A FRAMEWORK FOR DISASTER MITIGATION NETWORKS

Gin-Xian Kok[1], Chee-Onn Chow[2], Hiroshi Ishii[3], and Keisuke Utsu[4]

[1]Electrical Department, Faculty of Engineering, University of Malaya, Kuala Lumpur, Malaysia, Tel: 60(012)6347880 [1], 60(03)79674457 [2], e-mail: xian_kgx@hotmail.com[1], cochow@um.edu.my[2]
[3]Department of Communication and Network Engineering, School of Information and Telecommunication Engineering, Tokai University, Tokyo, Japan, e-mail: ishii@ishiilab.net
[4]Graduate School of Science and Technology, Tokai University, Tokyo, Japan, e-mail: utsu@ishiilab.net

Received Date: April 26, 2013

Abstract

The existing communication infrastructure could be destroyed during a disaster and this could lead to the disruption of communication activities. Communication is extremely important during disasters; therefore, it is vital that a temporary network is quickly setup to restore communication activities. Temporary networks could be setup using programmable sensor nodes. In this paper, we present SafeNet, a framework for disaster mitigation networks using programmable sensor nodes. We discussed several aspects such as: i) placement of sensor nodes, ii) modes of operation, and iii) the required components.

Keywords: Ad hoc networks, Disaster mitigation, Network coding, Reducing broadcast redundancy, Sensor nodes

Introduction

Wireless sensor networks (WSNs) are communication networks formed using sensor nodes. They are generally designed for simple tasks such as monitoring of physical phenomena (temperature, intensity of light, barometric pressure, etc). Sensor nodes are battery-powered, small in size, light weight, and are equipped with sensors/are able to interface with sensor boards. Their programmability makes them very flexible in their applications. Hence, they can also be used for other purposes such as setting up a network for disaster mitigation.

A disaster is a sudden and unexpected event that causes great damage to an existing system. Disasters can be natural or man-made. Examples of natural disasters are floods, earthquakes, storms, and tsunamis while examples of man-made disasters are road accidents, fire, and building collapses. After a disaster, the existing communication infrastructure could be destroyed. As a result, intra and inter network communications are prevented, and victims are unable to communicate with each other and also with the outside world. This makes critical activities such as search-and-rescue (SAR) difficult if not impossible. Therefore, one of the main priorities in disaster response is to quickly set up a temporary network to allow communication activities to resume.

In this paper, we introduce a framework for disaster mitigation networks called SafeNet. The organization of this paper is as follows. In Section II, we discuss about various aspects that are beneficial for disaster mitigation networks. In particular, in Section II.C.1, we describe why ad hoc routing is crucial for disaster mitigation networks and then propose to incorporate network coding into a conventional ad hoc routing protocol to improve its performance. Beside unicast routing, broadcasting is also vital in a communication network. In Section II.C.2, we propose to adopt an efficient multi-hop broadcast scheme called Improved Partial Dominant Pruning (IPDP) [1] into SafeNet. In Section II.C.3, we propose a

service discovery scheme for service providing nodes to allow nodes to acquire the addresses of the service providing nodes in a network. Finally, any network will require some amount of maintenance. Although SafeNet is designed to minimize the amount of maintenance needed, due to physical placement restrictions/constrains in an actual deployment, inevitably some of the nodes will have to run on batteries and the required amount of maintenance increases. Centralized maintenance will be helpful in this case to help reduce the required maintenance and we discussed about this in Section II.C.4. In Section III, we conducted some simple tests to evaluate the performance of the proposed unicast routing protocol. Finally, we conclude the paper in Section IV.

SafeNet

Placement of Nodes

One of the issues that require investigation in disaster mitigation networks is the placement of sensor nodes. Regarding this issue, we propose that sensor nodes be placed at locations where there is an electrical power outlet. By placing nodes close to an electrical power source, SafeNet has the following advantages:

- requires minimal maintenance as nodes are powered by electricity rather than batteries
- can operate at maximum capacity as energy supply of nodes are not constrained
- Electrical power outlets are easily found in every part of a building. In outdoor areas, nodes could be placed on objects such as street lights, sign boards, and traffic lights.

Different Modes of Operation

Prevention is better than cure. To be effective, it is important that disaster mitigation networks are deployed prior to an actual disaster. However, deploying a network for the sole purpose of disaster mitigation during peaceful times might be hard to be justified because a disaster might not even occur. To help justify deployment cost, we propose that the network should be used and not left idle during peaceful times. Due to its dual purpose (disaster mitigation and normal usage during peaceful times), we propose that the network could operate in two different modes.

During peaceful times, SafeNet should operate in the NORMAL mode. As opposed to sitting idle, SafeNet could be used for monitoring of critical events (for example the water level of a river, and the seismic activity level), and also as a regular wireless communication network where it could be used for local inter-network communication, or to offload the load of existing communication networks. In peaceful times, nodes are mostly powered by an electrical source; therefore SafeNet can operate at the maximum capacity as energy supply is not a concern.

When a disaster strikes, the electrical supply could be cut off and nodes begin to operate on batteries (the batteries are charged during peaceful times). In this situation, SafeNet should switch to the CRITICAL mode. In this mode, the main priority of the network is to support critical activities such as search-and-rescue (SAR) and dissemination of important messages. In this mode, the usage of SafeNet should be regulated/controlled to allow only the most important tasks to conserve valuable remaining energy (non-critical activities during this time such as gaming and multimedia are disallowed/blocked).

Components

To support its targeted applications, the following components are required in SafeNet:

- Unicast routing protocol
- Broadcast routing protocol
- Service discovery scheme
- Centralized maintenance scheme

Unicast Routing Protocol

Unicast communication is the communication between a pair of nodes in a network. In WSNs, a pair of nodes that wishes to communicate which each other could be out of range of each other due to physical limitations, i.e. the nodes are located far from each other, and they have limited radio transmission range. Hence, unless the deployment area is small, multi-hop communication is almost certainly required.

It is also highly unlikely that nodes are placed in an orderly/structured manner during/after a disaster. The nodes are more likely to be scattered across the affected area. They could also be mobile (for example if they are mounted onto vehicles, carried by humans or animals, and etc.). Thus, it is vital that one or more paths could be found at any time between the two end-to-end nodes wishing to communicate. For this purpose, a reactive ad hoc routing protocol could be used. Examples of reactive routing protocols are Ad-hoc On-demand Distance Vector (AODV) [2] and Dynamic Source Routing (DSR) [3].

In this paper, we suggest using an extended version of AODV as the routing protocol in SafeNet. AODV is an on-demand ad hoc routing protocol. This means that in AODV, routes are formed only when they are used/needed and this reduces communication overhead. In contrast, pro-active ad hoc routing protocols like Destination-Sequenced Distance Vector (DSDV) [4] maintain routes to every destination at every node, even if the routes are not used/needed. This causes unnecessary bandwidth consumption and increased energy consumption which is bad for a network deployed for disaster mitigation because increased energy consumption causes batteries in nodes to run dry which could in turn affect the network in various ways such as network partitioning, decreased network lifetime, and losing functionality.

