right) = = D_{per-hop} \cdot \left(Q_k(i) + 2 + k \cdot \frac{d_{Euclid}\,(nb_{exit}\,(i), Sink)}{r} \right) \quad (3.19)$$

In the above expressions, the following notation is used:

- $NH_{throug\,h-mob}\left(s_k \mid mob(i)\right)$ represents the number of hops that a bit of stream s_k traverses from the inflow node $\left(nbs_k(1)\right)$ to the sink, when routed through the mobile bridge placed in the vicinity of boundary node $nb(i)$;
- $Q_k(i)$ represents the hop distance between the inflow node of stream s_k $\left(nbs_k(1)\right)$ and boundary node $nb(i)$ currently 'hosting' the mobile bridge – please see Figure 3.12.

Again, by ignoring the two constants – $D_{per-hop}$ and $E_{per-hop}$ – and then substituting (3.18) and (3.19) into (3.17), we obtain the following final expression for $EDP_{throug\,h-mob}\left(s_k \mid mob(i)\right)$:

$$EDP_{throug\,h-mob}\left(s_k \mid mob(i)\right) = \quad (3.20)$$

$$= w_1 \cdot \left[Q_k(i) + 1 + k \cdot \frac{d_{Euclid}\left(nb_{exit}(i), Sink\right)}{r} + \frac{d^2{}_{Euclid}\left(mob(i), nb_{exit}(i)\right)}{r^2} \right] +$$

$$+ w_2 \cdot \left(Q_k(i) + 2 + k \cdot \frac{d_{Euclid}\left(nb_{exit}(i), Sink\right)}{r} \right) =$$

$$= Q_k(i) + 2 + k \cdot \frac{d_{Euclid}\left(nb_{exit}(i), Sink\right)}{r} + + w_1 \cdot \left[\frac{d^2{}_{Euclid}\left(mob(i), nb_{exit}(i)\right)}{r^2} - 1 \right]$$

(c) Calculation of $BestEDP\left(s_k \mid mob(i)\right)$

By substituting (3.16) and (3.20) back in (3.10), we obtain the following expanded expression for $BestEDP\left(s_k \mid mob(i)\right)$ $(k = 1, \dots, K)$, as should be used by boundary nodes to calculate their bidding values (see (3.11)):

$$BestEDP\left(s_k \mid mob(i)\right) = \min \begin{cases} (M_k - 1) + k \cdot \dfrac{d_{Euclid}\left(nbs_k(M_k), Sink\right)}{r} \\ Q_k(i) + 2 + w_1 \cdot \left[\dfrac{d^2_{Euclid}\left(mob(i), nb_{exit}(i)\right)}{r^2} - 1 \right] \\ + k \cdot \dfrac{d_{Euclid}\left(nb_{exit}(i), Sink\right)}{r} \end{cases} \quad (3.21)$$

A quick inspection of (3.21) reveals that in a location-aware WSN, the calculation of $BestEDP\left(s_k \mid mob(i)\right)$ – at each particular boundary node $nb(i)$ and for each

particular traffic stream s_k – is relatively inexpensive, as it requires that just the following be known:

- the physical locations of $Sink$, $nbs_k(M_k)$, $nb_{exit}(i)$ and $mob(i)$ (if deployed in the vicinity of $nb(i)$);
- M_k – overall number of boundary nodes affected by stream s_k;
- $Q_k(i)$ – number of hops between the inflow node of stream s_k $\left(nbs_k(1)\right)$ and the node $nb(i)$ itself.

Out of the above parameters:

- In most application scenarios, the location of the $Sink$ already is (i.e. must be) known by all nodes in the network.
- The locations of $mob(i)$ and $nb_{exit}(i)$ can be relatively quickly estimated by boundary node $nb(i)$, especially if one single cost function is deployed – either just delay or just energy (see Section 3.3.2).
- The location of $nbs_k(M_k)$ and the value of M_k have the same value at/for each boundary node $nb(i)$, as they only depend on the original (around the hole) path of the observed stream s_k. Hence, the information regarding these two parameters could be disseminated to all boundary nodes in a single 'around the hole' round of message exchange.
- The value of $Q_k(i)$ is the only parameter that depends on both – observed boundary node $nb(i)$ and observed traffic stream s_k. Still, this parameter could also be

estimated in a single 'around the hole' round of message exchange, in which a packet with a gradually increasing counter is generated by the inflow node $nbs_k(1)$ and passed from one node to another.

Based on the above, the overall communication overhead associated with the calculation of bidding values for all participating boundary nodes is of the order of '$O(K)$ *rounds of around the hole message exchange*', where K represents the overall number of traffic streams impeded by the observed routing hole.

3.3.4. OPlaMoN-2

As stated at the beginning of this section (Section 3.3), the goal of our work has been to develop OPlaMoN-2 – an improved version of the original algorithm for determining the optimal deployment location of a mobile bridge in the interior of a routing hole. Two major improvements over the original algorithm include:

(a) optimized selection of 'exit node' (as discussed in Section 3.3.2);

(b) improved bidding function and bidding procedure (as discussed in Section 3.3.3).

We end Section 3.3 with a more detailed algorithmic representation of OPlaMoN-2 – provided in

Figure 3.13, and a few general concluding remarks regarding the algorithm. For a descriptive outline of OPlaMoN-2 please refer back to 3.3.

(1) **Routing Hole Identification Phase:**

Use TENT and BOUNDHOLE to discover routing hole boundary nodes [24].

(2) **Traffic Stream Estimation and Dissemination Phase:**

(2.a) During a period T_1

if $nb(i) = nbs_k(1)$

- measure $Rate(s_k)$;

else

- wait for T_1 to elapse.

(2.b) If $nb(i) = nbs_k(1)$

- send $[Rate(s_k), NodeId(nbs_k(1)), count = 0]$ to next boundary clockwise neighbor $nb(i+1)$, with itself as the final destination;

(2.c) During a period T_2

- wait for information packets from (other) inflow nodes;
- upon receiving an inflow-node information packet, locally store $Rate(s_k)$, $NodeId(nbs_k(1))$, and $count$++;
- forward any inflow-node information packets to next clockwise neighbour $nb(i+1)$, unless the packet has originated at itself.

(3) **Bidding Value Calculation Phase**

(3.a) Calculate $nb_{exit}(i,j)$ according to conclusions of Section 3.2.2.

(3.b) Calculate node $nb(i)$ bidding value according to equation (3.10) .

Algorithm continuous on the next page.

(4) Bidding Procedure

(4.a) If $nb(i) = nb(1)$*

- $highestBid = Bid(nb(1))$;
- $higestBidderId = NodeId(nb(1))$;
- send $[highestBid, highestBidderId]$ to node $nb(i+1)$;

else

- wait for $[highestBid, highestBidderId]$ from node $nb(i-1)$;
- calculate:
  $highestBid = max\{highestBid, Bid(nb(i))\}$;
  $highestBidderId = NodeId\{highestBid\}$
- forward $[highestBid, highestBidderId]$ to node $nb(i+1)$.

(4.b) If $nb(i) = nb(1)$ (after receiving $[highestBid, highestBidderId]$ from $nb(i-1)$), set the winning-bid parameters:

- $winningBid = highestBid;$
- $winningBidderId = highestBidderId;$
- send $[winningBid, winningBidderId]$ to node $nb(i+1)$;

else

- wait for $winningBid$ and $winningBidderId$ from node $nb(i-1)$;
- after receiving and storing $winningBid$ and $winningBidderId$, forward $[winningBid, winningBidderId]$ to node $nb(i+1)$

(5) Routing Activation Phase

If $NodeId(nb(i)) = winningBidderId$

- Calculate optimal mobile node deployment location
- Broadcast $[mobileNodeDeoplymentRequest]$

else if $NodeId(nb(i)) = NodeId(nbs_k(1))$

- Wait for mobile node reply

* $nb(1)$ can be any arbitrary node on the hole's boundary selected according to the criteria such that if $n =$ number of candidate nodes, and if a data unit was to traverse every $nb(i)$ in order, starting with $nb(1)$, then $nb(n)$ would be the last $nb(i)$ node reached.

Figure 3.13 Overview of OPlaMoN-2

As can be seen from Figure 3.13, OPlaMoN-2 (similar to the original algorithm) attempts to determine the optimal deployment location of the mobile bridge through distributed and cooperative computation involving a minimum number of sensor nodes, i.e. involving only the nodes that happen to lie on the boundary of the observed routing hole. (Note: OPlaMoN-2 is 'distributed' since it assumes a collective effort of multiple 'agents' to solve a task at hand through combining of the agents' information and capabilities[49]. OPlaMoN-2 is also cooperative, given that each node uses incomplete input data while simultaneously exchanging the intermediate results of its processing with other nodes to cooperative construct a complete solution [50].) Some known advantages of distributed (over centralized) and cooperative (over non-cooperative) problem solving include: increased reliability and fault tolerance, lower communication and processing cost per agent/node, reduced software complexity, etc. All of these properties make OPlaMoN-2 is highly suited for use in energy and processing constrained large-scale wireless sensor network.

Chapter 4

Simulations

In this chapter we present some of our selected simulation results. These results are generally aimed at:

1) providing an insight into the performance of two major components of the OPlaMoN-2 algorithm: *optimal exit node selection* and *bidding value calculation*; and

2) verifying some of our (other) key theoretical findings and contributions – as discussed in Chapter 3.

4.1. Simulation Framework

The simulations were performed using the Qualnet [21] network simulator and a simulation framework borrowed from Moniz [51]. The framework was modified in order to fit the purpose of our experiments, but much of the underlying structure remained unchanged and included the following four phases:

1. **Neighbor Discovery** – Nodes discover the locations of their topological neighbors.
2. **Hole Discovery** – Nodes use the TENT [40] rule in order to identify whether they are stuck (i.e. cannot use greedy routing) relative to the destination. Next, each stuck node generates a special control packet, which is sent around the boundary of the routing hole in order to identify other routing hole boundary nodes and report them back to the packet's origin.
3. **Hole Boundary Information Dissemination** – After the first control packet from the hole discovery phase returns to its originating node, the node generates a second control packet containing the final list of all boundary nodes, which is then sent around the hole.
4. **Sensing and Routing** – In this phase source nodes begin generating data packets containing information obtained by the nodes' sensors. The data packets are routed through the network, and towards the destination, using geographic routing. When

[21] Qualnet is a network modeling tool developed by Scalable Network Technologies [68].

a packet encounters a routing hole, it can no longer be forwarded using the greedy approach, so the network switches to the so-called Mobile Assisted Routing Strategy (MARS). According to this strategy, packets are forwarded either: 1) around the boundary of the hole, to the outflow node of the respective stream, after which point the regular greedy routing commences, or 2) through the mobile bridge and its respective exit node, after which point the regular greedy routing again commences. In our simulations, MARS is modified so to accommodate the simulation of two central OPlaMoN-2 components: *optimal exit node selection* and *bidding value calculation*, as discussed in the earlier chapter.

For more details on the simulation framework, please refer to [51].

4.2. General Simulation Setup

Each of the simulation experiments was performed on a network area that was divided into a grid, with each grid cell occupying an area of 50x50 units. Every cell contained a randomly placed node, with exception of the cells which fell into the area occupied by the routing hole. Each node was able to communicate with (up to) 8 neighbors located in cells adjacent to its own. Such network configuration is generally known as the *virtual network grid topology* and is useful for simplifying formal analysis [46].

4.3. Simulation Experimentation 1: Optimal Exit Node (OEN) Selection

The main purpose of this set of simulation-based experiments was to determine whether the theoretical 'optimal exit node' selection criteria (as discussed in Chapter 3), based on the combined energy-delay metric, would be in line with the actual energy/delay conditions as found in real-world WSNs.

4.3.1. OEN Simulation Setup and Execution

The **optimal exit node** (OEN) simulation involved deploying nodes in a virtual grid of 28x28 cells with the routing hole occupying 21x5 cells, as shown in Figure 4.1. The red triangle in Figure 4.1 represents the assumed location of the mobile bridge responsible for forwarding packets across the routing hole towards one of nine possible destinations - marked with letters **A** through **P**. In each simulation run, a packet was forwarded to one of the given destinations using one of seven possible exit nodes – labeled with **E1** through **E7**. As each packet traversed from the source to the destination, the number of hops made and the Euclidean distance between mobile node and the exit node, was recorded in the packet. Once at the destination, this data was used to calculate the cost based on the combined delay/energy metric.

The simulations were performed for four different weights of w_1, as appearing in the combined energy-delay cost function (3.6).

- $w_1 = 1.00$ – only energy cost was considered;
- $w_1 = 0.20$ – both cost metrics, energy and delay, were combined;
- $w_1 = 0.02$ – major emphasis was put on delay cost;
- $w_1 = 0.00$ – only delay cost was considered.

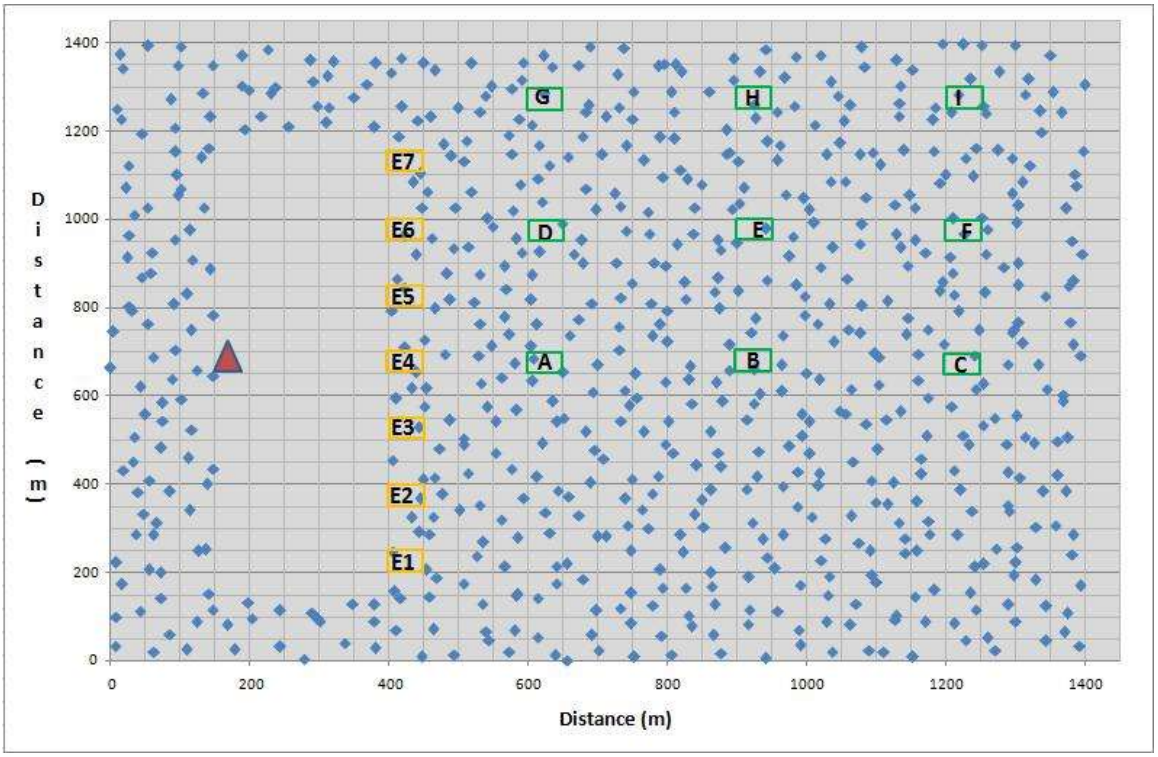


Figure 4.1 OEN simulation layout

4.3.2. OEN Simulation Results

(a) OEN selection using energy (only) cost criteria: $w_1 = 1.00$

The simulation results for $w_1 = 1.00$ – only energy was used as the criteria for OEN selection – can be seen in Figure 4.2 and Figure 4.3.

