Future Network & MobileSummit 2010 Conference Proceedings
Paul Cunningham and Miriam Cunningham (Eds)
IIMC International Information Management Corporation, 2010
ISBN: 978-1-905824-18-2

Routing with (*p-percent*) Partial Flooding for Opportunistic Networks

Mustafa ERDOGAN[1], Kadir GUNEL[2], Tugba KOC[2], H. Umit SOKUN[1], Tamer DAG[2]
[1]Kadir Has University, Electronics Engineering Dep., Cibali, Istanbul, Turkey
Email: {mustafa.erdogan, umit.sokun}@khas.edu.tr
[2]Kadir Has University, Computer Engineering Dep., Cibali, Istanbul, Turkey
Tel: +90 212 5336532/1422, Email: {kadir.gunel, tugba.koc, tamer.dag}@khas.edu.tr

Abstract: Opportunistic networks are one of the fast developing research areas in mobile communications. Under opportunistic networks, mobile nodes try to communicate with other nodes without any prior information and knowledge about the network topology. Furthermore, the network topologies are dynamic and can rapidly change. In addition, communication under opportunistic networks can be erratic, thus routes between a source node and a destination node sometimes might not exist. These issues would make traditonal routing approaches insufficient and unusable for opportunistic networks. In this paper, a new routing approach for opportunistic networks is proposed. The approach is called is *p%* partial flooding algorithm. With flooding, it is possible to reach a destination node with the minimum number of hops and minimum end-to-end delay. But, the major disadvantage of flooding is the excessive usage of the network resources. With *p%* partial flooding algorithm, the aim is to decrease the network traffic by randomly selecting neighbor nodes and routing traffic through them. This paper explains these two approaches (flooding and *p%* partial flooding) and compares their performance through various simulations. It is observed that *p%* partial flooding can result in the same benefits of flooding while decreasing the network traffic.

Keywords: Opportunistic Networks, Routing, Flooding, Selective Flooding.

1. Introduction

Wireless Sensor Networks (WSN) [1] is a rapidly developing technology where sensor nodes are positioned in a geographical area for monitoring the physical and/or environmental changes such as sound, vibration, pressure and movement. Each sensor device can share its knowledge autonomously with others. WSNs can be established quickly and they are much cheaper in contrast to traditional systems. The early attempts for constructing WSNs were mainly for military purposes. But later, it was realized that WSNs can be adapted to other application areas as well and they were used in various civilian projects. It is predicted that WSNs can be used effectively in areas such as wild life monitoring, structure health monitoring, and border enforcement; where all time tracking-monitoring is expensive, hard or impossible.

Opportunistic Networking [2] is a recently introduced concept that has started to attract the attention of many researchers. In this new network structure, the need of building an infrastructure is not necessary and the existing technologies might be appropriate. Research for opportunistic networks has focused on ad-hoc networks` application areas. Industry, disaster struck areas and undeveloped areas are amongst the examples of application areas.

For example, after a disaster such as an earthquake, a hurricane, a tsunami or a forest fire, due to the damage on the technological infrastructure, communication can be halted or disrupted. Opportunistic networks can be a solution for such issues. Thus, one of the main objectives of opportunistic networks is to provide best-effort communication.

The ratio of urbanization in some parts of the world is still very low and in such areas millions of people still have no network connection. With the help of opportunistic networks, at least a limited network connectivity and information acquisition can be provided for those people.

Various projects have been developed for opportunistic networks. Some of these projects in the literature are Zebranet [3] and Daknet [4]. In the Zebranet project, special collars are worn by zebras to maintain the data flow. In this project, the data flow is secure and useful but it is very expensive. The aim of the Daknet project is to connect the undeveloped villages and towns to the rest of the world. In this project, mobile base stations are built and mobile access points are attached to vehicles such as a bus, a motorcycle or a bicycle. With the help of the access points, short range wireless communication is provided. Data is collected (transmitted) by these access points from (to) undeveloped areas. When a mobile access point is in the range of a network connection, the collected data is transferred to (from) the network. By setting up kiosks in villages, short range wireless communication is set and the mobile access point is used in data switching by kiosks.

Thus, opportunistic networks aim to establish a communication between two nodes in a specific area where the source node sends packets to the destination node by using the help of other nodes in the same area. However, the routing characteristics of opportunistic networks are very different than the traditional network structured. The network might be disconnected. A path [5] between a source node and a destination node may or may not exist. A source node might not have information about the destination node's connectivity to the network. The topology of the network may change dynamically. Due to these issues traditional routing approaches are not suitable for opportunistic networks.