We now briefly explain how AODV works. In AODV, when a node has a packet to send, it looks up its routing table for a route entry to the packet's destination. If a valid route entry is found, the node sends the packet to the next-hop specified in the route entry. Otherwise, the node initiates a route discovery process. In a route discovery process, Route Request (RREQ) packets are flooded throughout the network to search for the destination. During this process, route entries to the RREQ source are formed at intermediate and the destination nodes. When the destination is found, it replies with a Route Reply (RREP) packet which will travel in the reverse direction of that traversed by the received RREQ packet. During this process, the intermediate and RREQ source nodes add or update their route entry to the RREQ destination. When the RREP packet reaches the RREQ source, a two-way route is formed between the source and destination and communication can begin. Due to congestion and nodes mobility, routes can fail; AODV handles this through the use of Route Error (RERR) packets.

Sensor nodes are usually battery-powered devices. Hence, they have energy, computational, and bandwidth constrains. During/after a disaster, a lot of communication activities will occur in the form of distress calls which could cause a network to suffer from congestion collapse. The multi-hop property of communications in ad hoc networks further

adds to the problem because nodes share a common channel, and a packet might require multiple transmissions/hops to reach its destination.

In existing communication networks, data are transmitted from the source to the destination in a store-and-forward manner. In this method, a packet is stored at a node and a copy of it is forwarded to the next node via an output link. Recently, the concept of network coding was introduced. The advantage of network coding over the conventional store-and-forward scheme was first demonstrated in [5], thus refuting the traditional belief that there is no need for data processing at the intermediate nodes except for data replication. The advantage of network coding includes higher throughput and security. The butterfly network (Figure 1) is most commonly used to demonstrate the advantage of network coding in throughput.

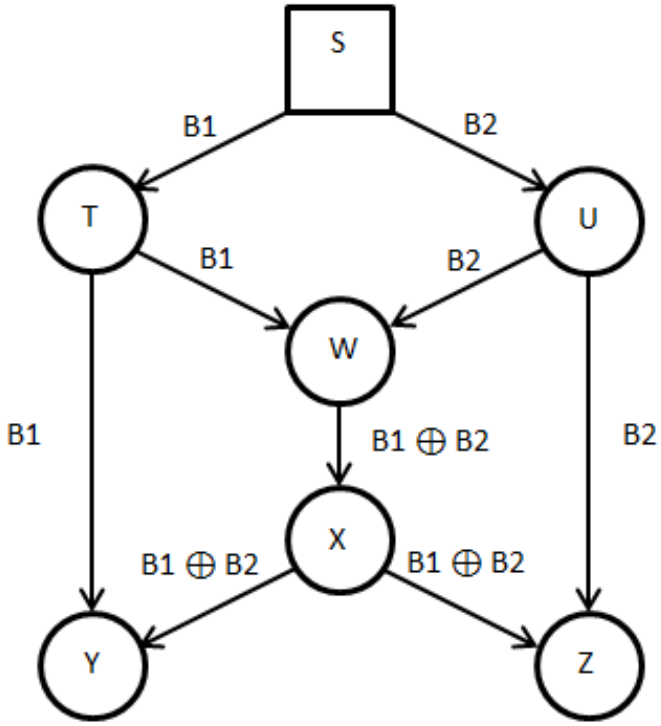


Figure 1. The butterfly network

With the store-and-forward scheme, five transmissions are required for nodes Y and Z to receive packets B1 and B2. With simple processing at node W, packets B1 and B2 can be combined together (using simple XOR operation) before transmitted. This simple processing reduces the number of transmissions required. With network coding, to transfer the same information (packets B1 and B2) to nodes Y and Z, only four transmissions are required. This constitutes to a $(1/4 - 1/5)/(1/5) = 25\%$ gain in throughput.

We propose to incorporate network coding into AODV. We name the resulted routing protocol AODV-Network Coding (AODV-NC). The motivation for this is to make use of in-network processing/network coding to improve network performance. The basic ad hoc routing algorithm is similar to AODV.

Network coding is actually how a node encodes several packets together so that the source packets can be retrieved (correctly decoded) at the destinations. One of the easiest network coding is X-OR of packets from two different flows that flow in opposite directions

in a three nodes topology as shown in Figure 2.

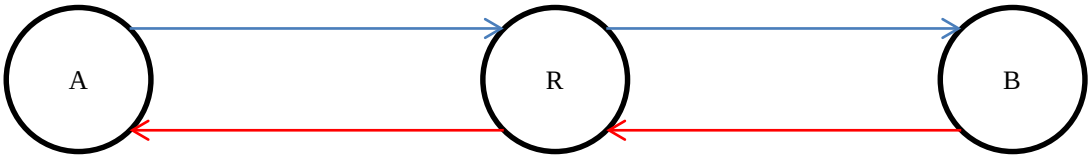


Figure 2. Two flows of opposite directions in a two hops chain

We call such a topology as the chain topology. Suppose node A wants to send a packet to node B, and node B wants to send a packet to node A. Without network coding, the above scenario would require four transmissions (so that node B receives node A's packet, and node A receives node B's packet). However, when network coding is employed, one transmission could be saved, by making node R encode packet A and packet B together and then transmit the encoded packet to node A and node B in one transmission. Assuming node A temporary buffers the packet that it sent to node B, it can retrieve the packet sent to it by node B by decoding the encoded packet with the packet that it sent to node A. Similarly, node B would be able to retrieve the packet intended to it by node A. With potentially more than one flow (one flow is a unique source-destination pair) in an ad hoc network, the above network is easily seen as part of a larger network. We model a network as a directed graph, G(N, E), where N is the set of nodes in the network, and E is the set of edge/links in the network. In Fig. 3, we show the occurrence of such a sub-network within a larger network. In the figure, the set of source nodes, S = {A, F} and the set of receiver nodes, R = {C, E} where S and R are subsets of N. Instead of finding a network code for the whole network, G, we find a network code for the sub-network, G'. Since nodes E and C forward packets for other nodes, and these packets should be retrieved by receiver nodes C and E, we can view node C as the source for packets from node A that needs to received correctly at node E and conversely, we can view node E as the source for packets from node F that needs to be correctly received at node C. Hence, we say we have a sub-network, G' within a larger network, G. The sub-network, G' is circled in red and one of the possible codes for this network is the XOR code as depicted in Figure 2 which is described previously.