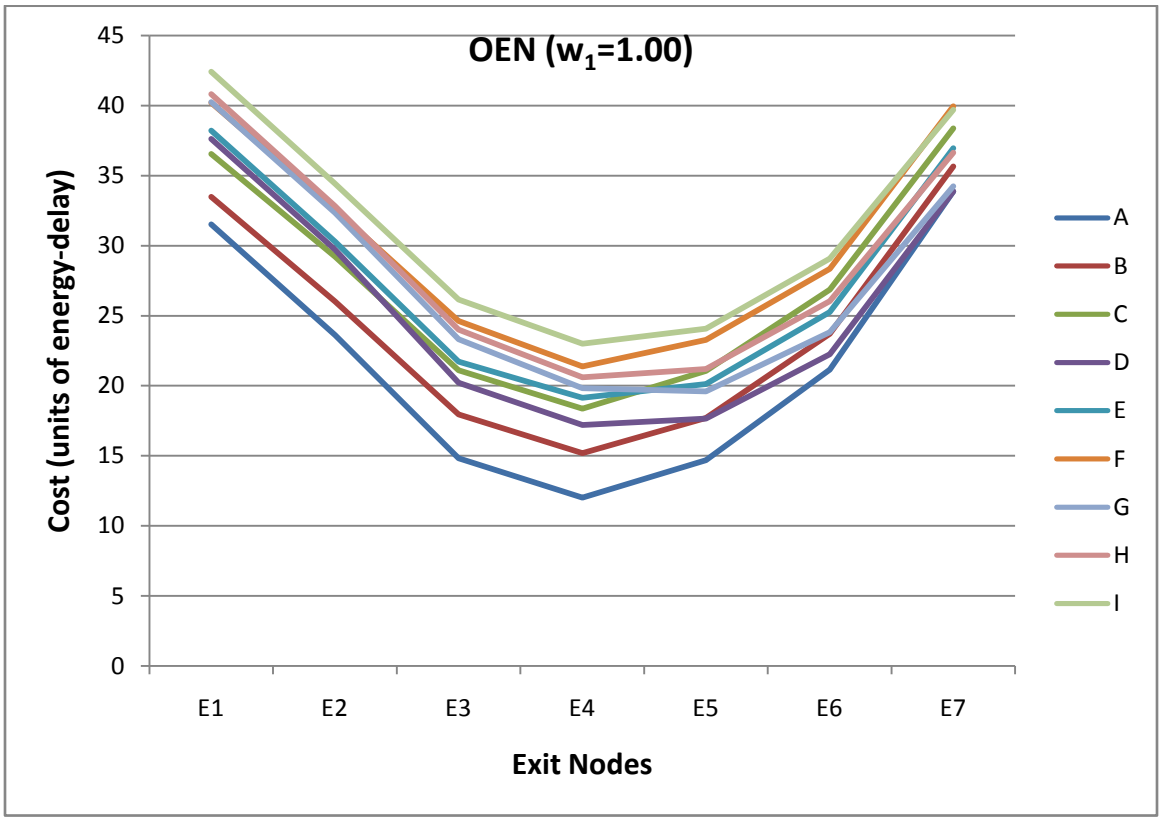


Figure 4.2 OEN simulation results: $w_1 = 1.00$

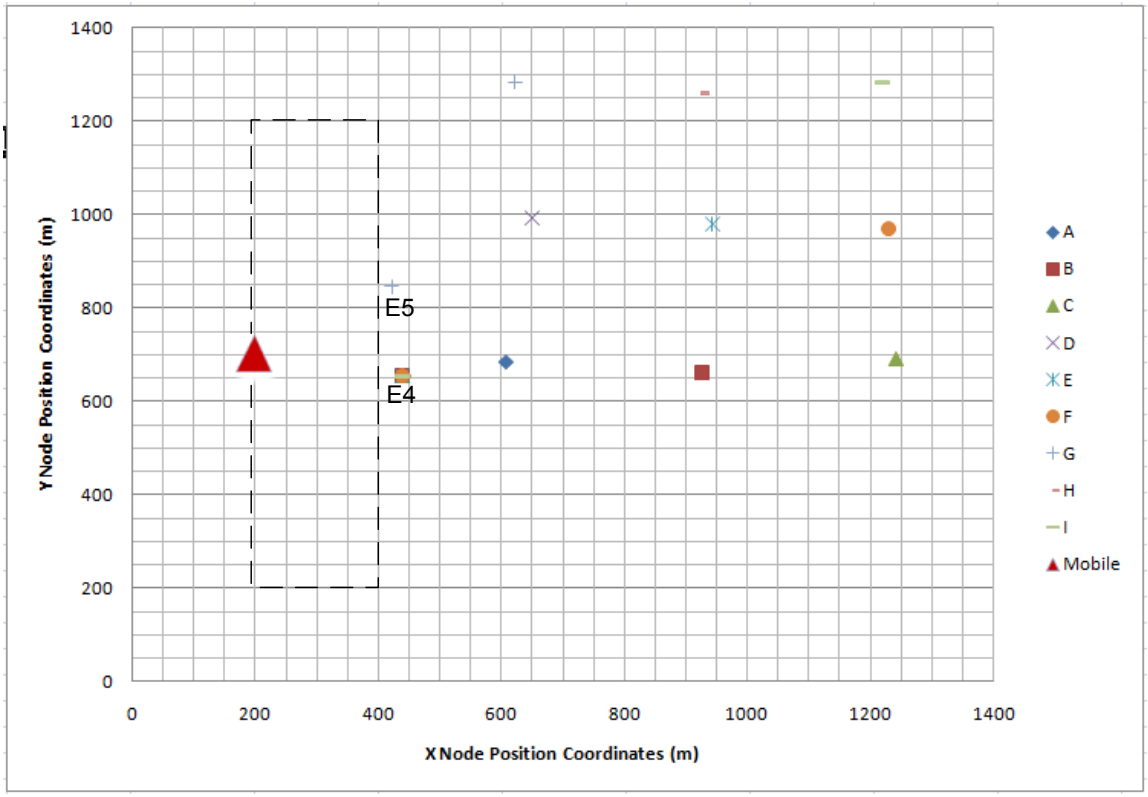


Figure 4.3 OEN simulation ($w_1 = 1.00$) results: min-cost exit nodes (OENs) with respect to their destinations.[22]

From Figure 4.2 and Figure 4.3 it can be observed that all considered destination (**A** through **I**) except **G,** the exit node with the smallest energy cost was always near **E4**, which is inline with our conclusions from Chapter 3 (Lemma 3.3). To elaborate, when energy is used as the main performance criterion, the optimal (minimum energy) exit node is the one that happens to be at the minimum physical distance from the mobile, regardless of the actual location of the sink, i.e. its distance from the observed routing hole.

[22] Since the exit nodes' positions are discrete in the simulations, the symbols of the OENs may overlap in the figure, and hence obstruct one another from view.

Figure 4.4 shows the theoretical positions of OENs – obtained through the procedure described in Chapter 3, Section 3.3.2 c) – relative to their corresponding destinations. The theoretical position of the optimal exit nodes can be seen positioned closely together along the *y*-axis (at 400 meters along the *x*-axis), with an approximate distribution range of 675 to 720 meters along the *y*-axis. To explain, all optimal exit nodes are positioned closely together so to (nearly) minimize the distance between themselves and the mobile node.

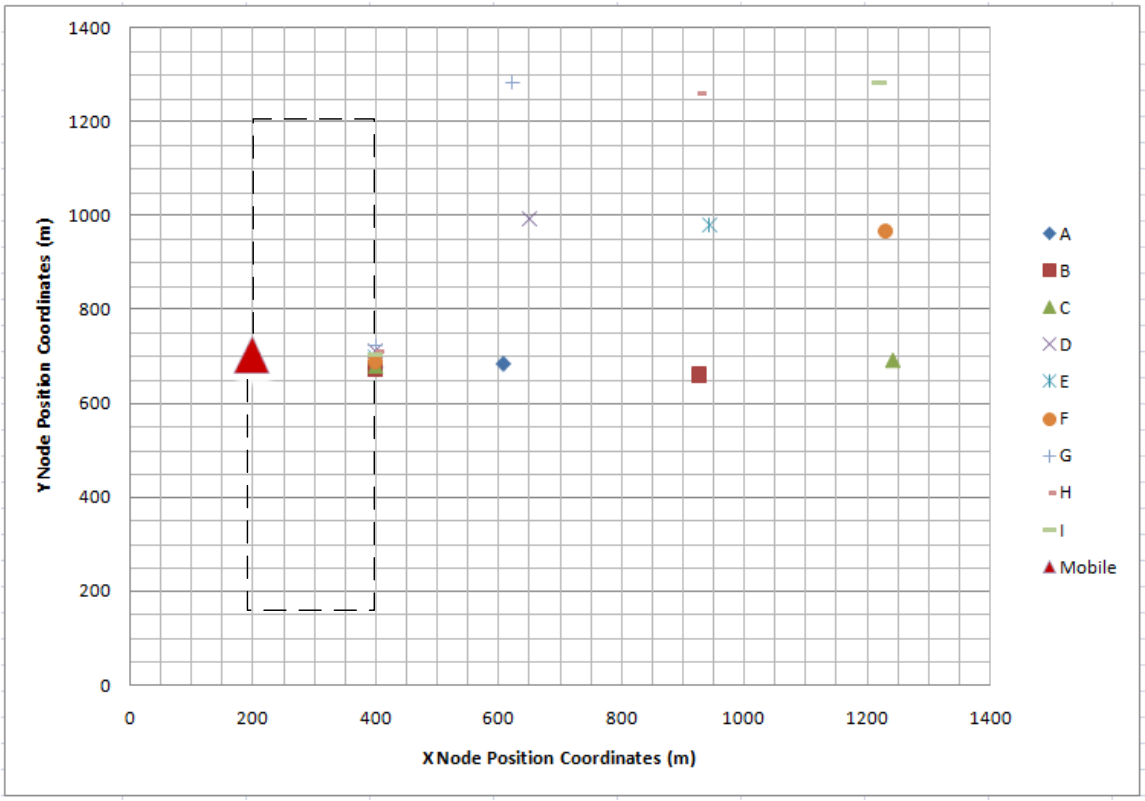


Figure 4.4 Theoretical positions of OENs relative to their corresponding destinations ($w_1 = 1.00$). The dashed rectangle indicates the boundaries of the routing hole.

As previously mentioned, the simulation results (Figure 4.2 and Figure 4.3), have shown that the optimal exit node is **E4** (i.e. the exit node with the minimal distance to the mobile node). Exit node **E4** is found at 654 along the *y*-axis, and is closest from among the simulated exit nodes to the positions of the theoretically calculated optimal exit nodes that are found in the range of 675 to 720 meters along the *y*-axis[23]. Therefore, it is reasonable to conclude that our simulations provide further support for our theoretical analysis.

[23] The failure of **E4** to fall directly into the position range of the theoretical optimal exit nodes is attributed to the randomness in distribution of the discrete exit nodes.

(b) OEN selection using combined energy-delay criterion: $w_1 = 0.20$

The simulation results for w_1 = 0.20 – delay becomes a factor in OEN selection – can be seen in Figure 4.5 and Figure 4.6. The results show that as delay begins to affect the combined (energy-delay) cost function, **E4** is no longer the optimal exit node for all cases. For example, **E5** is now the OEN for destination **D**, while **E6** is the OEN for destination **G**.

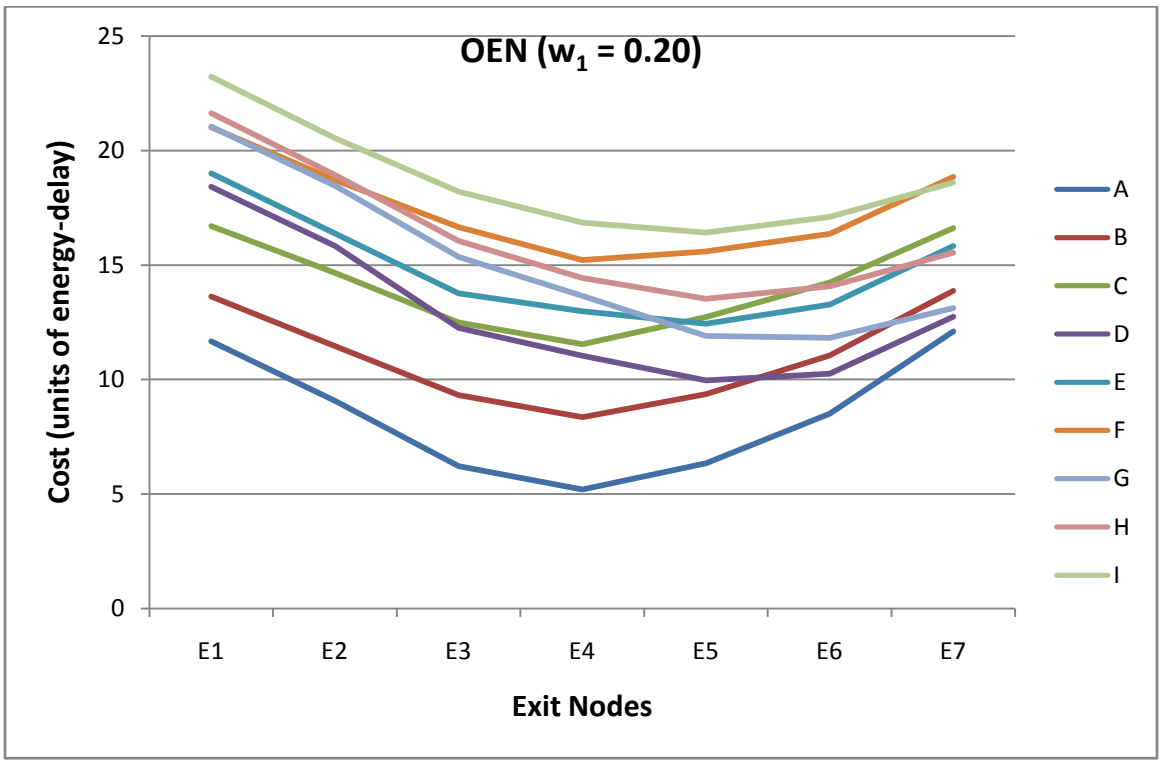


Figure 4.5 OEN simulation results: $w_1 = 0.20$

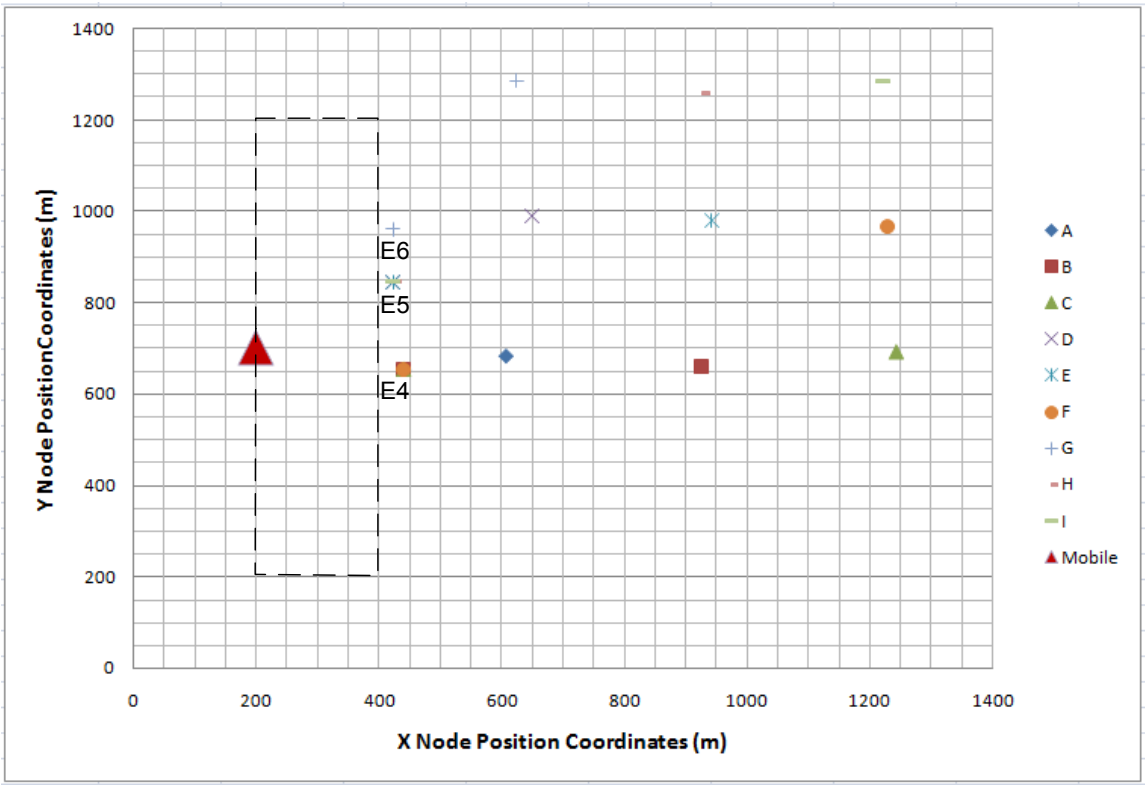


Figure 4.6 OEN simulation ($w_1 = 0.20$) results: min-cost exit nodes (OENs) with respect to their destinations.

The above experimental findings appear to be in line with the theoretical results derived using the procedure described in Chapter 3, Section 3.3.2 c) and are presented in Figure 4.7. For instance, according to Figure 4.7, the (theoretically) optimal exit node for destination **D** is located around low 800 [m] along the *y*-axis. This is consistent with the simulation results from Figure 4.5 in which **E5**, located at/near the same coordinate(s) (see Figure 4.1), is the OEN for the same destination – **D**. The other OEN locations, obtained theoretically and through simulation for $w_1 = 0.2$, also appear consistent with each.