Dissemination based routing algorithms [6] have been considered to be appropriate for opportunistic networks. In these algorithms, the method of dissemination is flooding where data is spread throughout the network. The aim of dissemination is to increase the chances of delivery of the data to the destination node. However, the dissemination based routing algorithms suffer from excessive resource consumption. An increase in the number of nodes in the network increases the network traffic substantially. Some trade-off methods have been proposed [7] to overcome these issues:

- k-hop forwarding
- probabilistic forwarding
- limited time forwarding

In this paper, the *p%* partial flooding algorithm is proposed as a routing method for opportunistic networks. *p%* partial flooding algorithm might be used in dissemination or context based networks without infrastructure. Since the traditional flooding algorithms can exploit the network resources easily, *p%* partial flooding algorithm can be an alternative for such networks as the usage of network resources is reduced. The rest of the paper is organized as follows. Section 2 describes the flooding and *p%* partial flooding methods briefly. Section 3 describes the simulation model and the results obtained to compare the performance characteristics of the above algorithms. The paper ends with the conclusions made and brief description of the team's ongoing research as described in Section 4.

2. Flooding and *p%* Partial Flooding Algorithms

In this paper the flooding and *p%* partial flooding methods for opportunistic networks are described. Although flooding is a commonly used method for broadcasting information across a network, *p%* partial flooding method can substantially decrease the network traffic and might be used effectively in an opportunistic network. The following subsections describe these methods briefly.

2.1 Flooding Algorithm

Flooding is the main algorithm for broadcasting in a computer network. The same algorithm can also be used for opportunistic networks as well. As the network nodes do not have any information about the network topology and the network topology can change dynamically in such networks, to send data from a source node to a destination node or to access a destination, the data is transmitted to every neighboring node at the source node. Any node that receives the data repeats the same steps and by this way the data can be delivered to the destination node. One of the major problems that can be seen under flooding algorithm is the occurrence of loops that might cause excessive network traffic without even reaching the destination. In order to prevent loops, a network node transmits an incoming packet to the neighboring nodes except the one that the packet is coming from. In addition, if a network node receives the same packet again, it drops it. With flooding, it is possible to reach a destination node from a source node with minimum number of hops, with minimum delay but with maximum network traffic.

Figure 1 illustrates the typical steps at a network node, when the network node receives a packet with flooding.

```
if the packet has not been received for the first time
        drop the packet
        end
if the packet has been received for the first time
        if the destination is within range
                transmit the packet to the destination
                end
        if the destination is not within range
                find all the nodes within the current node's range
                transmit the packet to every node except the sender node
                end
        end
```

Figure 1: The operation of a network node under flooding method

2.2 p% Partial Flooding Algorithm

p% partial flooding algorithm is proposed as a way to reduce the network traffic that is observed under the flooding algorithm. The main difference between partial flooding and flooding algorithm is the selection of neighboring nodes. With partial flooding, a network node transmits the data not to all of its neighboring nodes but to a randomly selected subset. For the simulations conducted, 50% and 25% partial flooding algorithms have been used. If a network node has *N* neighbors, the number of randomly selected neighbors is equal to *N/2* and *N/4* respectively for 50% and 25% partial flooding.

Since partial flooding algorithm only selects some of the neighboring nodes and transmits data to them, the network traffic will decrease correspondingly. It might be possible to reach a destination node with more delay and more number of hops but as opportunistic networks are delay tolerant in nature, partial flooding might be easily adopted.

Figure 2 illustrates the typical steps at a network node, when the network node receives a packet with partial flooding.

```
if the packet has not been received for the first time
    drop the packet
    end
if the packet has been received for the first time
    if the destination is within range
        transmit the packet to the destination
        end
    if the destination is not within range
        find all the nodes within the current node's range
        randomly select p% of these nodes except the sender node
        transmit the packet to every selected node
        end
    end
```

Figure 2: The steps of a network node under p% partial flooding method

3. Simulation Model and Results

3.1 Simulation Model

The simulations for performance analysis of flooding and partial flooding algorithms are developed in C++ language by using Bloodshed Dev C++ IDE. For the simulations, an area of 1000m x 1000m has been selected and various numbers of nodes have been placed in this area. The location of the nodes, thus *(X, Y)* coordinates, has been picked up from uniform distribution. In addition to these nodes, two nodes marked as source node and destination node have been placed at *(0, 0)* and *(1000, 1000)* coordinate. The simulations are based on discrete-time event simulation [8] in which the state variables change only at a discrete set of points in time. There are two main events that trigger each other. One of them is transmission of a packet and the other one is receiving the packet. The events are managed through an event scheduler. An event scheduler is a dynamic list and the events are listed according to their event execution times.