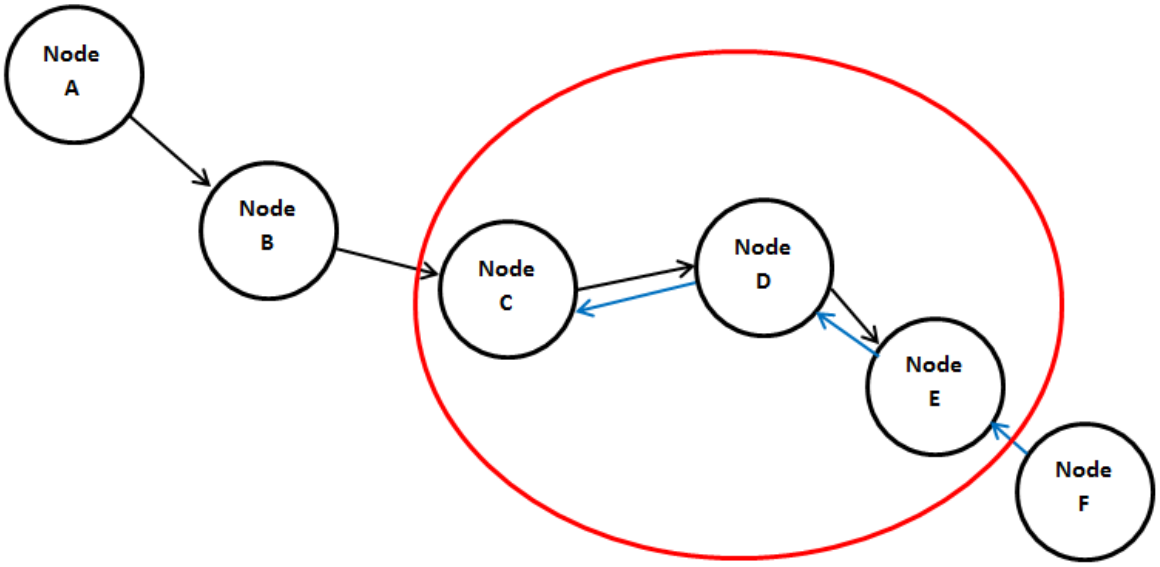


Figure 3. A sub-network within a larger network

Besides the chain topology, there are also other topologies where network coding can occur. If nodes are allowed to operate in promiscuous mode (i.e. they can overhear transmissions of other nodes that are not intended for them), then network coding can also occur in the cross (X) topology (Figure 4).

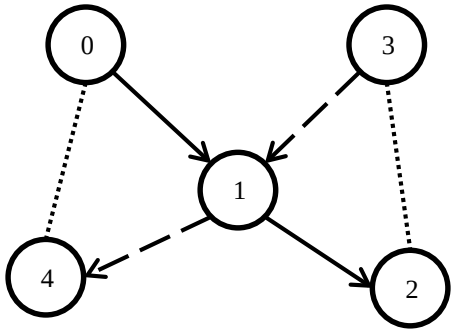


Figure 4. Cross (X) topology

Certain conditions have to be satisfied before a pair of packets can be encoded together at a coding node, and we call such conditions as the coding conditions. At a potential coding node, the coding conditions for encoding a received native packet with a native packet in its output queue is as follows [6]:

$$|UPSTR\_F1 \cap DOWNSTR\_F2\_AND\_NB| > 0 \text{ and } |UPSTR\_F2 \cap DOWNSTR\_F1\_AND\_NB| > 0$$

or

$$|UPSTR\_F1\_AND\_NB \cap DOWNSTR\_F2| > 0 \text{ and } |UPSTR\_F2\_AND\_NB \cap DOWNSTR\_F1| > 0$$

where:

- $UPSTR\_F1$ is the set of upstream nodes of the considered coding node on the path of the received packet.
- $UPSTR\_F2$ is the set of upstream nodes of the considered coding node on the path of the packet in the interface queue.
- $DOWNSTR\_F1\_AND\_NB$ is the set of downstream nodes of the considered coding node on the path of the received packet and their neighbours.
- $DOWNSTR\_F2\_AND\_NB$ is the set of downstream nodes of the considered coding node on the path of the packet in the interface queue and their neighbours.
- $UPSTR\_F1\_AND\_NB$ is the set of upstream nodes of the considered coding node on the path of the received packet and their neighbours.
- $UPSTR\_F2\_AND\_NB$ is the set of upstream nodes of the path of the considered coding node on the packet in the interface queue and their neighbours.
- $DOWNSTR\_F1$ is the set of downstream nodes of the considered coding node on the path of the received packet.
- $DOWNSTR\_F2$ is the set of downstream nodes of the considered coding node on the path of the packet in the interface queue.

We now provide an example to show how the above set of coding conditions is used. We use the topology shown in Figure 4 which consist of five nodes {0, 1, 2, 3, 4}, and flows F1: (0-1-2) and F2: (3-1-4). Node 1 is the potential coding node, node 4 can overhear node 0, and node 2 can overhear node 3. At node 1, $UPSTR\_F1 = \{0\}$, $DOWNSTR\_F2\_AND\_NB = \{0, 4\}$, $UPSTR\_F2 = \{3\}$ and $DOWNSTR\_F1\_AND\_NB = \{2, 3\}$. Then, $|UPSTR\_F1 \cap DOWNSTR\_F2\_AND\_NB| = |\{0\}| = 1$ and $|UPSTR\_F2 \cap DOWNSTR\_F1\_AND\_NB| = |\{3\}| = 1$. Since the coding conditions are satisfied, node 1 can encode the two packets together.

In a real network, packets do arrive at a node at different times. We refer again to Figure 4. There are problems if node 1 is to encode every packet from F1 with every packet from F2 as the rates of the two flows could be different. Node 1 would be required to delay sending of some packets to wait for other packets to be encoded with them (from other flows) to arrive. This leads to higher packet delay. Instead, we make nodes encode packets opportunistically. The flowchart in Figure 5 shows the procedure taken by a node when it receives a packet.