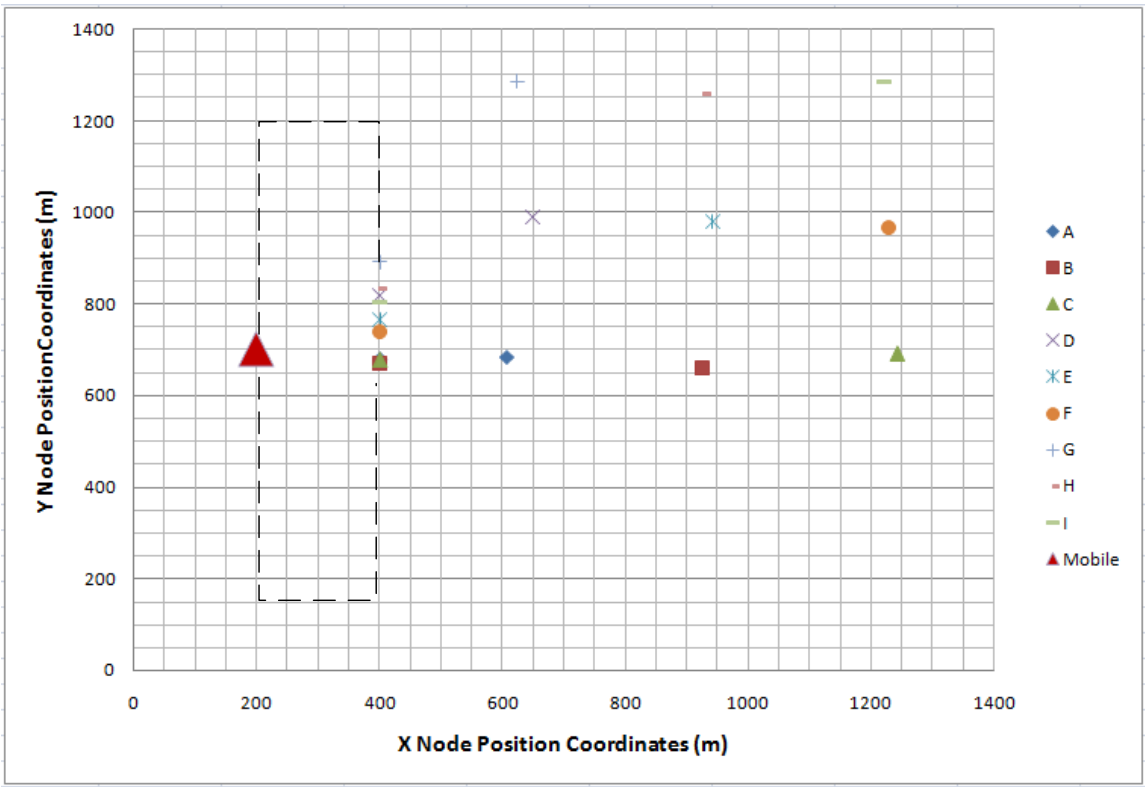


Figure 4.7 Theoretical positions of OENs relative to their matching destinations ($w_1 = 0.20$).

(c) OEN selection using combined energy-delay criteria: $w_1 = 0.02$

The simulation results for $w_1 = 0.02$ – delay becomes a much more significant factor in OEN selection – can be seen in Figure 4.8 and Figure 4.9.

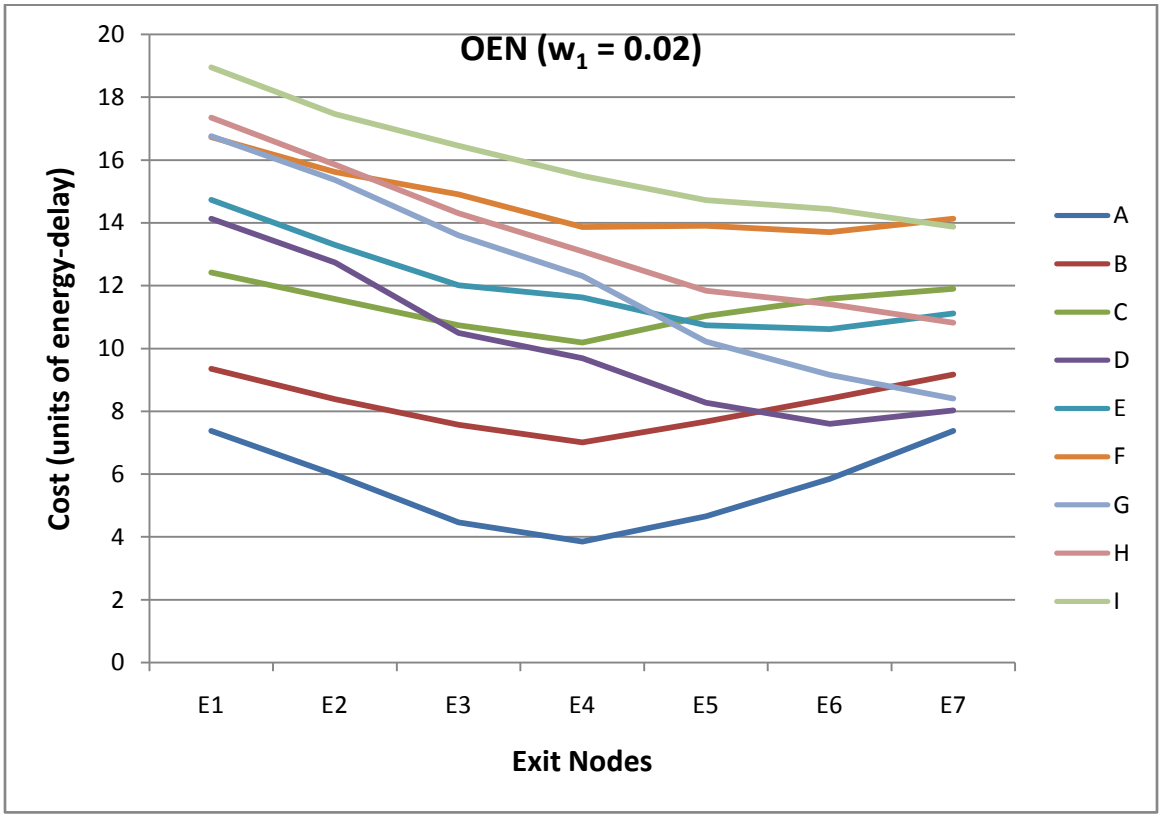


Figure 4.8 OEN simulation results: $w_1 = 0.02$

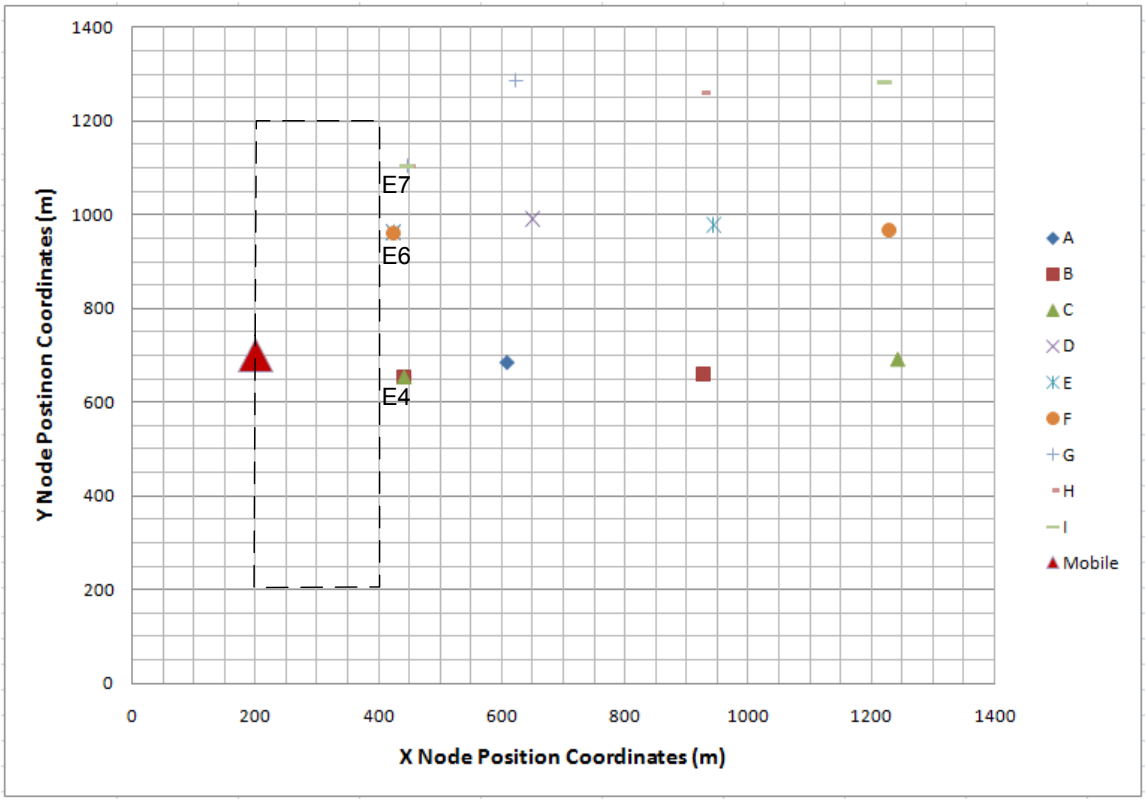


Figure 4.9 OEN simulation ($w_1 = 0.02$) results: min-cost exit nodes (OENs) with respect to their destinations.

The above results show that as delay gains a more dominant role in the combined cost function, it becomes advantageous to send data to exit nodes that are physically very close to the destinations (i.e. sink). (These results are fully in line with our earlier general findings from Chapter 3 – see

Lemma 3.1.) For example, when forwarding data to destinations **I**, **H** or **G**, exit node **E7** emerges as 'optimal' (i.e. one that minimizes the cost), even though it is farthest from the mobile node. At the same time, when forwarding data to destinations **A**, **B** and **C,**

exit node **E4** is used as it minimizes the cost function by significantly reducing delay, though inadvertently energy cost is minimized as well.

The respective theoretically obtained optimal exit nodes are presented in Figure 4.10. Clearly, there is a high degree of correlation between the theoretical and the above (simulation-based) results.

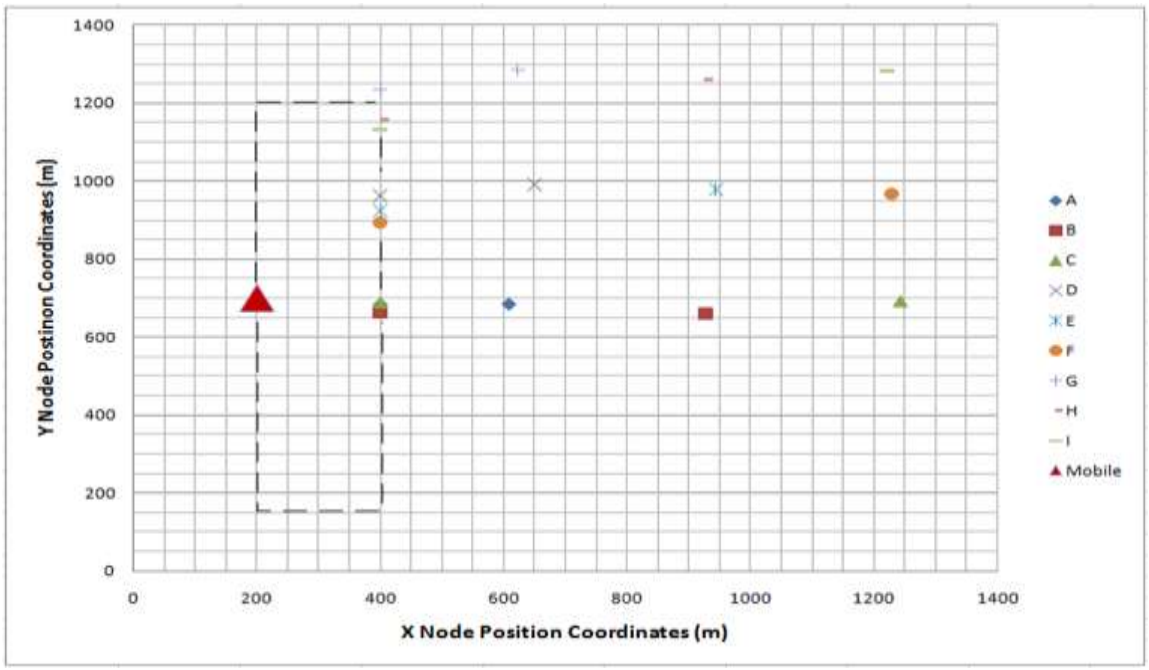


Figure 4.10 Theoretical positions of OENs relative to their matching destinations ($w_1 = 0.02$).

(d) OEN selection using combined delay (only) criteria: $w_1 = 0.00$

The simulation results for $w_1 = 0.00$ – delay is the only factor in OEN selection – can be seen in Figure 4.11. The respective theoretical OEN results are presented in Figure 4.13.

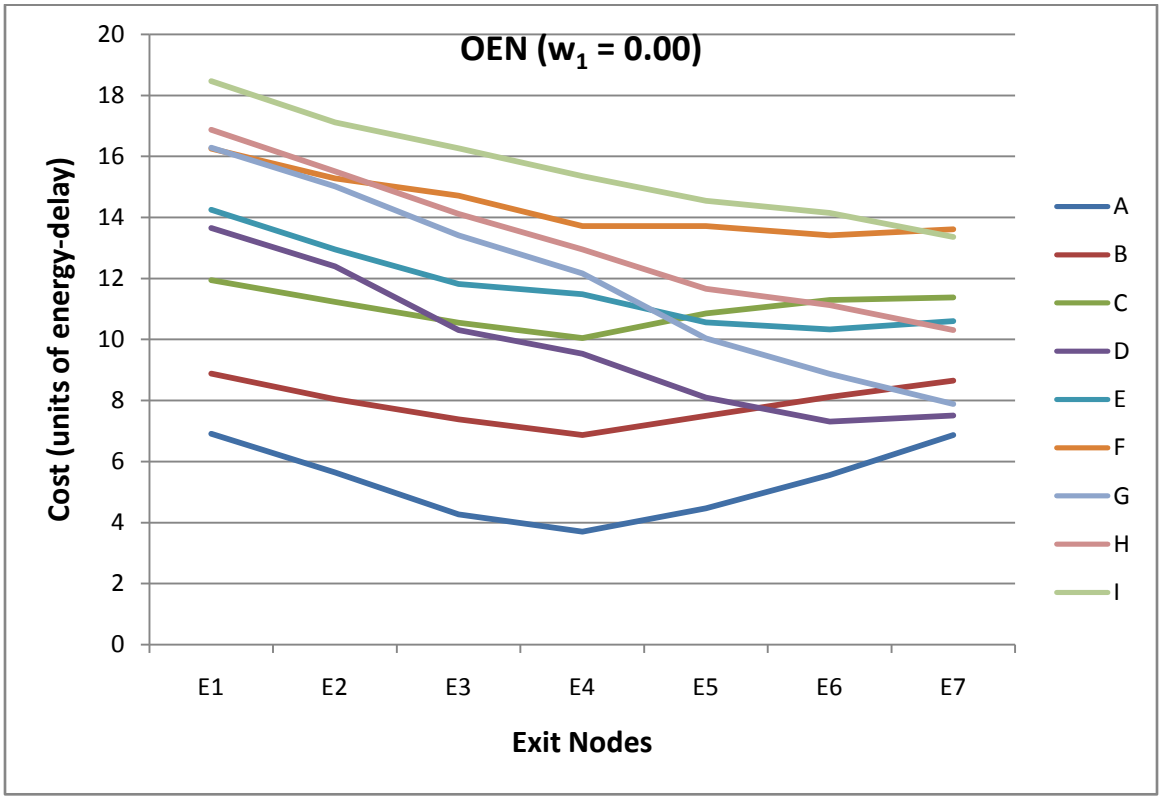


Figure 4.11 OEN simulation results: $w_1 = 0.00$

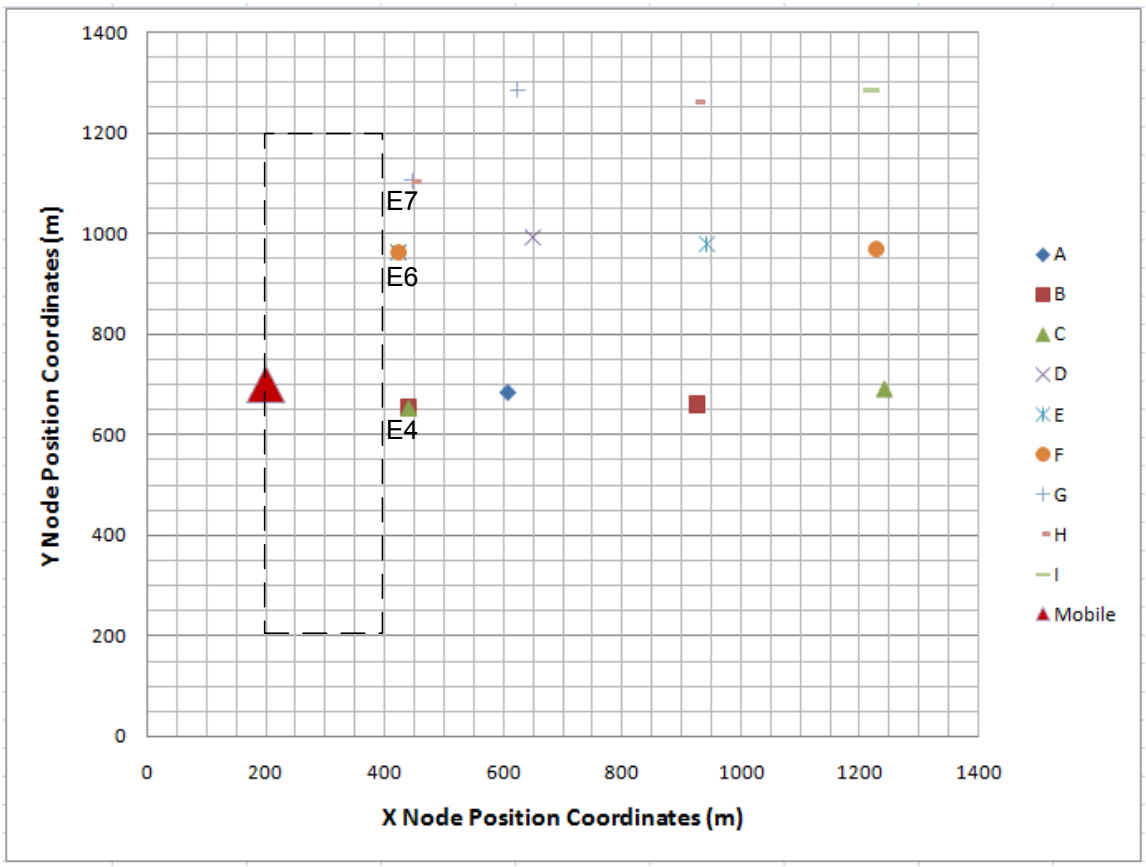


Figure 4.12 Theoretical positions of OENs relative to their matching destinations ($w_1 = 0.00$).