A packet is sent from the sender node to the receiver node by using flooding and partial flooding methods. With the flooding algorithm, each node sends the packet to every other node in its range and with *p%* partial flooding algorithm; each node sends the packet to some of the nodes in its range. The simulations for *p%* partial flooding have been conducted for two cases when $p=50$ and $p=25$.

For the simulations, three parameters have been used to evaluate and compare the performance of the algorithms. These parameters are:

- the total delay between the source and destination nodes,
- the number of hops between the source and destination nodes,
- the total network traffic caused by the algorithms.

Simulation results have been obtained for these parameters by varying the number of nodes (*N=250, 500, 750 and 1000*) and node ranges. To calculate the total delay between the source node and the destination node, the processing delay at a network node is assumed to be equal to $t_{proc} = 0.05$ sec, and the transmission time of a packet from a node to a neighboring node is calculated based on the formula, $t_{prop} = d/C + p/R$.

In the above formula, d represents distance between two nodes, C represents the speed of light, p represents the size of the packet and R is the transmission rate. In all the simulations, packet sizes are constant and equal to 1000 bytes. Transmission rate in the network is 100 KB/sec, and all the nodes are identical to each other.

3.2 *Simulation Results*

3.2.1 The Impact on Network Traffic

Figure 3 illustrates the simulation results for the incurring network traffic as a result of flooding, 50% partial flooding and 25% partial flooding algorithms as node ranges vary. The simulations are conducted for $N = 250, 500, 750$ and 1000 nodes. An increase in the number of nodes available in the network increases the network traffic because of the increased number of packet forwarding under any flooding based routing algorithm. However, when the individual algorithms are compared, the network traffic in the partial flooding algorithm is considerably less than the network traffic in flooding. From the results, it can be observed that as p-value in partial flooding decreases, the network traffic also decreases proportionally. The traffic generated in 25% partial flooding is almost half of the traffic generated in 50% partial flooding which is also almost half of the traffic generated in flooding algorithm. This result is very important for partial flooding algorithm, because if data transmission is possible between the source node and destination node; thus the network is connected, partial flooding can provide communication with less consumption of network resources.

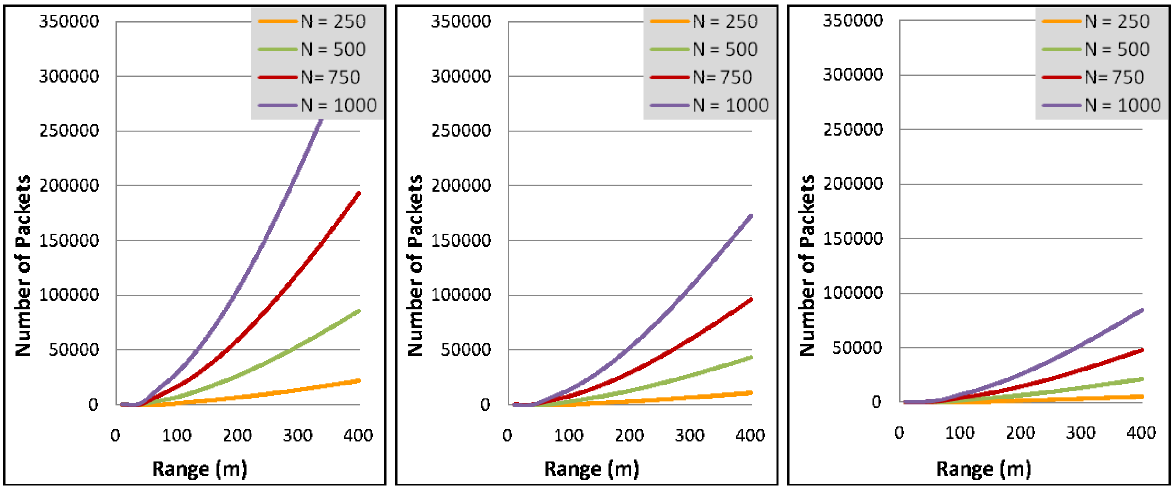

Figure 3: Network Traffic (Flooding, 50% and 25% Partial Flooding)

3.2.2 The Impact on Hop Count

Figure 4 illustrates the simulation results for the hop count to transmit a packet from the source node to the destination node under flooding, 50% partial flooding and 25% partial flooding algorithms as node ranges vary. The simulations are conducted for $N = 250, 500, 750$ and 1000 nodes. When the node ranges are small, the network becomes disconnected at some locations and in some cases communication cannot be provided between the source and destination nodes. Therefore, the packet cannot reach the destination node. In addition, for all three algorithms, the network connectivity is worse when the number of nodes is smaller.