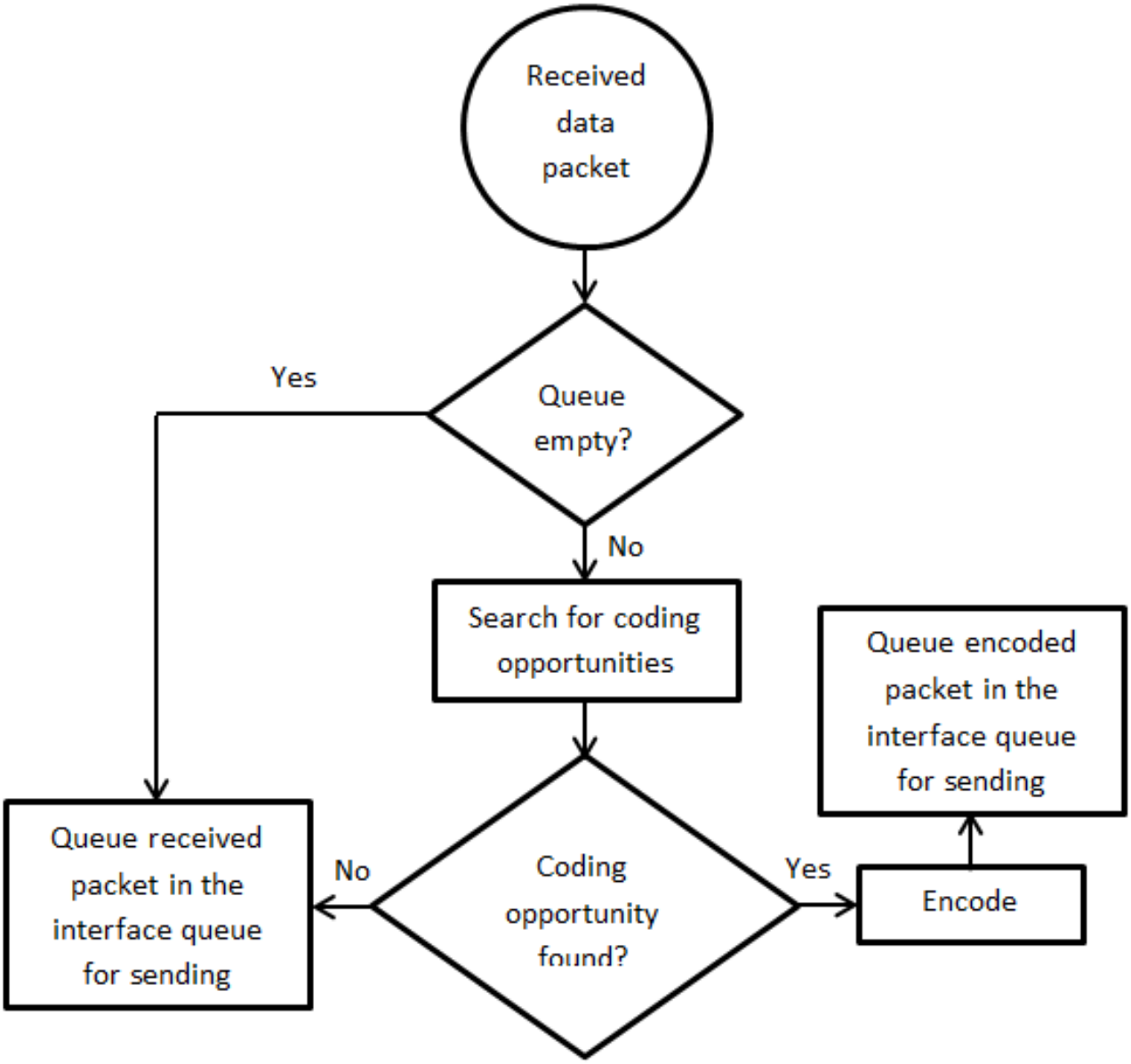


Figure 5. Flow diagram to describe how a data packet is processed when it is received by a node

Broadcast Routing Protocol

Broadcasting refers to the operation of sending a packet to all other nodes in a network. However, traditional broadcasting (local broadcast) can only send a packet to all other nodes within one-hop from the source/transmitting node. Hence, local broadcast is unsuitable for ad hoc networks/WSNs which are generally multi-hop in nature.

Blind flooding is a popular multi-hop broadcast scheme for ad hoc networks. In blind flooding, a node rebroadcasts/forwards a packet when it is heard for the first time, otherwise the packet is discarded. However, blind flooding causes a lot of redundant transmissions. Consider the network shown in Figure 6 in which node 0 wants to broadcast a packet so that the packet reaches all the other nodes. With blind flooding, there are seven transmissions (one by each node). However, it can easily be observed that the transmissions of node 3, node 4, node 5, and node 6 are not actually required (four redundant transmissions).

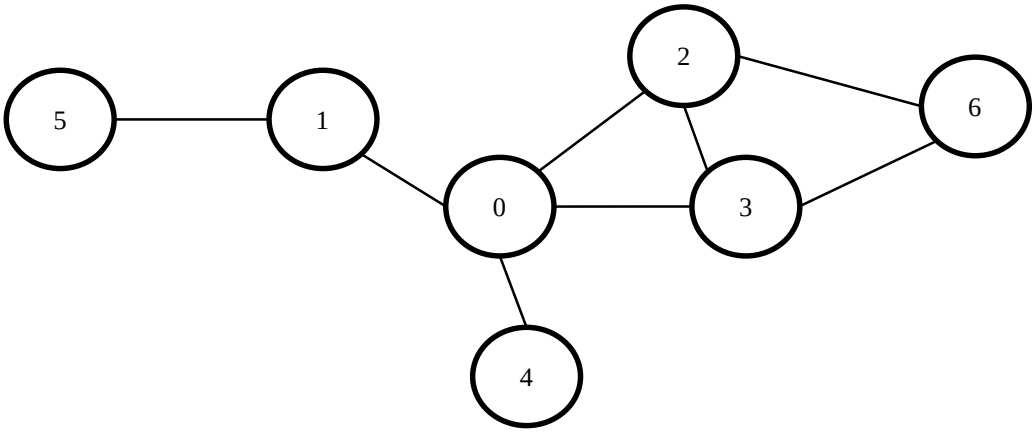


Figure 6. An example network to demonstrate transmission redundancy in blind flooding

By making nodes utilize local neighbourhood connectivity information, redundant transmissions could be reduced. The Partial Dominant Pruning (PDP) [7] broadcast scheme uses two-hop neighbour connectivity information to reduce redundant transmissions by making nodes select only a subset of their one-hop neighbours as forwarding nodes. The Improved Partial Dominant Pruning (IPDP) [1] further improves upon PDP by making forwarding nodes of the same previous hop coordinate among themselves implicitly/ indirectly. As AODV already have the HELLO neighbour discovery process, PDP can be easily incorporated into our framework/implementation. For the performance analysis of IPDP compared to blind flooding or PDP, we refer the interested reader to [1].

Service Discovery Scheme

Conventional unicast ad hoc routing protocols usually require every node to have a unique network address and a node is required to know the network address of the destination node before it can communicate with it. However, in a disaster mitigation network, a node might not know the address of the destination node that it wishes to contact. As a result, nodes providing services such as emergency relief and search-and-rescue (SAR) could not be contacted. Thus, it is vital that a service discovery scheme is used in a network deployed for disaster mitigation so that nodes know the services available/provided in the network and the network addresses of the service providing nodes.