Based on Figure 4.11 and Figure 4.12, when $w_1 = 0.00$, the positions of optimal exit nodes are clearly such that they minimize the hop distance between themselves and their respective destination (i.e. sinks). Once again, as can be seen from

Figure 4.11 (or Figure 4.12) and Figure 4.13, there is a high correlation between the simulation-based and theoretically obtained results.

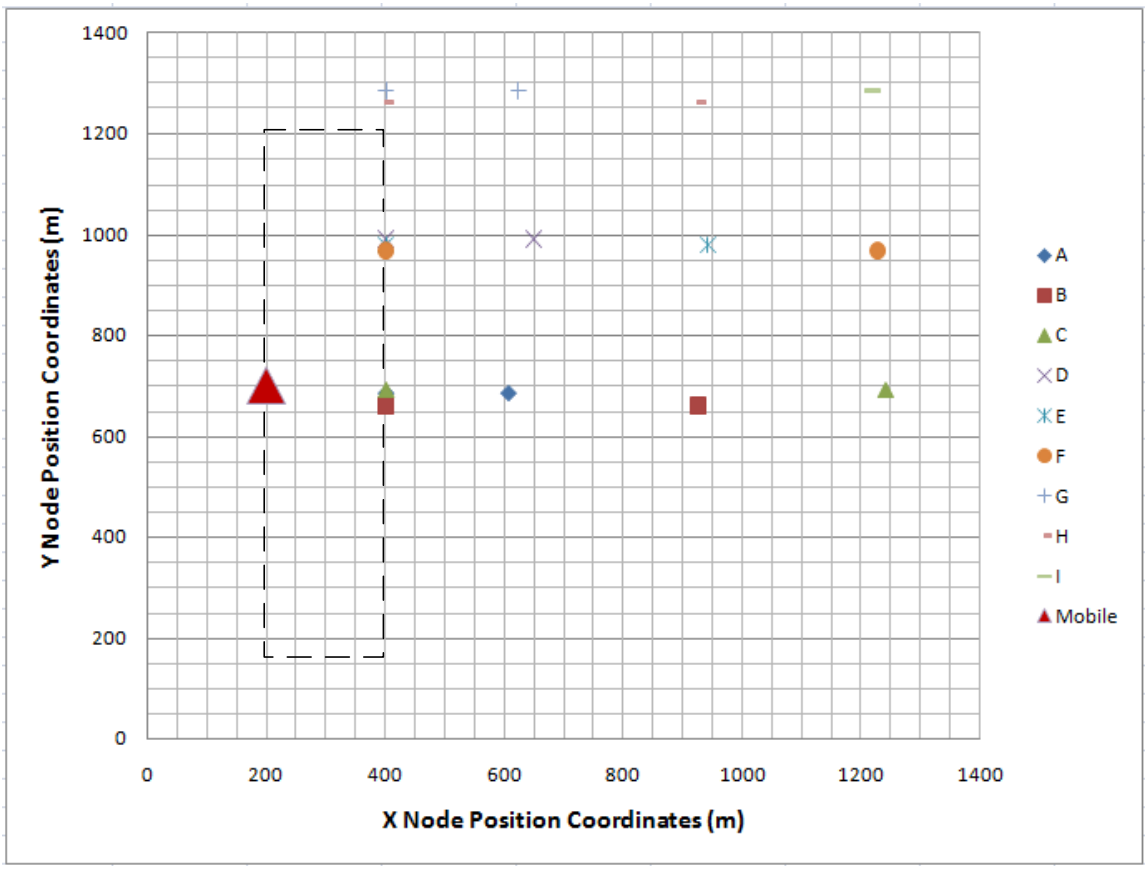


Figure 4.13 Theoretical positions of OENs relative to their matching destinations ($w_1 = 0.00$).

4.4. Simulation Experimentation 2: OPlaMoN-2

The main goal of the simulation-based experimentation described in this section was to determine whether the bidding values calculated by candidate boundary nodes using (3.10), i.e. (3.21), from Chapter 3, would be consistent with conditions found in real-

world WSNs and, as such, enable relatively inexpensive and quick deployment of the mobile bridge at the (most) optimal location in the interior of an observed routing hole.

4.4.1. Bidding Value Calculation – Simulation Setup and Execution

The validation of OPlaMoN-2 Bidding Procedure involved two separate sets of simulations – one assuming six and the other three traffic streams – with the corresponding network topologies consisting of virtual grids of 17x60 and 24x40 cells and routing holes occupying 3x50 cells and 8x25 cells respectively (see Figure 4.14 and Figure 4.15). In the two figures, the actual locations/directions of the traffic streams are marked with yellow arrows, and their respective data rates are indicated underneath. The yellow shaded square spans over the area, i.e. cells, containing 'candidate node's (i.e. potential mobile bridge hosts). Finally, the cells labeled with letters (**A** through **F** for case 1, and **A** through **D** for case 2) identify the possible data-sink locations.

Each of the two sets of simulations was executed three times on randomized network topologies (i.e. network nodes were placed randomly in virtual grids), where in each simulation run, a mobile node was placed near a different candidate node (i.e. $nb(i), i = 1 \dots 25|50$). Using available information about the traffic streams present in the network and general network topology, each of the mobile node deployment candidates would have calculated its bidding values for each of the destinations using expression (3.21) from Chapter 3, and logged these values for future comparison with the simulation results. At the same time, based on the mobile node location

information, the destination information and the locations of the boundary nodes, each of the inflow nodes ($nbs_k(1)$) would have decided whether to forward packets around the routing hole using perimeter routing or to take advantage of the mobile node bridge. The data required for calculating the cost (based on the combined delay/energy metric) was recorded at each hop by each transmitted packet, with the combined cost calculated at the final destination. The recorded data included information such as: the number of traversed hops, the (physical) length of each hop, and whether a mobile bridge was used or not. At the completion of each of the three simulation set runs, the average of the results was obtained for the purpose of mitigating the effects of outliers caused by random distribution of network nodes and in order to ensure the simulations/results consistency.

The simulations were performed for four different values of weight w_1(see (3.12)):

- $w_1 = 1.00$ – only energy cost was considered;
- $w_1 = 0.20$ – both cost metrics, energy and delay, were combined;
- $w_1 = 0.02$ – major emphasis was put on delay cost;
- $w_1 = 0.00$ – only delay cost was considered.

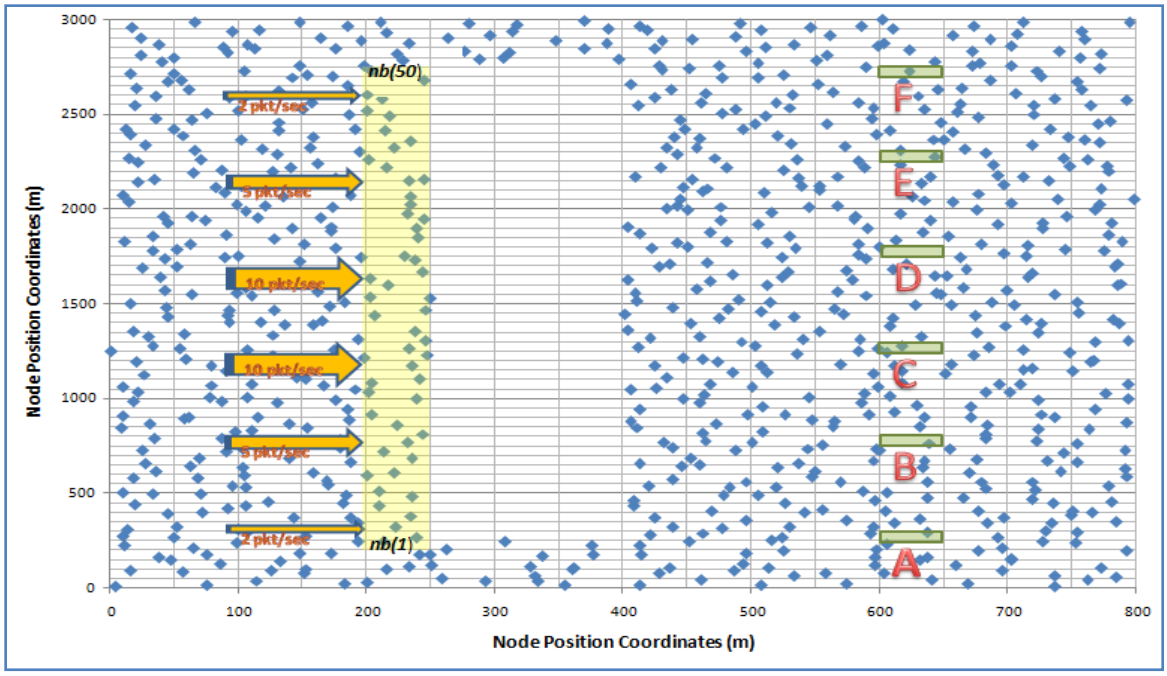


Figure 4.14 OPlaMoN-2 in a network with six streams impeded by routing hole.

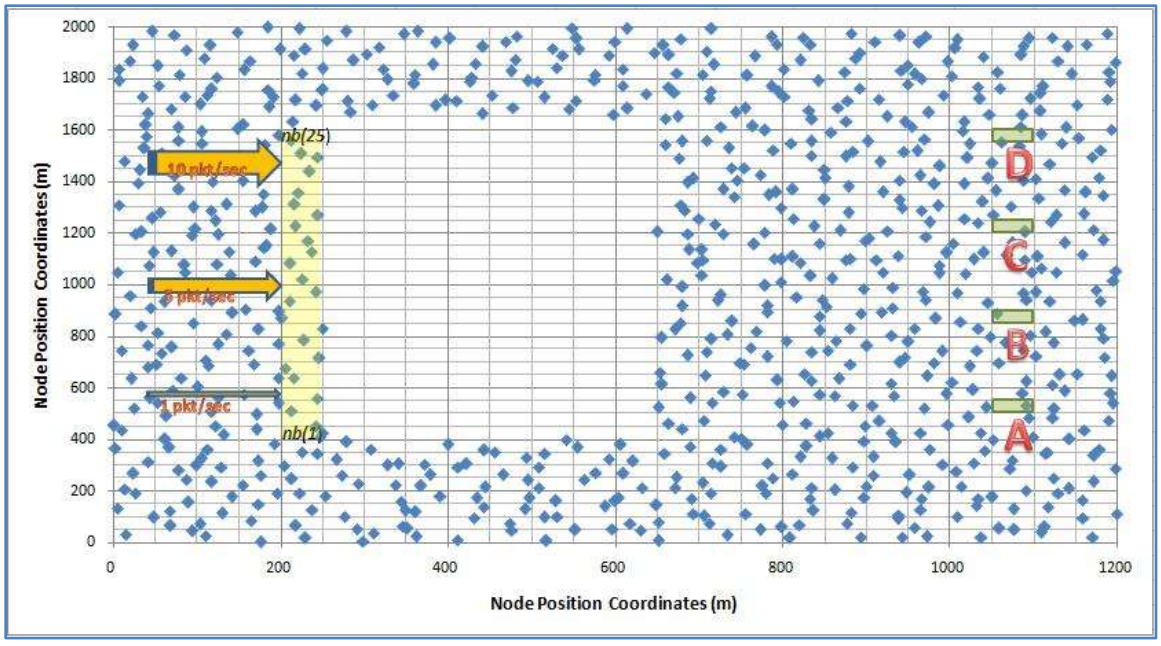


Figure 4.15 OPlaMoN-2 in a network with three streams impeded by routing hole.

4.4.2. Simulation Results: Six Streams Scenario

(a) OPlaMoN-2 in case of $w_1 = 1.00$

Figure 4.16 and Figure 4.17 present two types of results obtained while simulating the 6 traffic stream scenario, assuming $w_1 = 1.00$:

1) nodes' (OPlaMoN-2) bidding values, and

2) respective cost actually recorded in the network.

Table 4.1 provides a comparative overview of the most important results from the two figures. Examining the table and the figures it becomes apparent that the best bidder and the minimum cost candidate nodes are nearly identical.

More specifically, the shapes of the curves in the two graphs closely resemble each other, which signifies that in all cases, the bidding values are appropriate for identifying the optimal placement for the mobile node. As for any discrepancies between the actual costs and the calculated bids, these can be attributed to the random distribution of nodes throughout the network and the imprecise estimation of the node distribution constant k ($k = 1.18$ was used to counteract the effects of random node distribution). The value of k was calculated through a separate simulation in which the average hop distance was determined (value of about 120) and used as the divisor of the maximum hop distance r ($r = 142$ i.e. the distance that guarantees that a node can communicate/reach each of its eight immediate neighbours). As future work, it may be

desirable to devise theoretical methods for accurate calculation/estimation of k; and furthermore, calculation of k for individual data streams for even greater accuracy. However, even if precise estimation of k is achieved, some degree of error is still likely to exist due to the inherent randomness of network's node distribution.

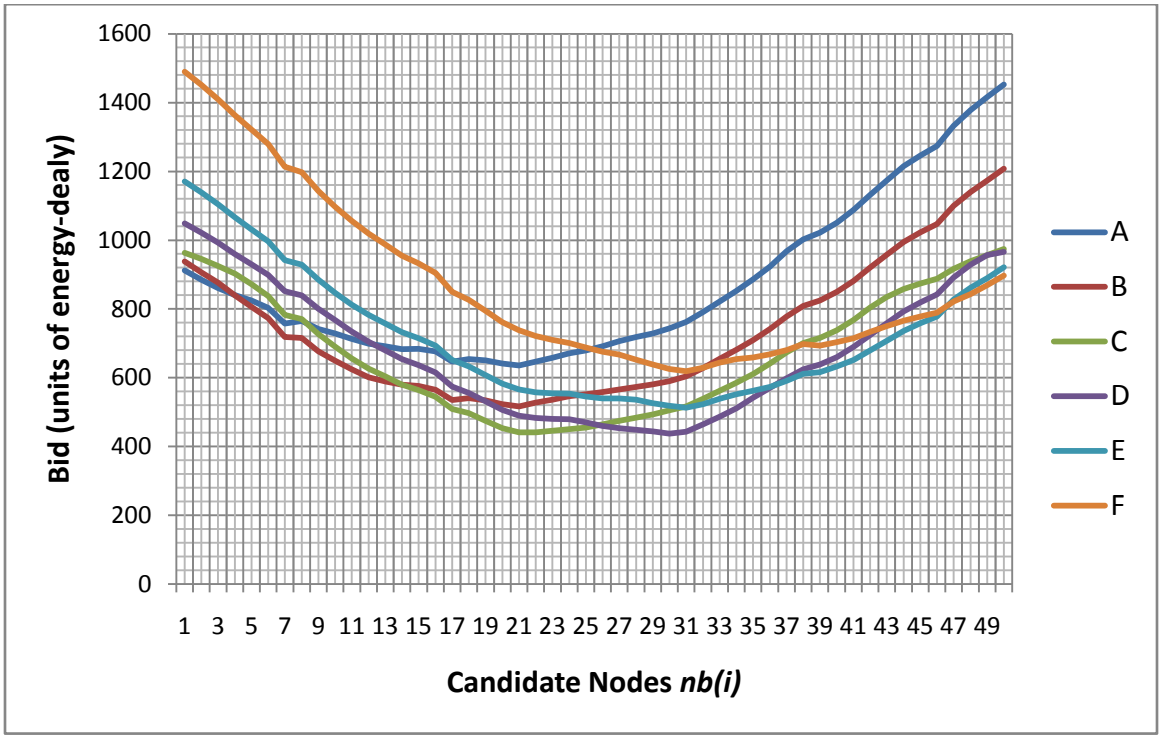


Figure 4.16 Bids calculated by each mobile node deployment candidate boundary node for each of the destinations ($w_1 = 1.00$, six stream scenario).

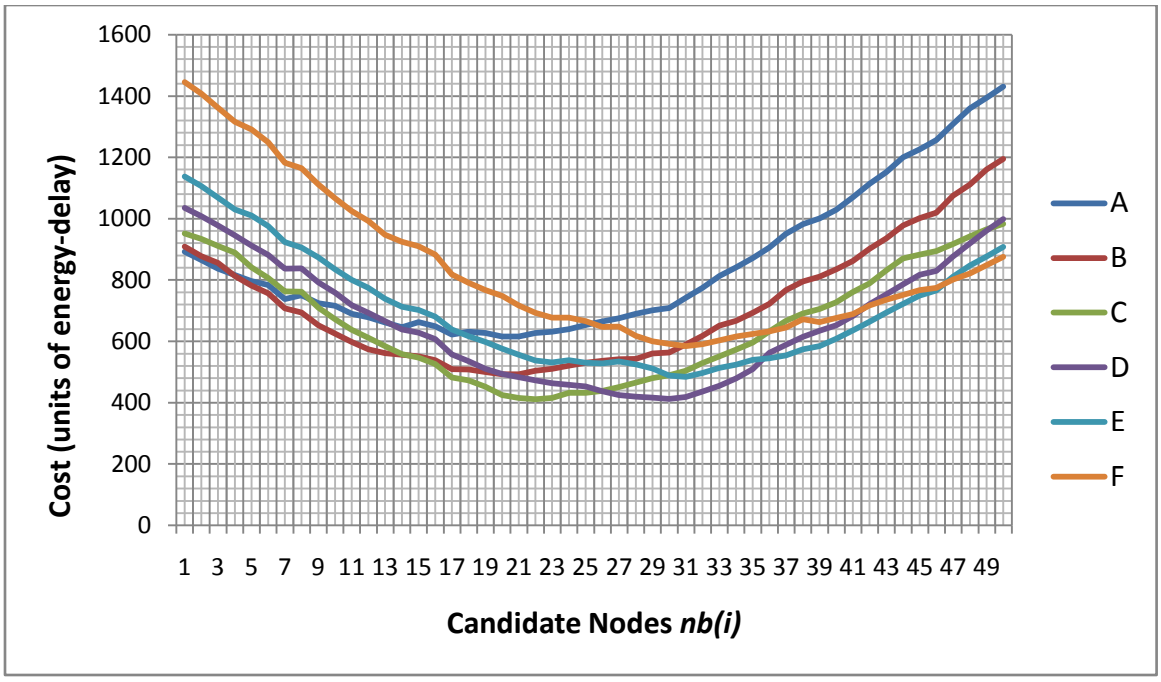


Figure 4.17 Actual costs calculated at the destination for each of the possible mobile node deployment positions ($w_1 = 1.00$, six stream scenario).