As the node range increases, the network becomes connected. Increasing the node range, decreases the hop count for all three algorithms. However, when one compares the hop counts for a given range value when the network is connected, it can be concluded that the results are very similar to each other. The hop count for partial flooding is either the same or slightly larger than flooding. For example, for *N=500* with a range of *190 m*, the packet can be delivered to the destination in 8 hops for flooding and in 9 hops for 50% and 25% partial flooding. Thus, partial flooding is almost as good as flooding in terms of hop count.

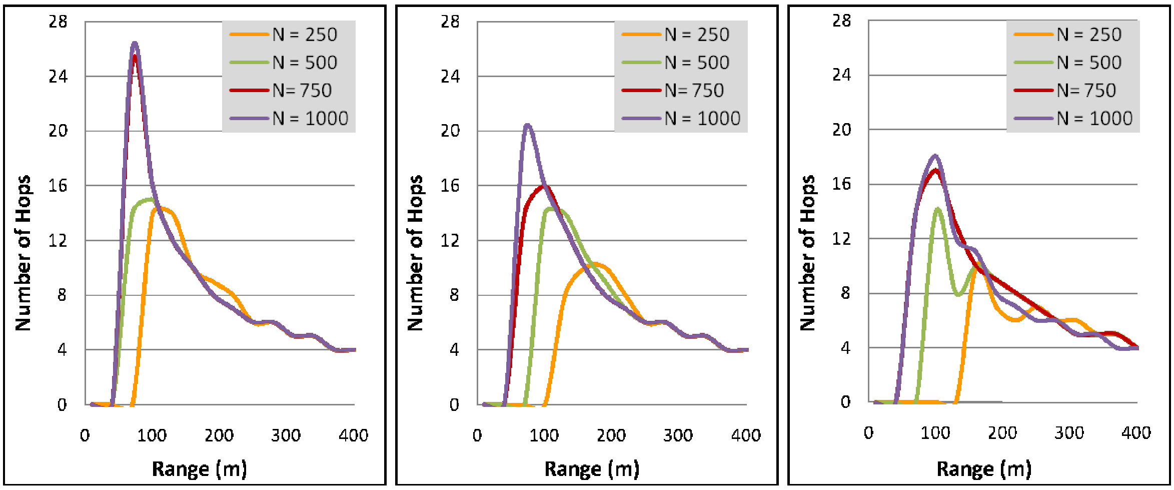

Figure 4: Hop Count (Flooding, 50% and 25% Partial Flooding)

3.2.3 *The Impact on End-to-End Delay*

Figure 5 illustrates the simulation results for the transmission time of the packet from the source node to the destination node as a result of flooding, 50% partial flooding and 25% partial flooding algorithms as node ranges vary. The simulations are conducted for *N = 250, 500, 750* and *1000* nodes. As the node range value increases, transmission time decreases because of the decrease in the number of hop count. The most important factor causing the delay is the number of packet transmission as the packet propagates toward its destination, thus when the hop count decreases, it will result in a decrease of total transmission time. Thus, the results for hop count and transmission time should be proportional. When the values for flooding, %50 partial flooding and %25 partial flooding are compared, it can be observed that they are very close to each other. Thus, partial flooding algorithm might be instead of flooding under opportunistic networks.