We propose that service providing nodes broadcast advertisement packets of their services together with their network addresses periodically using the adopted broadcast scheme that is proposed in Section II.C.2. With periodic advertisement broadcasts, nodes can discover the services available in a network and the addresses of the service providing nodes. A node can then setup a unicast route to a service providing node through the use of conventional ad hoc routing protocols such as AODV-NC that is proposed in Section II.C.1. Consider the example network in Figure 7 in which there are six nodes. Nodes 1 and 2 are service providing nodes. Node 1 provides Internet connectivity while node 2 provides video streams. Both of these nodes periodically broadcast advertisement packets of the services that they provide together with their network addresses. Suppose node 5 decides that it requires Internet connectivity, it can setup a unicast connection to node 1.

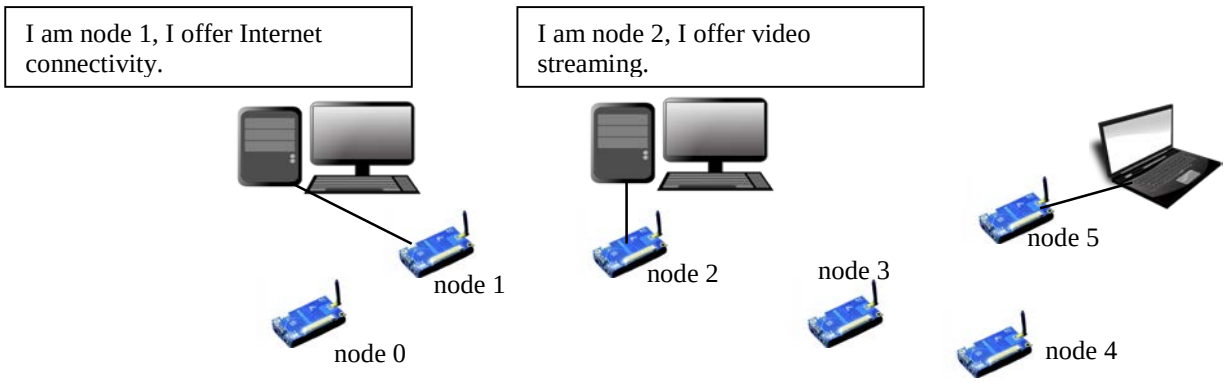


Figure 7. Service providing nodes periodically broadcast their unique network address together with their services

Centralized Maintenance Scheme

In SafeNet, nodes are placed close to an electrical power outlet. This reduces the need for maintenance such as replacing the batteries of sensor nodes. However, it might not always be possible to find an electrical power outlet in every part of the deployed area. Inevitably, some of the nodes might be required to run on batteries. The use of such nodes results to higher network maintenance.

To reduce network maintenance, we propose that during the NORMAL mode of operation in SafeNet, nodes send reports regarding their conditions periodically to a central node (the base station). Examples of information that could be included in such reports are battery voltage, geographical location, and error logs. With this information, a human network moderator can monitor the status of the nodes through a central point (the base station) and quickly pinpoint the nodes that require maintenance (nodes low on battery, dead nodes, etc).

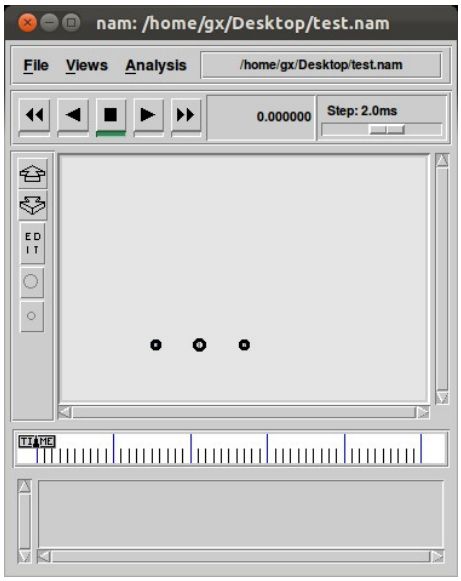
Results and Discussion

We used network simulator version 2 (ns-2) [8] which is a popular open source discrete event network simulator to evaluate the performance of the proposed AODV-NC routing protocol. We compared it against the original AODV [2] routing protocol. The MAC protocol used was IEEE 802.11g. The simulated sensing field is of dimension 1000 m by 1000 m.

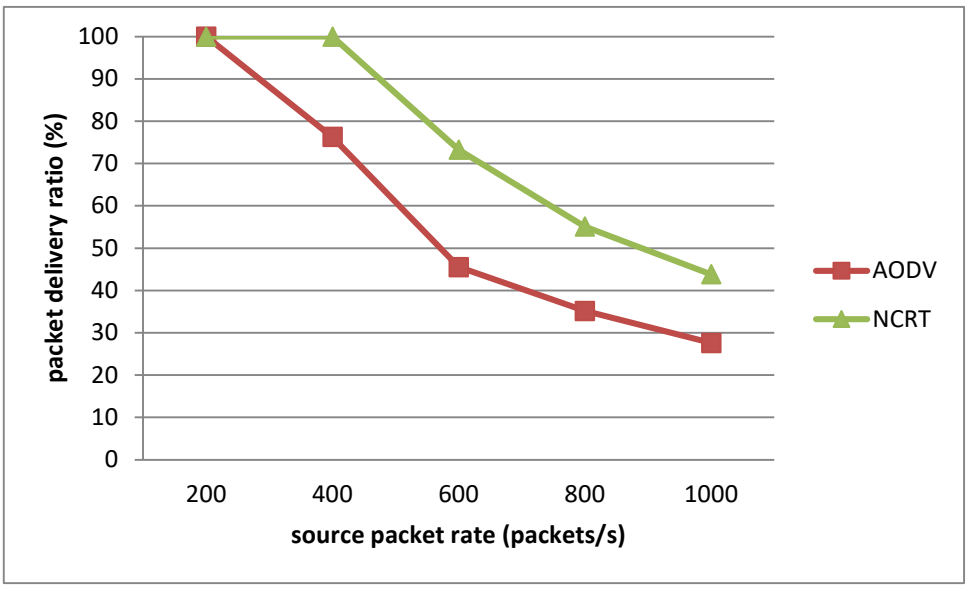
We evaluate the protocols under different artificial scenarios. In the first scenario, three nodes are placed in a straight line with an inter-node distance of 200 m. Two flows were set up in the network, one flow from the head node to the tail node, while the other flow flows in the reverse direction. As packets from the two flows have to travel two hops from source to destination, we call this scenario the “two-hop chain” scenario. AODV-NC and AODV are on-demand routing protocols. To make sure routes for the flows could be formed, the sources of the two flows first send a data packet to trigger the route discovery process. The sources start to send data packets at approximate 5 s simulation time. By then, the routes for both flows should have been set up. Simulation was terminated at 65 s giving sources 60 s to send data packets. The rate at which data packets are sent were varied from 200 packets/s to 1000 packets/s in increments of 200 packets/s. Figure 8a shows the topology of the network for the two-hop chain scenario. Figure 8b shows the packet delivery ratios obtained. It can be seen that AODV-NC obtained a higher packet delivery ratio at all levels of network load. Figure 8c shows the average packet delays. The figure shows that AODV-NC produces lower packet delays than AODV. Fig. 8d shows the average network throughputs. From the figure, we observed that AODV-NC produces higher average network throughputs than AODV.