Destination (simulation scenario)	Best bidder based on (3.12)	Min-cost candidate as observed in the network	Min-cost % relative error[24]
A	Node 21	Node 20	0.074
B	Node 21	Node 20	0.092
C	Node 21	Node 22	0.920
D	Node 30	Node 30	0.000
E	Node 31	Node 31	0.000
F	Node 31	Node 31	0.000

Table 4.1 Comparative overview of the results for the six streams scenario: $w_1 = 1.00$.

[24] The relative percentage error is derived by finding the absolute value difference between the min-cost candidate node's cost and the best bidder's cost as observed in the network, and dividing by absolute value of the min-cost candidate node's cost and multiplying by 100 to convert to percent.

(b) OPlaMoN-2 in case of w_1 = 0.20

The simulation results for w_1 = 0.20 – delay becomes a factor in bidding value calculation – can be seen in Figure 4.18 and Figure 4.19. The most important results from the two figures are summarized in Table 4.2.

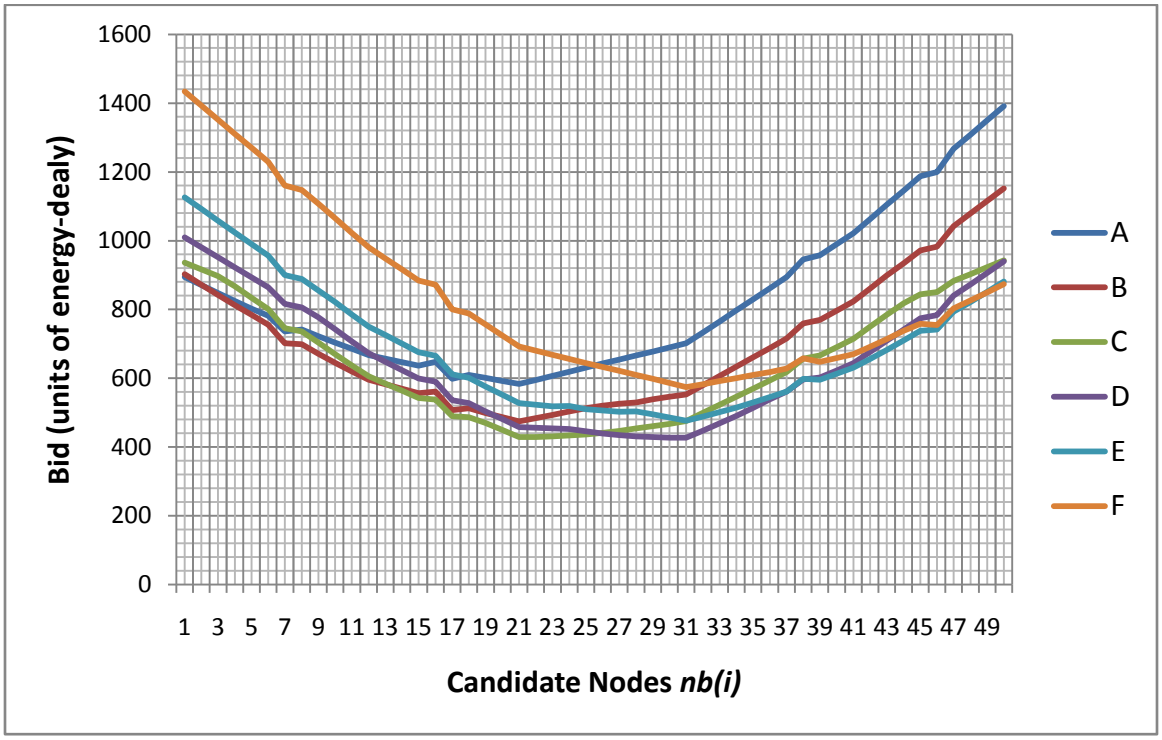


Figure 4.18 Bids calculated by each mobile node deployment candidate boundary node for each of the destinations ($w_1 = 0.20$, six stream scenario).

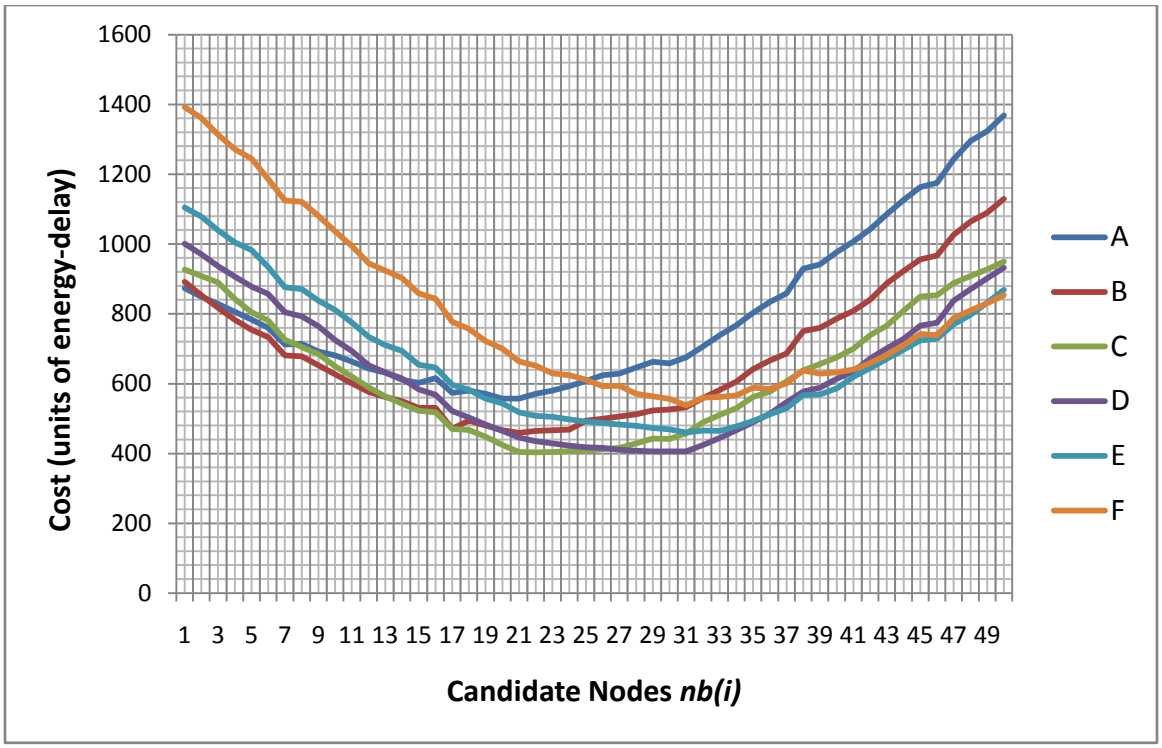


Figure 4.19 Actual costs calculated at the destination for each of the possible mobile node deployment positions ($w_1 = 0.20$, six stream scenario).

Destination (simulation scenario)	Best bidder based on (3.12)	Min-cost candidate as observed in the network	Min-cost % relative error
A	Node 21	Node 21	0.000
B	Node 21	Node 21	0.000
C	Node 21	Node 22	0.325
D	Node 31	Node 29	0.001
E	Node 31	Node 31	0.000
F	Node 31	Node 31	0.000

Table 4.2 Comparative overview of the results for the six streams scenario: $w_1 = 0.20$.

Examining Figure 4.18, Figure 4.19, and Table 4.2 it may be seen that the best bidder and the min-cost candidate values are nearly identical, with the shapes of the graphs closely resembling each other.

(c) OPlaMoN-2 in case of $w_1 = 0.02$

The simulation results for w_1 = 0.02 – delay becomes a significant factor in bidding value calculation – can be seen in Figure 4.20 and Figure 4.21. The most important results from the two figures are summarized in Table 4.4.

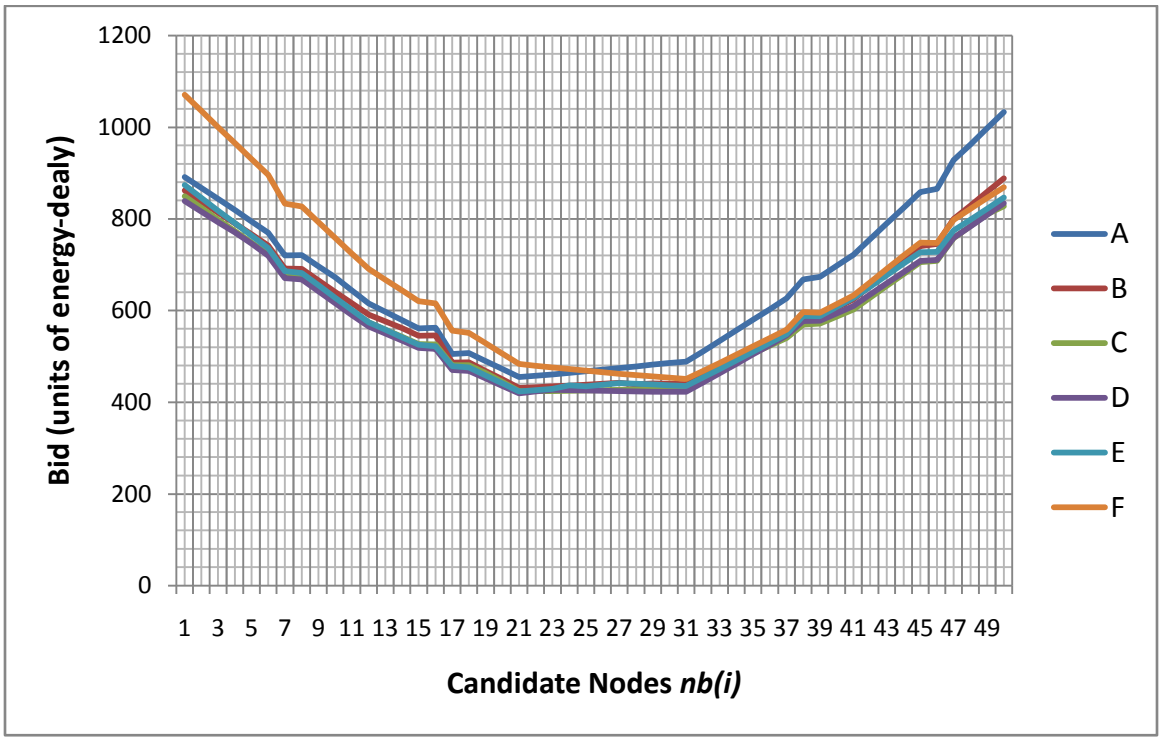


Figure 4.20 Bids calculated by each mobile node deployment candidate boundary node for each of the destinations ($w_1 = 0.02$, six stream scenario).

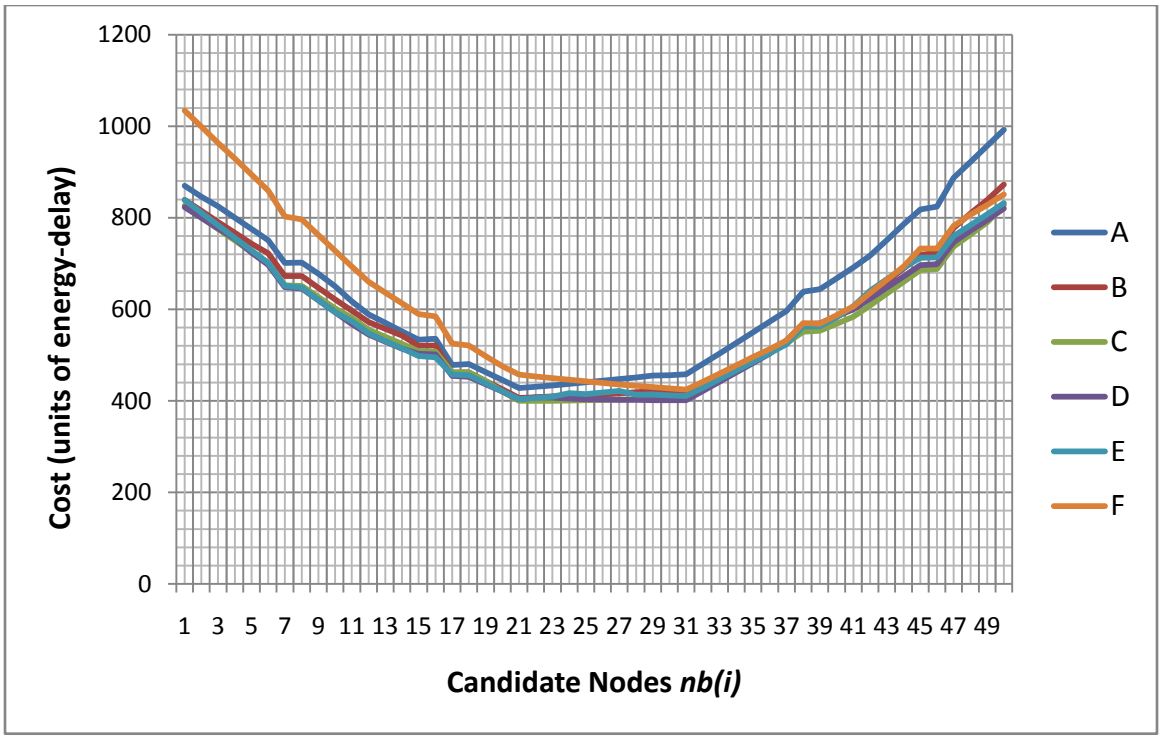


Figure 4.21 Actual costs calculated at the destination for each of the possible mobile node deployment positions ($w_1 = 0.02$, six stream scenario).

Destination (simulation scenario)	Best bidder based on (3.12)	Min-cost candidate as observed in the network	Min-cost % relative error
A	Node 21	Node 21	0.000
B	Node 21	Node 21	0.000
C	Node 21	Node 21	0.000
D	Node 31	Node 21	0.596
E	Node 21	Node 21	0.000
F	Node 31	Node 31	0.000

Table 4.3 Comparative overview of the results for the six streams scenario: $w_1 = 0.02$.

Table 4.3 demonstrates what appears to be discrepancy for destination **D** between best bidder and the observed minimum cost candidates. However, by examining Figure 4.20 and Figure 4.21, it may be seen that the graphs are relatively flat at the positions where the best bidder and the min-cost candidate are found. Therefore, there exists a large range of acceptable candidate nodes which values are in close proximity to the actual min-cost candidate node. Table 4.3 clearly confirms this as the relative error of the min-cost is less than 1%.

(d) OPlaMoN-2 in case of $w_1 = 0.00$

The simulation results for $w_1 = 0.00$ – delay is the only factor in bidding value calculation – can be seen in Figure 4.22 and Figure 4.23. The most important results from the two figures are summarized in Table 4.4.

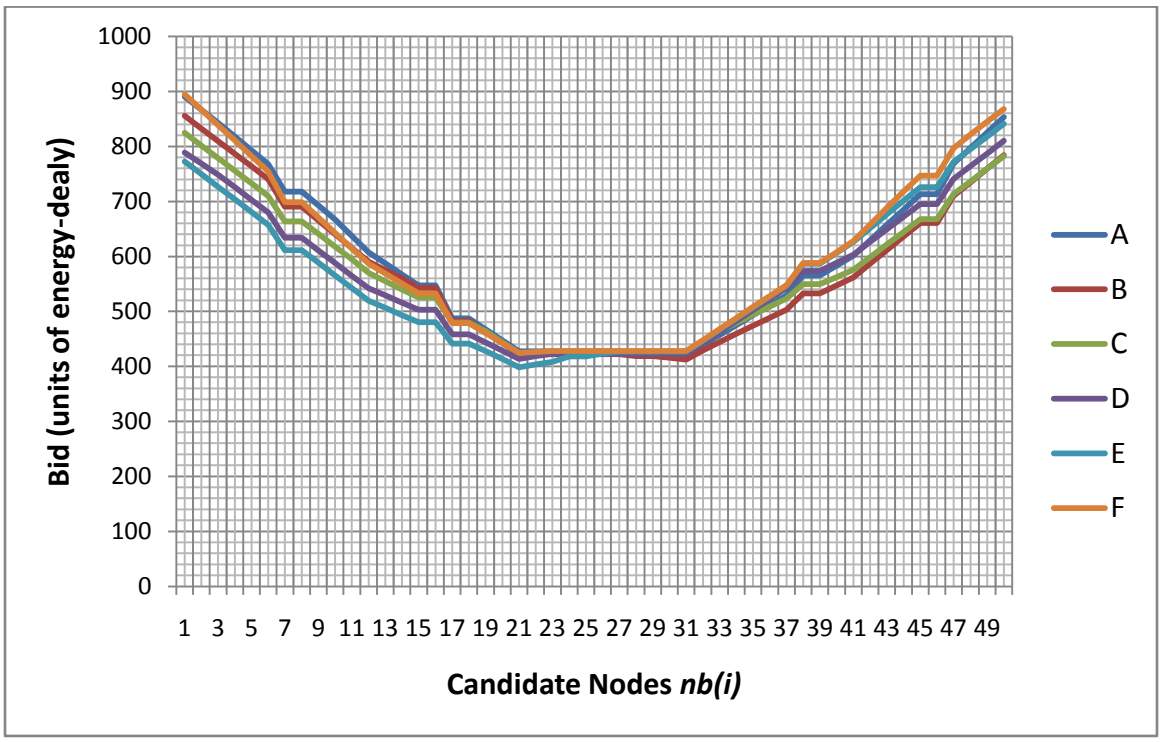


Figure 4.22 Bids calculated by each mobile node deployment candidate boundary node for each of the destinations ($w_1 = 0.00$, six stream scenario).