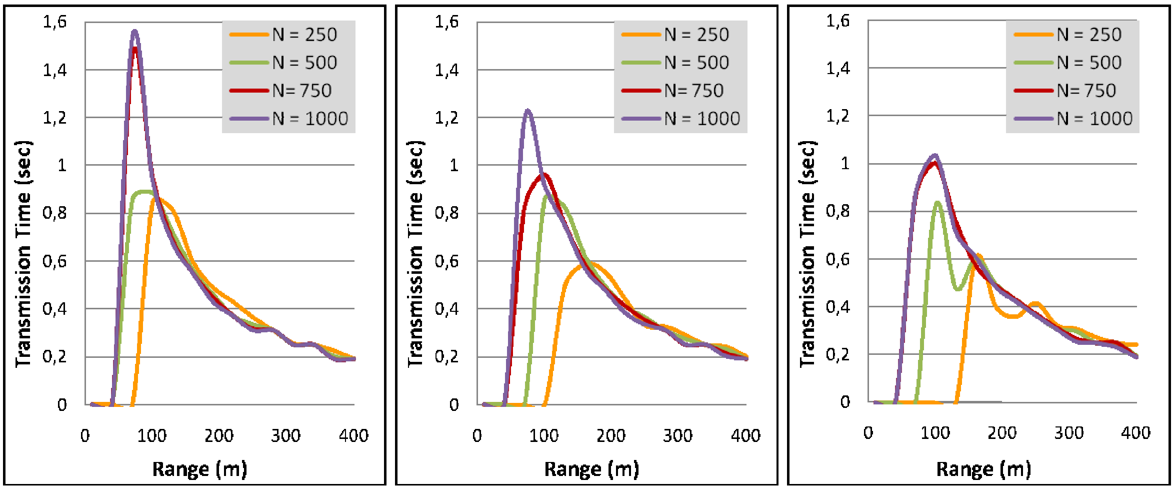

Figure 5: Total Transmission Time (Flooding, 50% and 25% Partial Flooding)

4. Conclusions

In this paper, the p% partial flooding algorithm for opportunistic networks is introduced and it is compared with the flooding algorithm. Under opportunistic networks, network nodes do not have any information about the network topology; furthermore the nodes might be simple sensors and they need to be energy efficient, they can be mobile. The network might sometimes be disconnected, thus a route from a source node to a destination node might not exist. Because of these reasons, the usability of traditional routing algorithms for opportunistic networks is extremely unlikely. Flooding algorithm might be very useful for reaching anyone connected to the network. By using flooding, it is possible to reach a destination node with minimum delay and minimum number of hops. However, flooding requires excessive usage of network resources, nodes that are not on the path from a source to a destination node will receive irrelevant packets and the network traffic caused by flooding algorithm will be extreme. Thus, p% partial flooding algorithm has been proposed to decrease the excessive usage of the network resources while still having the benefits of flooding. With p% partial flooding, only a randomly selected number of neighbors receive the incoming data. As a result of the simulations, it is observed that the network traffic in partial flooding method is far less than the network traffic in the flooding method. On the other hand, it is observed that the hop count and the end-to-end delay in %p partial flooding algorithm are very similar to the flooding method. Thus, p% partial flooding algorithm can be considered to be a more effective algorithm than flooding.

In this study, neighbor nodes are chosen randomly in the p% partial flooding algorithm. If the selection of neighbor nodes is fulfilled eclectically, it is expected that performance will be improved further. Our future work is concentrated on this aspect. In addition, the impact of different characteristics of network nodes will be studied, as in this study it is assumed that all the network nodes are identical.

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramania and E. Cayirci, "*Wireless Sensor Networks: A Survey*," IEEE Computer, Vol. 38, No. 4, pages 393-422, Mar. 2002.

[2] Luciana Pelusi, Andrea Passarella and Marco Conti, *"Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad Hoc Networks,"* IEEE Communications Magazine, Nov. 2006.

[3] P. Juang et al., *"Energy-Efficient Computing for Wildlife racking: Design Trade-Offs and Early Experiences with ZebraNet" ACM SIGPLAN Notices*, Vol. 37, pages 96-107, 2002.

[4] A. Pentland, R. Fletcher and A. Hasson, *"DakNet: Rethinking Connectivity in Developing Nations,"* IEEE Computer, Vol. 37, No. 1, pages 78-83, Jan. 2004.
[5] L. Chen, C. Chiou, and Y. Chen, "*An Evaluation of Routing Reliability in Non-Collaborative Opportunistic Networks*", International Conference on Advanced Information Networking and Applications, 2009.
[6] J. Widmer and J.Y. Le Boudec, *"Network Coding for Efficient Communication in Extreme Networks,"* Proceedings of ACM SIGCOMM 2005, Wksp. Delay Tolerant Networks, Philadelphia, PA, Aug. 22–26, 2005.
[7] *X. Zhang, G. Neglia, and J. Kurose, D Towsey. "Performance modeling of epidemic routing",* Computer Networks, Volume 51, Issue 10, July 2007, pages 2867-2891.
[8] Jerry Banks and John S. Carson, *"Discrete Event Simulation,"* Prentice Hall, 1984.