The next scenario in our test is the “three-hop chain” scenario. This scenario is largely similar to the “two-hop chain” scenario discussed previously except that there is one more node in the network and the flows are longer by one hop. Figure 9a shows the topology of the network while Figures 9b-9d shows the results obtained for the three-hop chain topology. AODV-NC produces slightly higher packet delivery ratio and better average packet delays. However, compared to the two-hop chain topology, the gain realized is not as high.

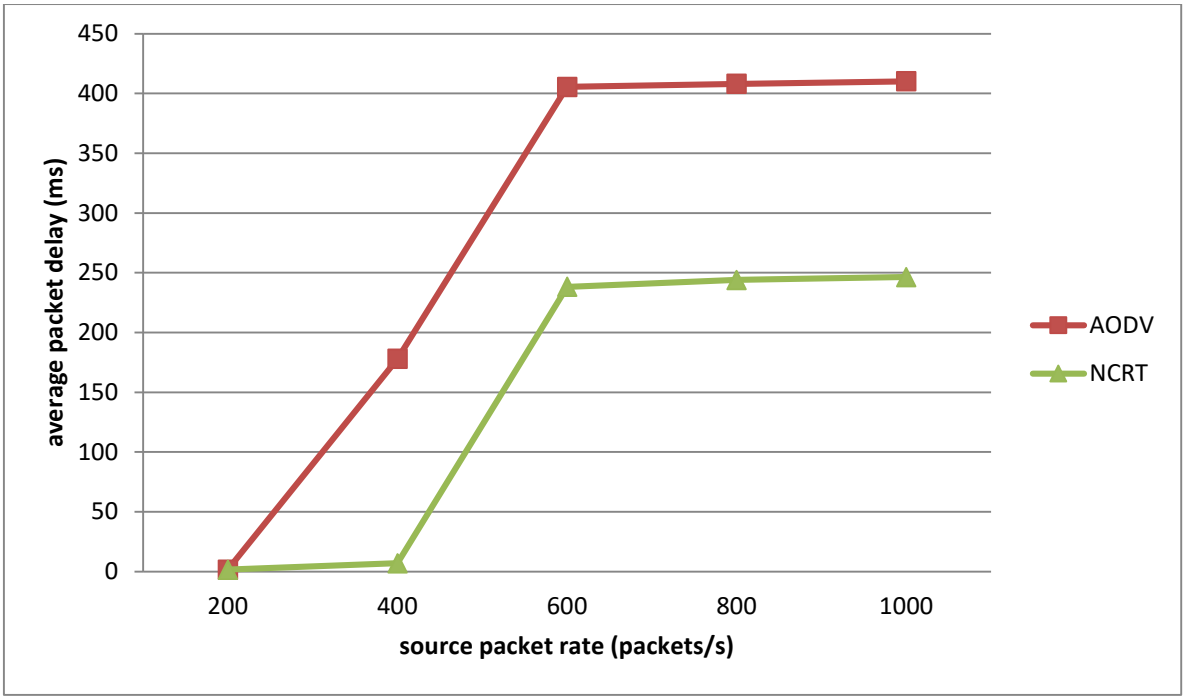
The final scenario in our test is the “cross topology” scenario. In this scenario, five nodes are arranged in such a way that they form a cross. Figure 10a shows the topology while Figures 10b-10d shows the result obtained for this topology. We found that the shape of the graphs is rather similar to the two-hop chain scenario. Indeed, both the two-hop chain topology and the cross topology are quite similar. The difference between them is that destination nodes use overheard packets to decode encoded packets in the cross topology. In contrast, destination nodes use their own buffered packets to decode encoded packets in the two-hop chain topology.



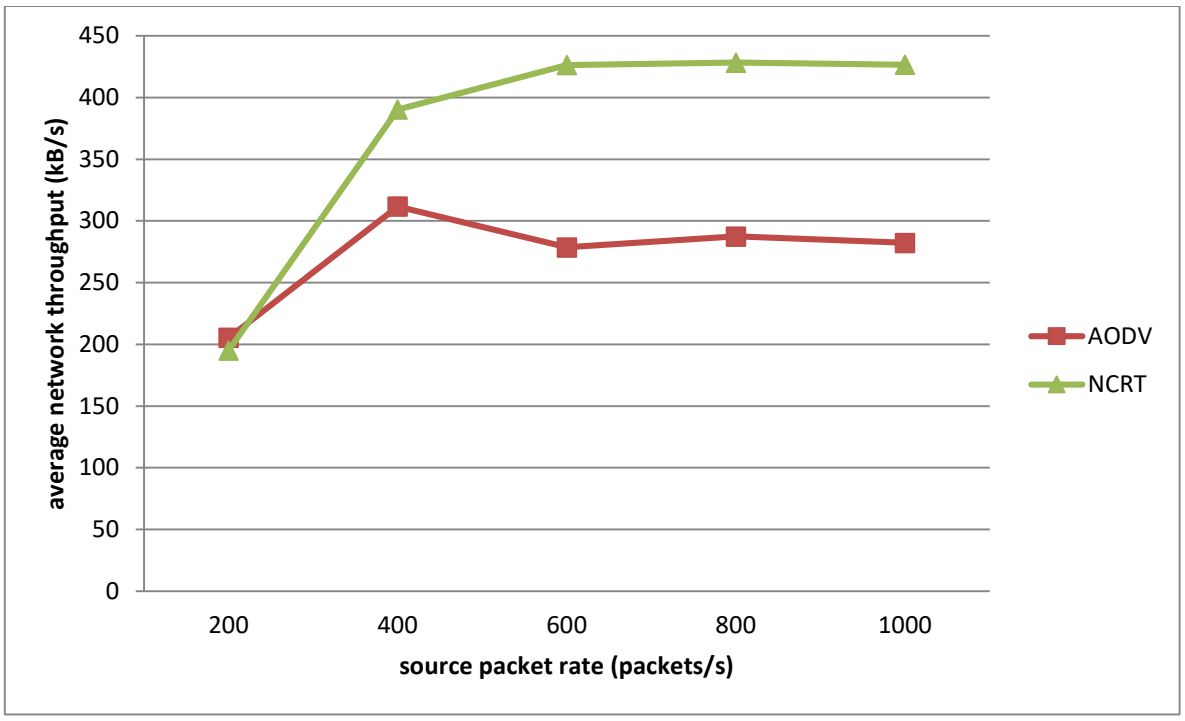
(a)