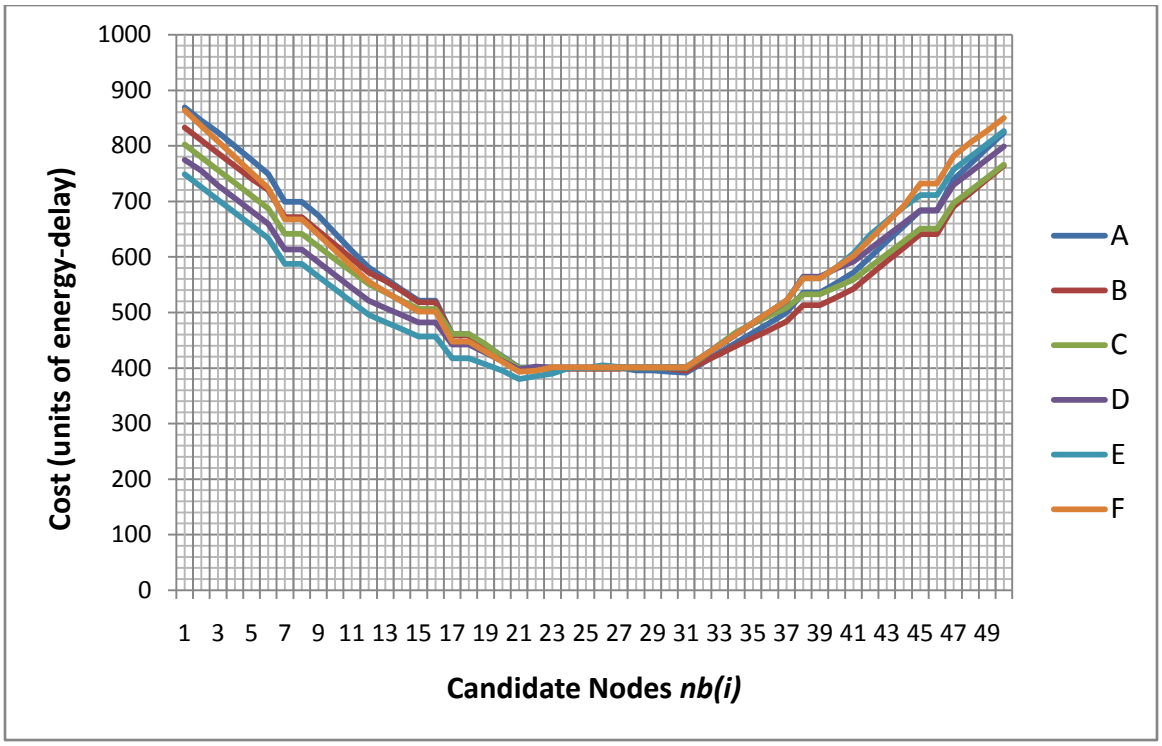


Figure 4.23 Actual costs calculated at the destination for each of the possible mobile node deployment positions ($w_1 = 0.00$, six stream scenario).

Destination (simulation scenario)	Best bidder based on (3.12)	Min-cost candidate as observed in the network	Min-cost % relative error
A	Node 31	Node 31	0.000
B	Node 31	Node 31	0.000
C	Node 21	Node 21	0.000
D	Node 21	Node 21	0.000
E	Node 21	Node 21	0.000
F	Node 21	Node 21	0.000

Table 4.4 Comparative overview of the results for the six streams scenario: $w_1 = 0.00$.

4.4.3. Simulation Results: Three Streams Scenario

The three streams scenario simulation results further corroborate the conclusions made from the six streams scenario simulation presented in the previous section.

(a) OPlaMoN-2 in case of $w_1 = 1.00$

The simulation results for $w_1 = 1.00$ – only energy is considered in bidding value calculation – can be seen in Figure 4.24 and Figure 4.25.

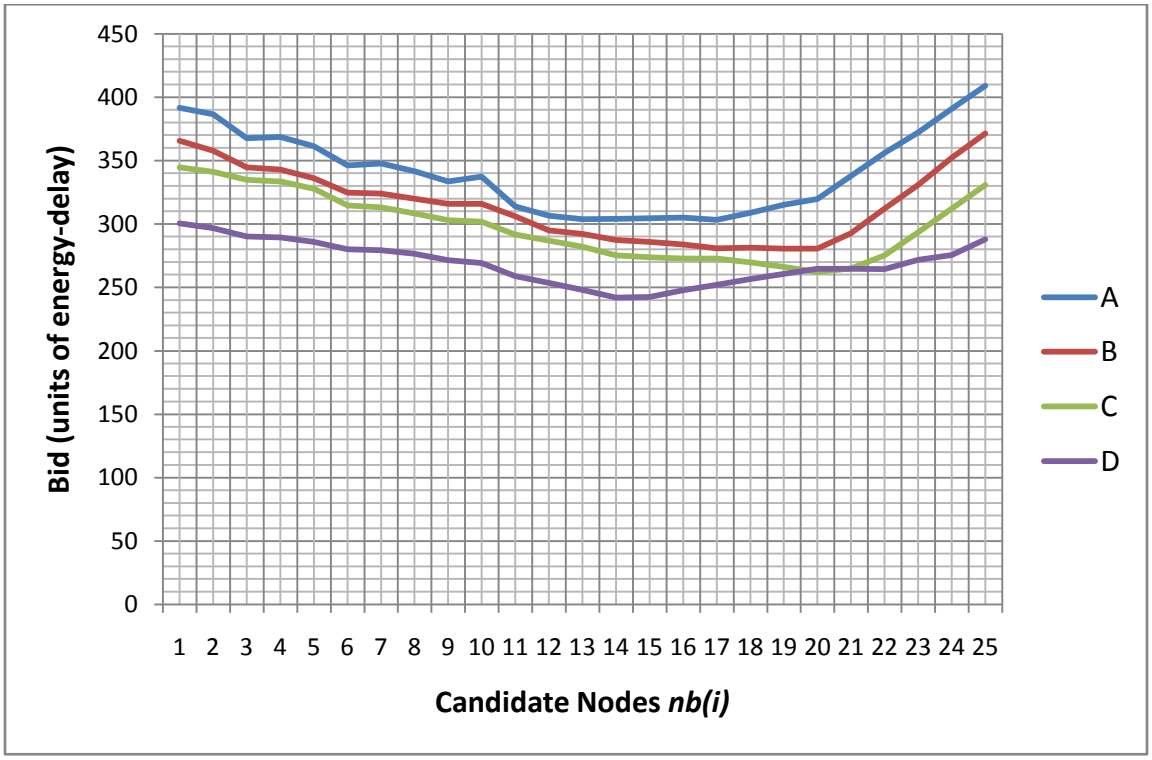


Figure 4.24 Bids calculated by each mobile node deployment candidate boundary node for each of the destinations ($w_1 = 1.00$, three stream scenario).

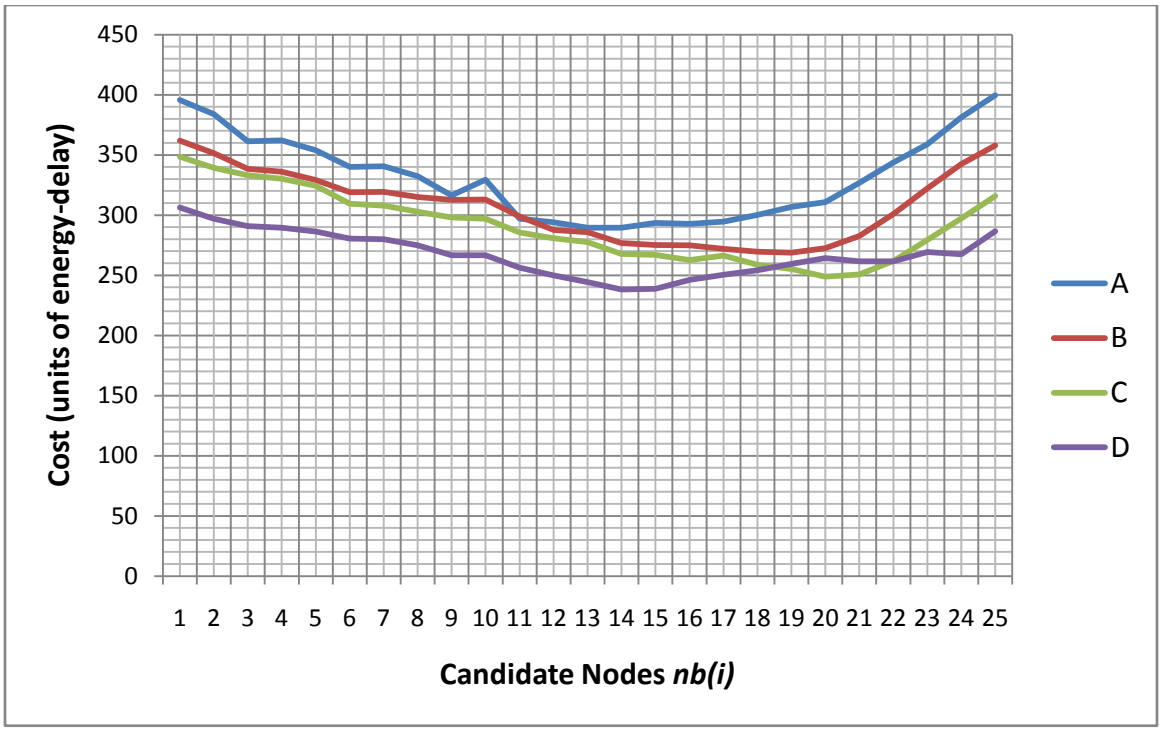


Figure 4.25 Actual costs calculated at the destination for each of the possible mobile node deployment positions ($w_1 = 1.00$, three stream scenario)

Destination (simulation scenario)	Best bidder based on (3.11)	Min-cost candidate as observed in the network	Min-cost % relative error
A	Node 17	Node 13	1.763
B	Node 19	Node 19	0
C	Node 20	Node 20	0
D	Node 14	Node 14	0

Table 4.5 Comparative overview of the results for the three streams scenario: $w_1 = 1.00$

(b) OPlaMoN-2 in case of $w_1 = 0.20$

The simulation results for $w_1 = 0.20$ – delay becomes a factor in bidding value calculation – can be seen in Figure 4.26 and Figure 4.27. The most important results from the two figures are summarized in Table 4.6.

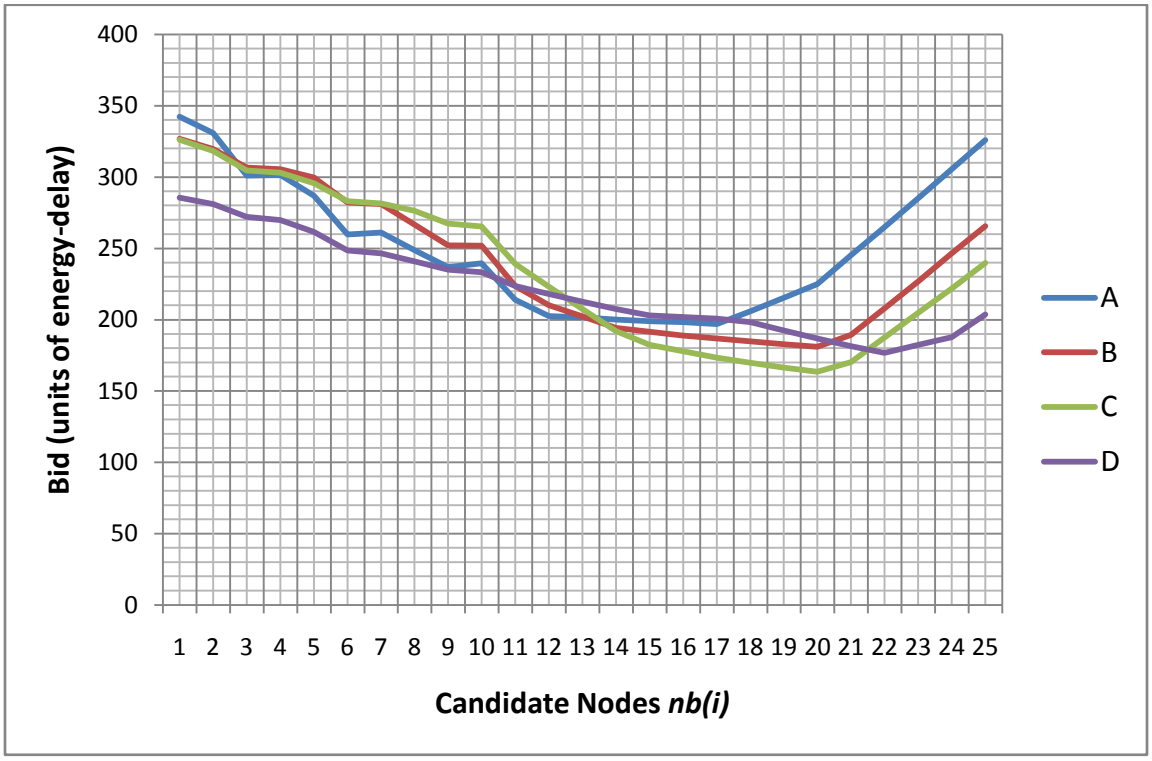


Figure 4.26 Bids calculated by each mobile node deployment candidate boundary node for each of the destinations ($w_1 = 0.20$, three stream scenario).

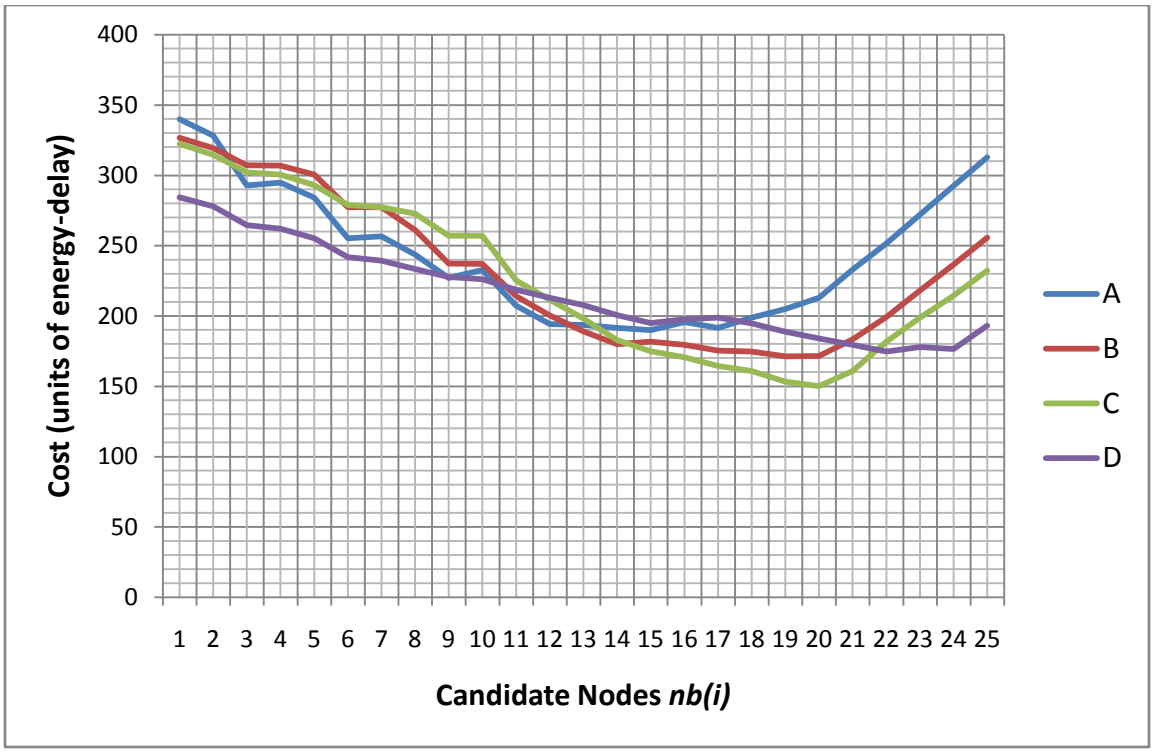


Figure 4.27 Actual costs calculated at the destination for each of the possible mobile node deployment positions ($w_1 = 0.20$, three stream scenario).

Destination (simulation scenario)	Best bidder based on (3.11)	Min-cost candidate as observed in the network	Min-cost % relative error
A	Node 17	Node 15	0.858
B	Node 20	Node19	0.199
C	Node 20	Node 20	0.000
D	Node 22	Node 22	0.000

Table 4.6 Comparative overview of the results for the three streams scenario: $w_1 = 0.20$.

(c) OPlaMoN-2 in case of $w_1 = 0.02$

The simulation results for $w_1 = 0.02$ – delay becomes a significant factor in bidding value calculation – can be seen in Figure 4.28 and Figure 4.29. The most important results from the two figures are summarized in Table 4.7.

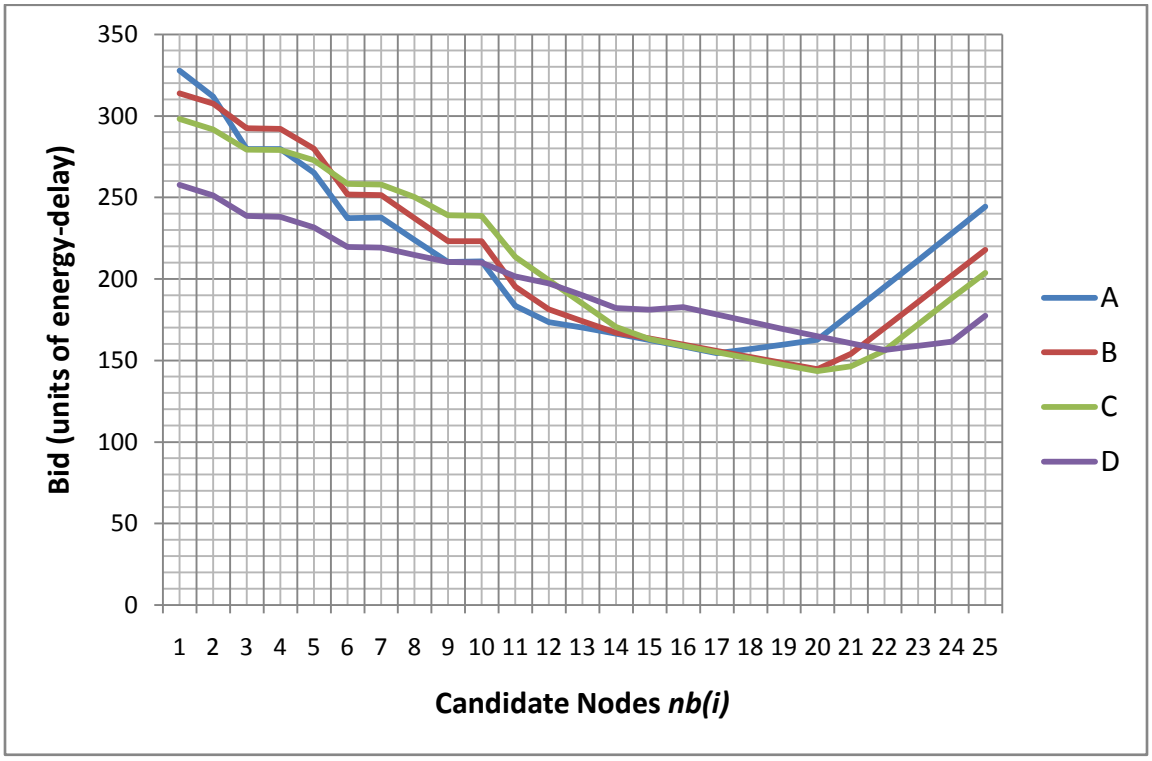


Figure 4.28 Bids calculated by each mobile node deployment candidate boundary node for each of the destinations ($w_1 = 0.02$, three stream scenario).

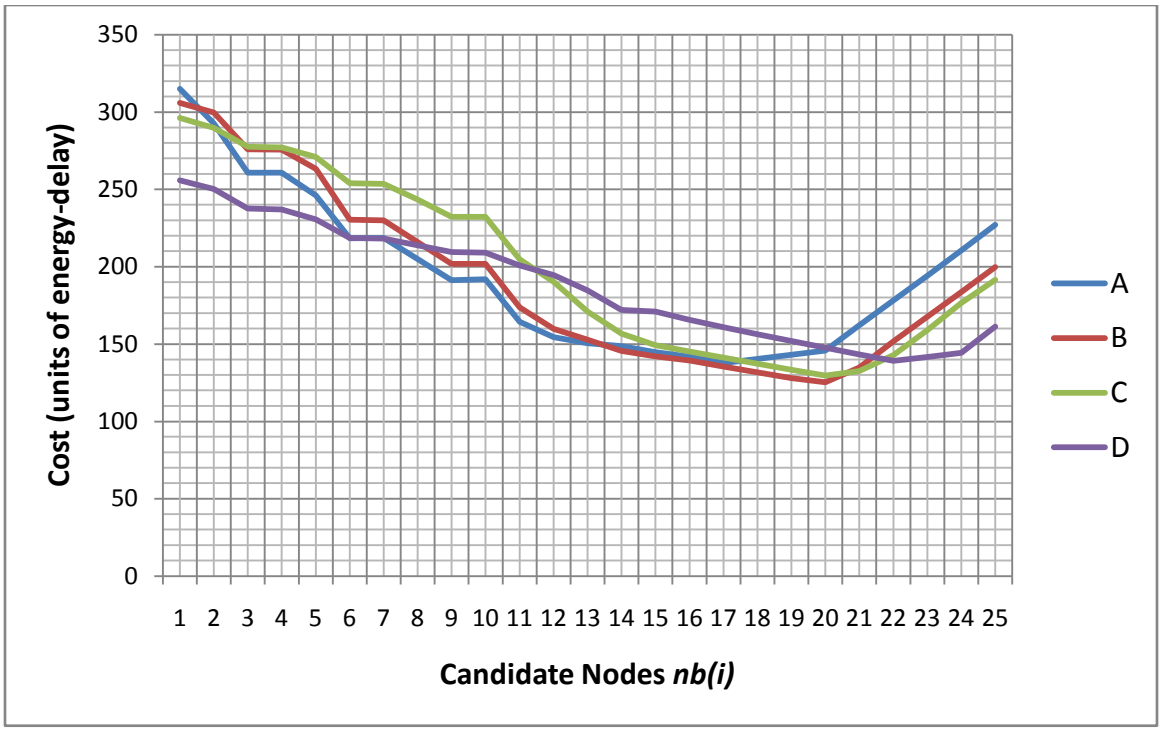


Figure 4.29 Actual costs calculated at the destination for each of the possible mobile node deployment positions ($w_1 = 0.02$, three stream scenario).

Destination (simulation scenario)	Best bidder based on (3.11)	Min-cost candidate as observed in the network	Min-cost % relative error
A	Node 17	Node 17	0.000
B	Node 20	Node 20	0.000
C	Node 20	Node 20	0.000
D	Node 22	Node 22	0.000

Table 4.7 Comparative overview of the results for the three streams scenario: $w_1 = 0.02$.

(d) OPlaMoN-2 in case of $w_1 = 0.00$

The simulation results for $w_1 = 0.00$ – delay becomes the only factor in bidding value calculation – can be seen in Figure 4.30 and Figure 4.31. The most important results from the two figures are summarized in Table 4.8.

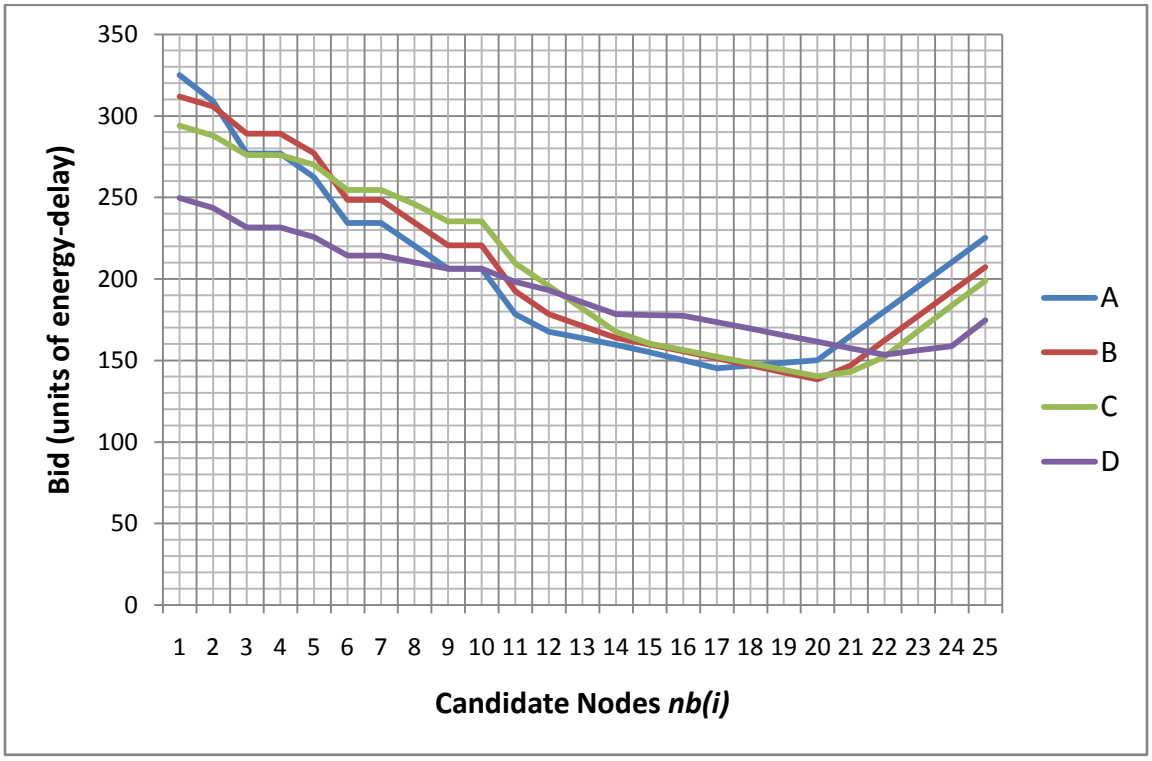


Figure 4.30 Bids calculated by each mobile node deployment candidate boundary node for each of the destinations ($w_1 = 0.00$, three stream scenario).

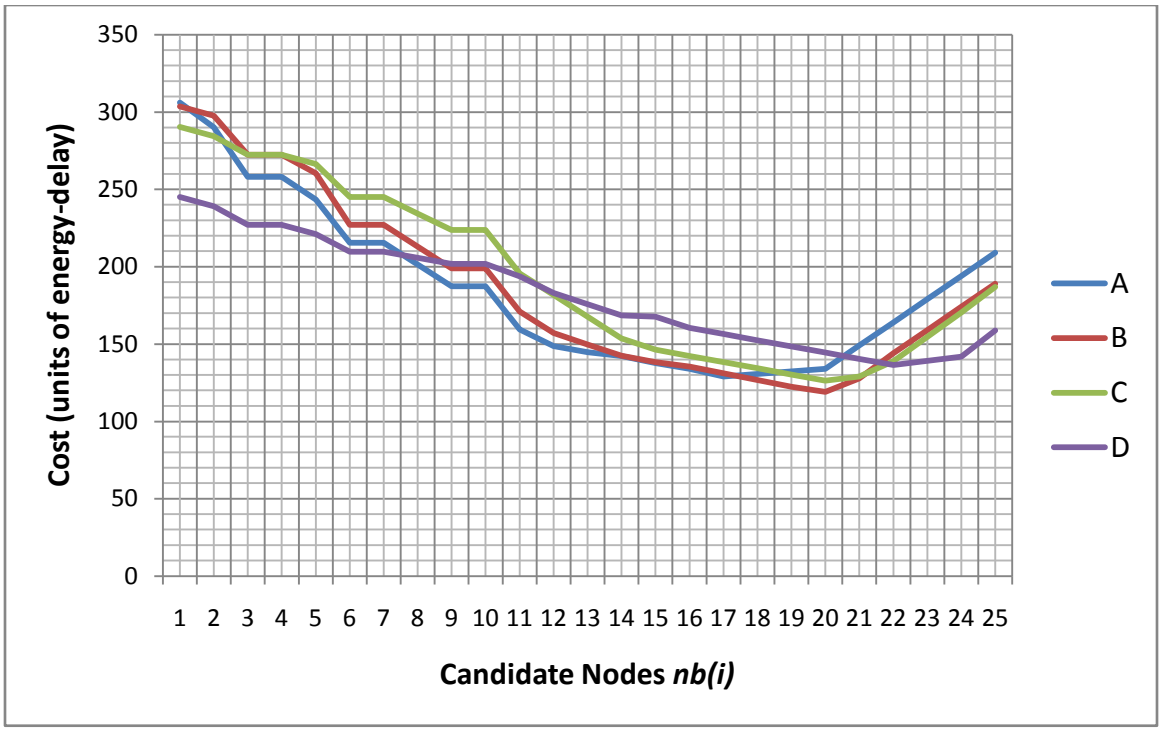


Figure 4.31 Actual costs calculated at the destination for each of the possible mobile node deployment positions ($w_1 = 0.00$, three stream scenario).

Destination (simulation scenario)	Best bidder based on (3.11)	Min-cost candidate as observed in the network	Min-cost % relative error
A	Node 17	Node 17	0.000
B	Node 21	Node 21	0.000
C	Node 21	Node 21	0.000
D	Node 22	Node 22	0.000

Table 4.8 Comparative overview of the results for the three streams scenario: $w_1 = 0.00$.

Consistent with the six streams scenario results, the three streams scenarios' estimated bids closely correlate with the actual costs for routing packet streams, and therefore correctly determine the optimal or near-optimal deployment positions of the mobile node.

Overall, we believe that the results of our simulations affirm the correctness of our theory, though improvements, such as parameter estimation (e.g. better estimation of k), may be subject to further investigation and improvement.

Chapter 5

Conclusions and Future Work

Through an extensive preliminary literature review we came to conclude that one of the most fundamental challenges of WSNs is that of energy supply – as network nodes are generally very limited in their energy supplies, yet are expected to operate for prolonged periods of time. Furthermore, based on research describing energy consumption models of WSNs, we have determined that in data intensive WSNs most of the energy is used for inter-nodal communications.

Given the apparent importance of minimizing energy consumption in WSNs, we chose to concentrate on routing holes – as relatively little research dealt with this type of network anomaly – and have determined that routing holes are detrimental to WSN network functionality since they can facilitate latency and uneven distribution of data

loads, which leads to disproportionate resource consumption and the possibility of network partitioning. Looking closer at mobile nodes as a potential solution for fixing routing holes, we have discovered little evidence of earlier research that tried to tackle the problem in this manner, though we did find other studies which were successful in improving WSN efficiency through the use of mobile nodes.

With the help of the newly defined concept of the *microhole,* we have examined the routing hole problem at the most basic level and have determined that even such trivial routing inefficiencies can be shown to have a cumulative detrimental effects over the network lifetime. Additionally, we have demonstrated that microholes can be successfully optimized by using mobile nodes to shorten the inter-hop distance of the routing inefficiency (i.e. using the mobile node to 'bridge' the communication gap created by the microhole).

The results of our analysis with respect to microholes has led us to conclude that our mobile bridge optimization technique is not limited to microholes, but can also be extended to traditional routing holes. Also, it became clear that our approach has to be carefully evaluated on a case-by-case basis, as it would not be advantageous in all scenarios. As a result, we have also introduced the concepts of *feasibility*, *justifiability* and *effectiveness* which can be used for autonomous decision making when determining whether it is cost effective to deploy a mobile node to a microhole or even deploying mobile nodes in a WSN in general.

Since in [52] it was demonstrated that energy alone is often insufficient for justifying the use of a mobile node bridge, we have introduced a new weighted metric which combines energy and delay and can be used by mobile nodes for deciding whether they should deploy to a routing hole. We further extended the work described in [52] to a *1-Hole/1-Moble/n-Traffic-Stream* scenario in which a distributed algorithm (OPlaMoN-2) is used by boundary nodes of the routing hole to distributively bid for a mobile node to deploy besides them. The theoretical analysis of OPlaMoN-2 produced interesting results which included, but were not limited to the following:

- The width of a rectangular routing hole has no effect on the position of the exit node.
- When only energy is considered, the distance between the theoretical optimal exit node position and the deployed mobile node in nearly minimized (with slight deviation due to the position of the destination).
- With increased weight (i.e. importance) of delay there is a tradeoff between how close the optimal exit node is placed to the mobile node (i.e. energy is conserved) or to the destination (i.e. delay is reduced).
- OPlaMoN-2 runs in $\mathrm{O}(N)$ with respect to the number (N) of boundary nodes.

The correctness of OPlaMoN-2 was validated with extensive simulations performed using the Qualnet simulator. The simulation results were in line with our theoretical

predictions, though some imperfections did exist due to randomness of the network topology.

Overall, we believe that our work serves as an important milestone for further investigation into the area of self-healing Hybrid WSNs and provides for a strong theoretical foundation for implementation of real world prototypes. Though in our work we only dealt with single hole/mobile scenarios, future research is open to examining the cases where there are multiple routing holes as well as multiple mobile nodes (see Figure 5.1).