(b)

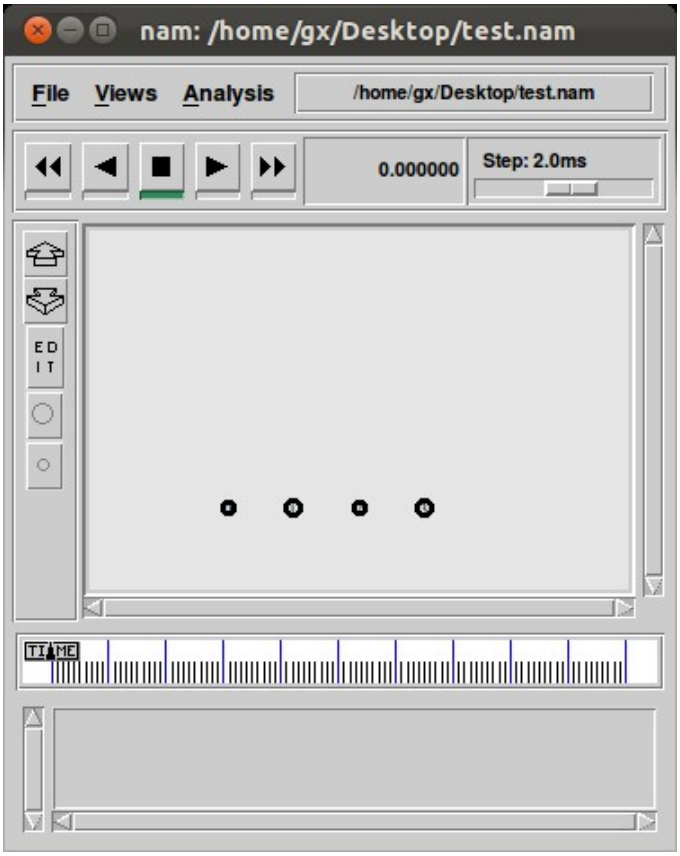


(c)

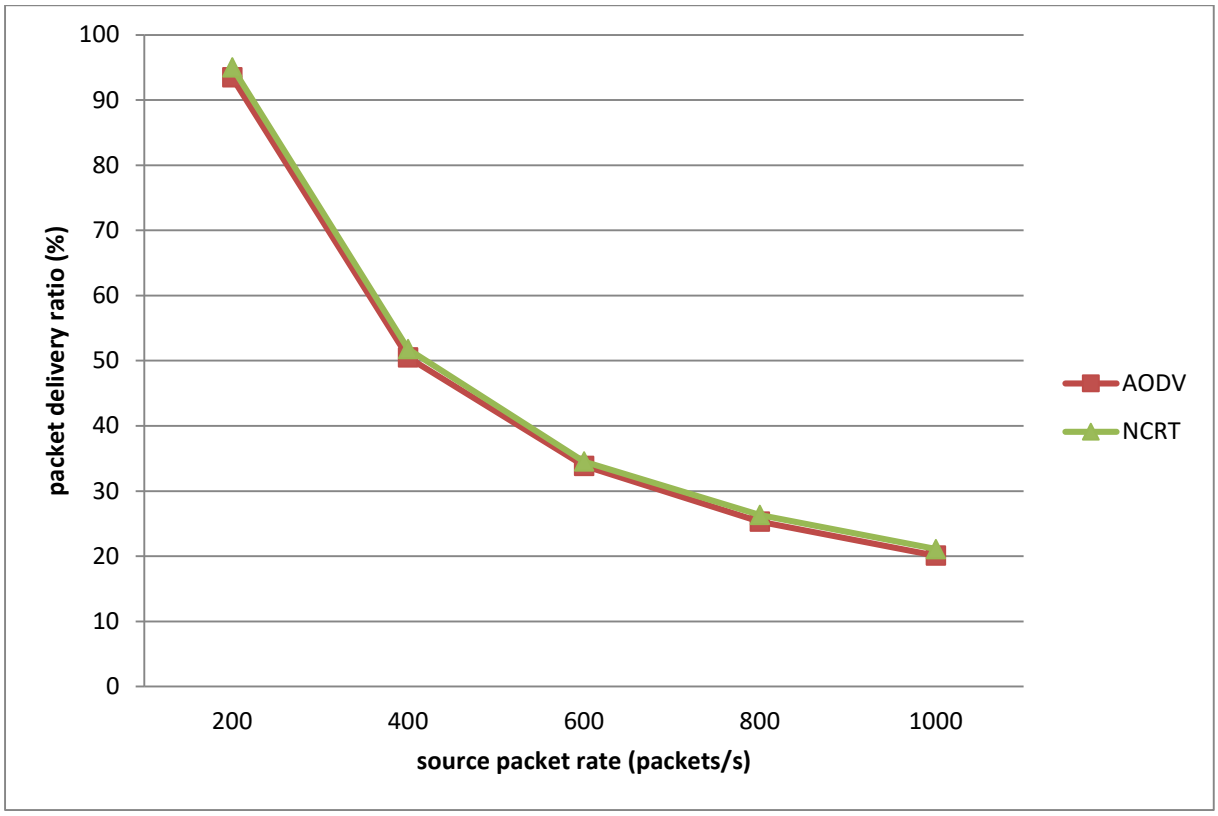


(d)