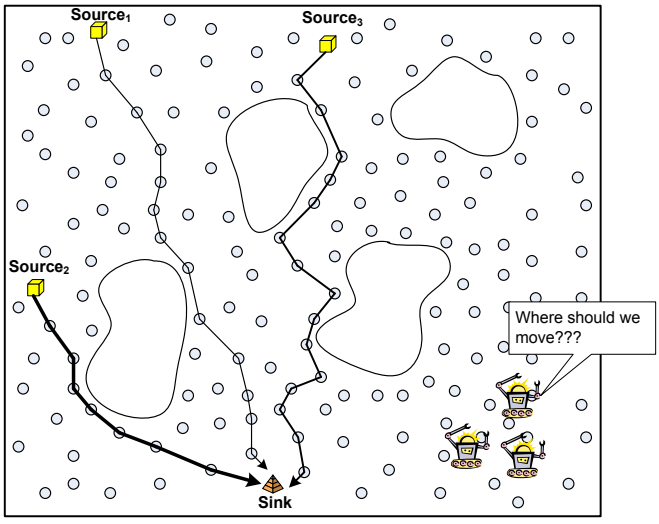


Figure 5.1 Multiple hole/stream/mobile nodes scenario. Where should the mobile nodes move to maximize their usefulness?

In such cases, OPlaMoN-2 has to be incorporated into a larger, network-wide algorithm, which involves the mobile nodes ranking the routing holes and distributively coordinating the deployments to routing holes among themselves so that the most

appropriate mobiles reach the most appropriate routing holes[25]. A step further would be to consider the possibility of multiple mobiles deploying to a single routing hole (see Figure 5.2) or the effects of positions and shapes of multiple holes on data load distribution (see Figure 5.3).

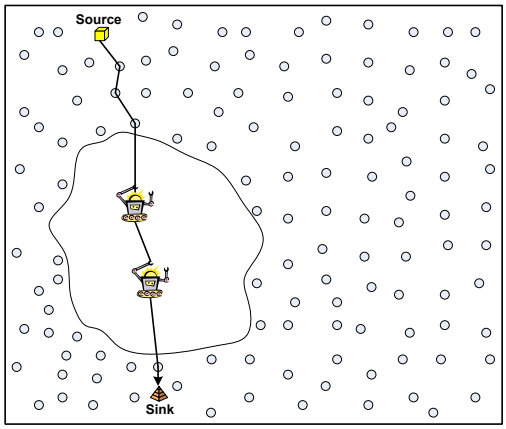


Figure 5.2 Multiple mobile nodes are used to bridge the routing hole.

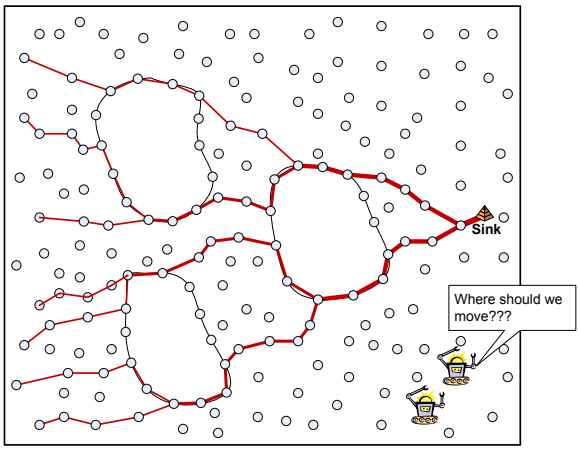


Figure 5.3 Multiple cascading routing holes force streams to converge.

[25] In [69] we have proposed such a Distributed Algorithm for Routing Microhole Abolishment (DARMA) which principles can be extended to combating traditional routing holes.

Other, more immediate research topics which can be improved include: parameter estimation; calculation or estimation of mobile node effectiveness in WSNs containing regular routing holes (as opposed to microholes); and extension of the bidding function beyond the energy/delay parameters.

Bibliography

1. **Zhao, F. and Guibas, L.** *Wireless Sensor Networks: An information Processing Approach.* San Francisco : Morgan Kaufmann Publishers Inc, 2004.

2. Wireless Sensor Networks. [Online] Crossbow Technologies. http://www.xbow.com/.

3. ScatterWeb: The self-configuring wireless communication platform. [Online] ScatterWeb GmbH. http://www.scatterweb.com/.

4. Intel Mote: Sensor Nets / RFID. [Online] http://www.intel.com/research/exploratory/motes.htm.

5. *Wireless sensor networks solutions for real time monitoring of nuclear power plant.* **Lin, Ruizhong, Wang, Zhi and Sun, Youxian.** s.l. : WCICA, 15-19 June 2004. Fifth World Congress on Intelligent Control and Automation. Vol. 4, pp. 3663-3667.

6. *Towards resilient geographic routing in WSNs.* **Abu-Ghazaleh, Nael, Kang, Kyoung-Don and Liu, Ke.** Montreal, Quebec, Canada : ACM, 2005. Q2SWinet 'o5: Proceeding of the 1st ACM international workshop on Quality of service & secuirty in wireless and mobile networks. pp. 71-78. 1-59593-241-0.

7. *Guest Editors' Introduction: Overview of Sensor Networks.* **Culler, David, Estrin, Deborah and Srivastava, Mani.** 8, August 2004, IEEE Computer, Vol. 37.

8. *Instrumenting Network Simulators for Evaluating Energy Consumption in Power-Aware Ad-Hoc Network Protocols.* **Margi, C. B and Obraczka, K.** Washington, DC : IEEE Computer Society, 2004. Proceedings of the the IEEE Computer Society's 12th Annual international Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (Mascots'04). Vol. 00, pp. 337-346.

9. *Energy and rate based MAC protocol for wireless sensor networks.* **Kannan, R., et al.** s.l. : SIGMOD, 2003. pp. Rec. 32, 4 (Dec. 2003), 60-65.

10. *Energy efficiency design challenge in sensor networks.* **Gao, Q., et al.** London, UK : LCS, 2002. pp. 69-72.

11. **T., Rapport.** *Wireless Communications: Principles and Practice.* Upper Saddle River : Prentice Hall PTR, 2001.

12. *Localized minimum-energy broadcasting in ad-hoc networks.* **J., Cartigny, D., Simplot and I., Stojmenovic.** s.l. : INFOCOM'03, 2003. Proc. of the 22nd Annual Joint Conference of the IEEE Comuter and Communications Societies.

13. *Data Gathering in Sensor Networks using the Energy*Delay Metric.* **Lindsey, S., Raghavendra, C. S. and Sivalingam, K.** s.l. : IPDPS, 2001. In Proc. of IPDPS Workshop on Issues in Wireless Networks and Mobile Computing.

14. *Cross-layer wireless sensor network radio power management.* **Brownfield, M.I., et al.** 2006. Wireless Communications and Networking Conference, 2006. WCNC 2006. IEEE. Vol. 2, pp. 1160-1165.

15. *Energy-efficient communication protocol for wireless microsensor networks.* **Heinzelman, W.R., Chandrakasan, A. and Balakrishnan, H.** MIT, Cambridge, MA, USA : s.n., January 2000. Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, 2000. Vol. 2, p. 10.

16. *Study on Communication Mode of Wireless Sensor Networks Based on Effective Result.* **Shi, J. F., Zhong, X.X. and Chen, S.** s.l. : Institute of Physics Publishing, 2006, Journal of Physics, Vol. Conference Series 48, pp. 1317-1321.

17. **Ahmed, N., Kanhere, S. S. and Jha, S.** The holes problem in wireless sensor networks: a survey. *Mobile Computing and Communications Review.* April 2005, Vol. 1, 2, pp. 4-18.

18. *GPSR: Greedy perimeter stateless routing for wireless networks.* **Karp, Brad and Kung, H. T.** s.l. : ACM/IEEE, 2000. MobiCom '2000. pp. 243-254.

19. *Compas routing on geometric networks.* **Kranakis, Evangelos, Singh, Harvinder and Urrutia., Jorge.** 1999. Proceeding of the 11th Canadian Conference on Computational Geometry. pp. 51-54.

20. *Routing with guranteed delivery in ad hoc wireless networks.* **Bose, P., et al.** 2001, Wireless Netowrks, Vol. 7, pp. 609-616.

21. *Geometric ad-hoc routing: Of theory and practice.* **Kuhn, Fabian, et al.** July 2003. 23rd ACM Symposium on Principles of Distributed Computing (PODC '03).

22. *Locating and bypassing routing holes in sensor networks.* **Fang, Q., Gao, J. and Guibas, L.** May 2004. 23rd Conference of the IEEE Communications Society (Infocom).

23. **S., Douglas, Couto, J. De and Morris, Robert.** *Location proxies and intermediate node forwarding for practical geographic forwarding.* Laboratory for Computer Science, MIT. s.l. : MITLCS-TR-824, June 2001. Technical Report.

24. *GLIDER: gradient landmark-based distributed routing for sensor netowrks.* **Fang, Qing, et al.** s.l. : INFOCOM, 2005. pp. 339-350.

25. *WEAR: a balanced, fault-tolerant, energy-aware routing protocol in WSNs.* **Sha, Kewei, Du, Junzhao and Shi, Weisong.** 3/4, 2006, International Journal of Sensor Networks, Vol. 1, pp. 156-168.

26. *Movement-assisted sensor deployment.* **Wang, Guiling, Cao, Guohong and Porta, Tom La.** s.l. : IEEE, June 2004. INFOCOM 2004.

27. *Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem.* **Howard, Andrew, Mataric, Maja J. and Sukhatme, Gaurav S.** June 2002. 6th International Symposium on Distributed Autonomous Robotics Systems (DARS02).

28. **Shucker, Brian and Bennett, John K.** *Virtual Spring Mesh Algorithms for Control of Distributed Robotic Macrosensors.* Department of Computer Science, University of Colorado at Boulder. May 2005. Technical Report CU-CS-996-05.

29. **Howard, Andrew, Mataric, Maja J. and Sukhatme, Gaurav S.** An incremental self-deployment algorithm for mobile sensor networks. *Autonomous Robots, Special Issue on Intelligent Embedded Systems.* September 2002, Vol. 13, 2, pp. 113-126.

30. *Crossbow technologies, inc.* [Online] 2006. http://www.xbow.com.

31. *A Tny Mobile Robot Platform for Large-Scale Sensor Networks.* **Gabriel T. Sibley, Mohammad H. Rahimi and Gaurav S. Sukhatme,.** 2002. Proceedings of the IEEE International Conference on Robotics and Automation (ICRA2002).

32. PIONEER P3-DX. *MobileRobots Inc.* [Online] http://www.activrobots.com/ROBOTS/p2dx.html.

33. PIONEER P3-AT. [Online] MobileRobots Inc. http://www.activrobots.com/ROBOTS/p2at.html.

34. *K-Team.* [Online] 2008. http://www.k-team.com/.

35. *Surveyor Corporation.* [Online] 2008. http://www.surveyor.com/.

36. *Autonomous deployment and repair of a sensor netowrk using an unmanned aerial vehicle.* **Croke, Peter, et al.** s.l. : IEEE, May 2004. Proceedings of the IEEE 2004 International Conference on Robotics and Automation. Vol. 4, pp. 3602-3608.

37. *The Analysis of an Efficient Algrithm for Robot Coverage and Exploration based on Sensor Network Deployment.* **Batalin, Maxim and Sukhatme, Gaurav S.** Barcelona, Spain : s.n., Apr 2005. In IEEE International Conference on Robotics and Automation. pp. 3489-3496.

38. *A bidding protocol for deploying mobile sensors.* **Wang, Guiling, Cao, Guohong and Porta, Tom La.** Nov 2003. In 11th IEEE International Conference on Network Protocols ICNP '03. pp. 315-324.

39. *Improving sensor network performance by deploying mobile sensors.* **Du, X. and Lin, F.** s.l. : IEEE, April 2005. Proceedings of the 24th IEEE International Performance, Computing, and Communications Conference IPCCC 2005. pp. 67-71.

40. *Locating and bypassing routing holes in sensor networks.* **Fang, Qing, Gao, Jie and Guibas, Leonidas J.** s.l. : IEEE, June 2004. INFOCOM 2004.

41. **Rappaport, T.** *Wireless Communications: Principles and Practice.* Upper Saddle River : Prentice Hall PTR, 2001.

42. *Effects of natural propagation environments on wireless sensor network coverage area.* **Fanimokun, A and Frolik, J.** March 2003. Proceedings of the 35th Southeastern Symposium on System Theory, 2003. pp. 16-20.

43. *Wireless sensor networks for habitat monitoring.* **Mainwaring, A., et al.** New York, NY, USA : ACM Press, 2002. WSNA '02: Proceedings fo the 1st ACM international workshop on Wireless sensor networks and applications. pp. 99-97.

44. ENERGIZER. [Online] http://data.energizer.com/PDFs/E91.pdf.

45. **Vlajic, N., Moniz, N. and Portnoy, M.** Self-Healing Wireless Sensor Networks. *Cooperative Wireless Communications.* s.l. : to be published by Auerbach Publications, CRC Press, Taylor&Francis Group, 2007.

46. *Near-Optimal Node Clustering In Wireless Sensor Networks.* **Xai, D. and Vlajic, N.** Washington, DC, USA : IEEE Comp. Society, 2007. Proc. 21st Conf. Advanced Networking and Applications. pp. 632-641.

47. Maplesoft. [Online] a division of Waterloo Maple Inc., 2008. http://www.maplesoft.com/.

48. **Feldman, Allan M.** *Welfare Economics and Social Choice Theory.* Kluwer, Boston : s.n., 1980.

49. *Functionally Accurate, Cooperative Distributed Systems.* **Lesser, V. and Corkill, D.** January 1981. IEEE Trans. on Systems, Man, and Cybernetics. Vols. SMC-11, Number 1, pp. 81-96.

50. **Stepankova, Marik O., Trappl, R. and (eds.).** *Multiagent Systems and Applications: Selected tutorial papers from the Ninth ECCAI Advanced Course (ACAI 2001) and AgentLink's Third European Agent Systems Summer School (EASSS 2001).* Berlin : Springer-Verlag Lecture Notes in AI 2086, 2001. pp. 118-149.

51. **Moniz, Nelson.** *Optimal Mobile Node Placement In Self-Healing Hybrid Wireless Sensor Networks.* Computer Science, York University. Toronto, Ontario : s.n., January 2008. Master's Thesis.

52. *Mobile Communications Group.* [Online] http://www.cse.yorku.ca/~lan/seminar/.

53. *Models and solutions for radio irregularity in wireless sensor networks.* **Zhou, Gang, et al.** 2, New York, NY, USA : ACM, 2006, ACM Trans. Sen. Netw., Vol. 2, pp. 221-262. 1550-4859.

54. **Agrawal, D.P. and Zeng, Q. A.** *Introduction to Wireless and Mobile Systems.* s.l. : Brooks/Cole, August 2002.

55. *A new statistical approach to geographic variation analysis.* **Gabriel, K. and Sokal, R.** 3, September 1969, Systematic Zoology, Vol. 18, pp. 259-278.

56. *The relative neighborhood graph of a finite planar set.* **G.T., Toussaint.** 4, 1980, Pattern Recognition, Vol. 12, pp. 261-268.

57. *Sending messages to mobile users in disconnected ad-hoc wireless networks.* **Li, Qun and Rus, Daniela.** s.l. : ACM, August 2000. MobiCom '00.

58. *Data mules: Modeling a three-tier architecture for sparse sensor networks.* **Shah, R., et al.** s.l. : IEEE, 2003. Workshop on Sensor Network Protocols and Applications (SNPA).

59. *Using Predictable Observer Mobility for Power Efficient Design of Sensor Networks.* **Chakrabarti, A., Sabharwal, A. and Aazhang, B.** Palo Alto, CA : s.n., April 2003. The second International Workshop on Information Processing in Sensor Networks (IPSN).

60. *Intelligent fluid infrastructure for embedded networks.* **Kansal, A., et al.** 2004. 2nd international conference on Mobile systems, applications, and services (MobiSYS).

61. *A message ferrying approach for data delivery in sparse mobile ad hoc networks.* **Zhao, W., Ammar, M. and Zegura, E.** s.l. : ACM Press, 2004. 5th ACM International symposium on Mobile ad hoc networking and computing (MobiHoc). pp. 187-198.

62. *Energy efficient schemes for wireless sensor networks with multiple mobile base stations.* **Gandham, S., et al.** Dec 2003. IEEE GLOBECOM.

63. *Exploiting sink mobility for maximizing sensor networks lifetime.* **Wang, Z. M., et al.** 2005. 38th Hawaii International Conference on System Sciences (HICSS).

64. *Joint mobility and routing for lifetime elongonation in wireless sensor netowrks.* **J.Luo and Hubaux, J. P.** Mar 2005. 24th IEEE INFOCOM.

65. *Using mobile relays to prolong the lifetime of wireless sensor networks.* **Wang, W., Srinivasan, V. and Chua, K.** Cologne, Germany : ACM Press, NY, August 28 - September 02, 2005. 11th Annual international Conference on Mobile Computing and Networking (MobiCom '05). pp. 270-283.

66. *Solving geometric problems with the rotating calipers.* **Toussaint, Godfried T.** Athens, Greece : s.n., May 1983. Proceedings of IEEE MELECON'83. pp. A10.02/1-4.

67. **Pirzadeh, Hormoz.** *Computational Geometry with the Rotating Calipers.* McGill University. 1999. Master's thesis.

68. —. *Rotating Calipers Homepage.* [Online] November 26, 1999. http://cgm.cs.mcgill.ca/~orm/rotcal.htm.

69. *Scalable Networks.* [Online] 2008. http://www.snplc.com.

70. *Combating Routing Microholes in Hybrid Wireless Sensor Networks.* **Portnoy, Michael and Vlajic, Natalija.** Montreal, Canada : s.n., May 2007. The 7th IASTED International Conference on Wireless and Optical Communications.