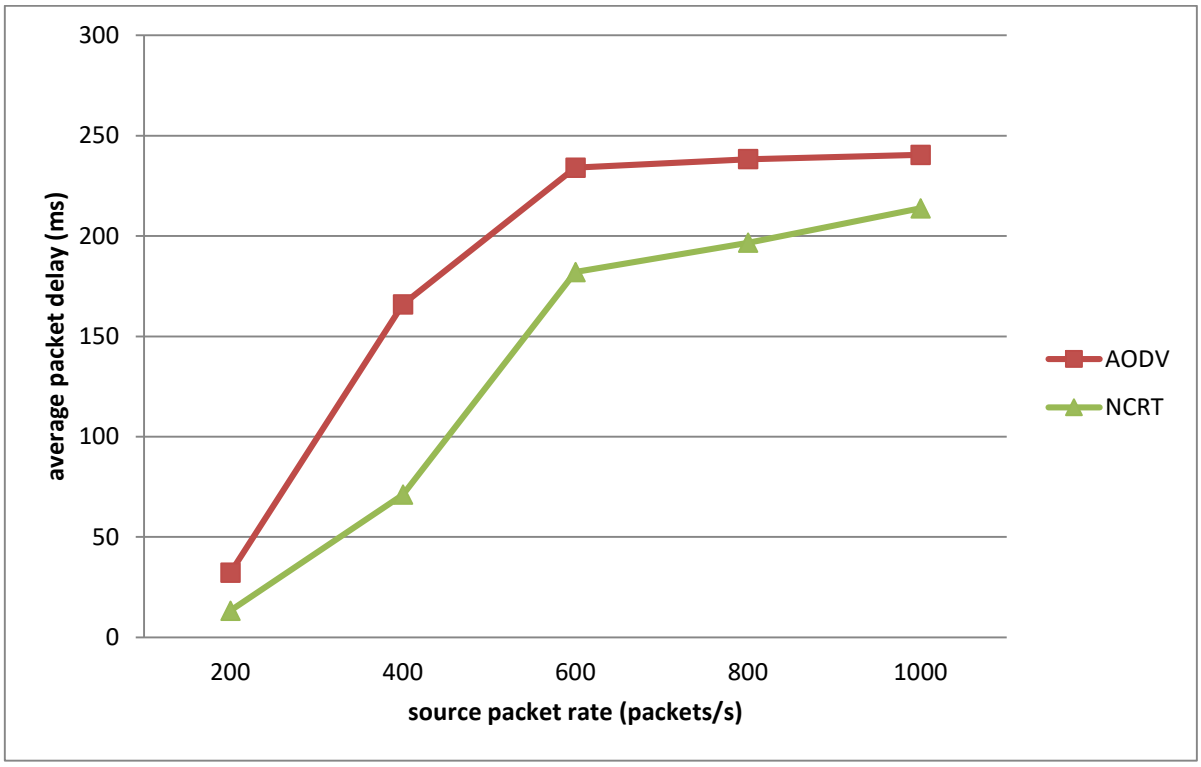
Figure 8. Two-hop chain: (a) topology (b) packet delivery ratio (c) average packet delay (d) average network throughput


nam: /home/gx/Desktop/test.nam
File
Views
Analysis
/home/gx/Desktop/test.nam
0.000000
Step: 2.0ms
ED
IT
TIME

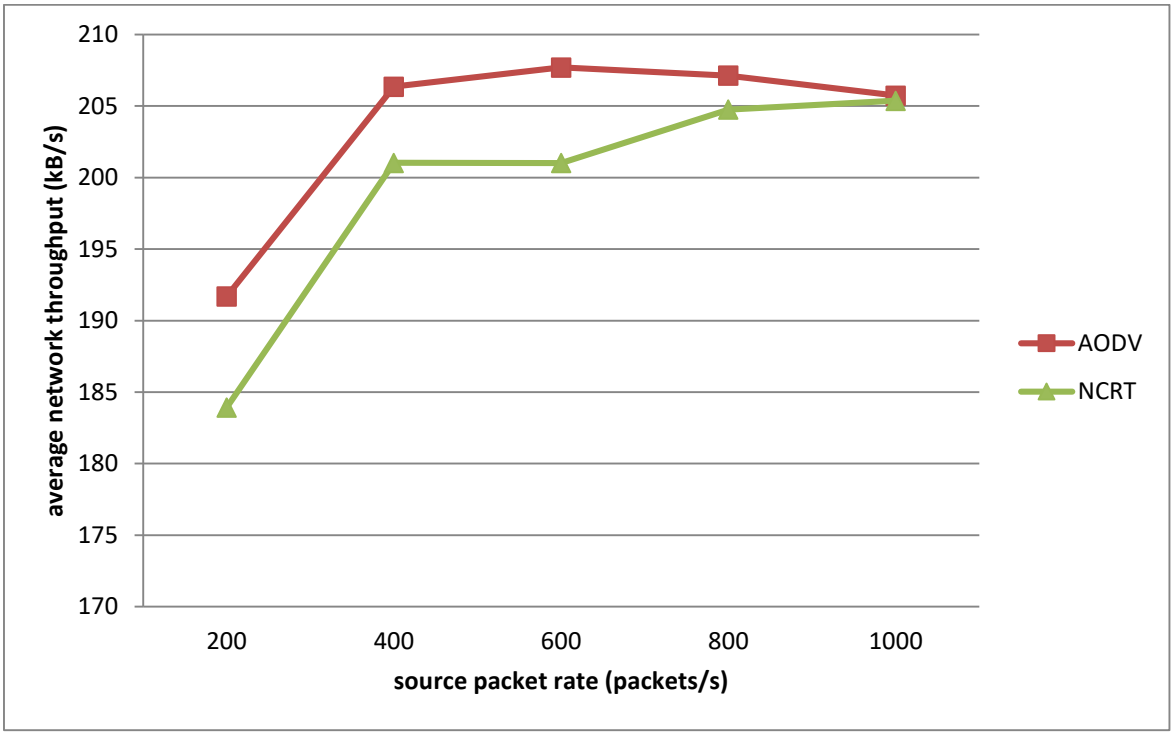
(a)


100
90
80
70
60
50
40
30
20
10
0
packet delivery ratio (%)
200
400
600
800
1000
source packet rate (packets/s)
AODV
NCRT

(b)

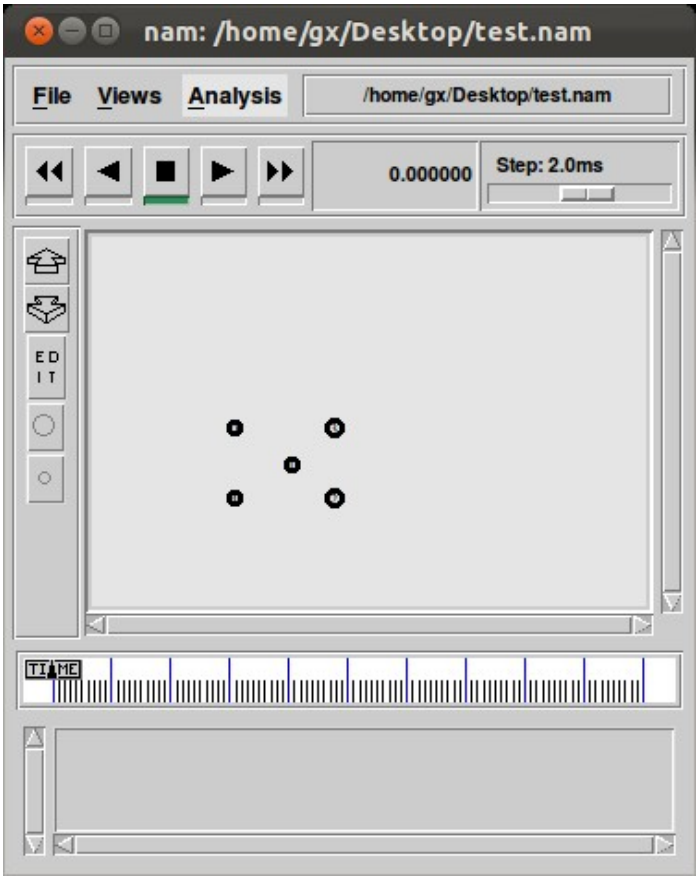


(c)

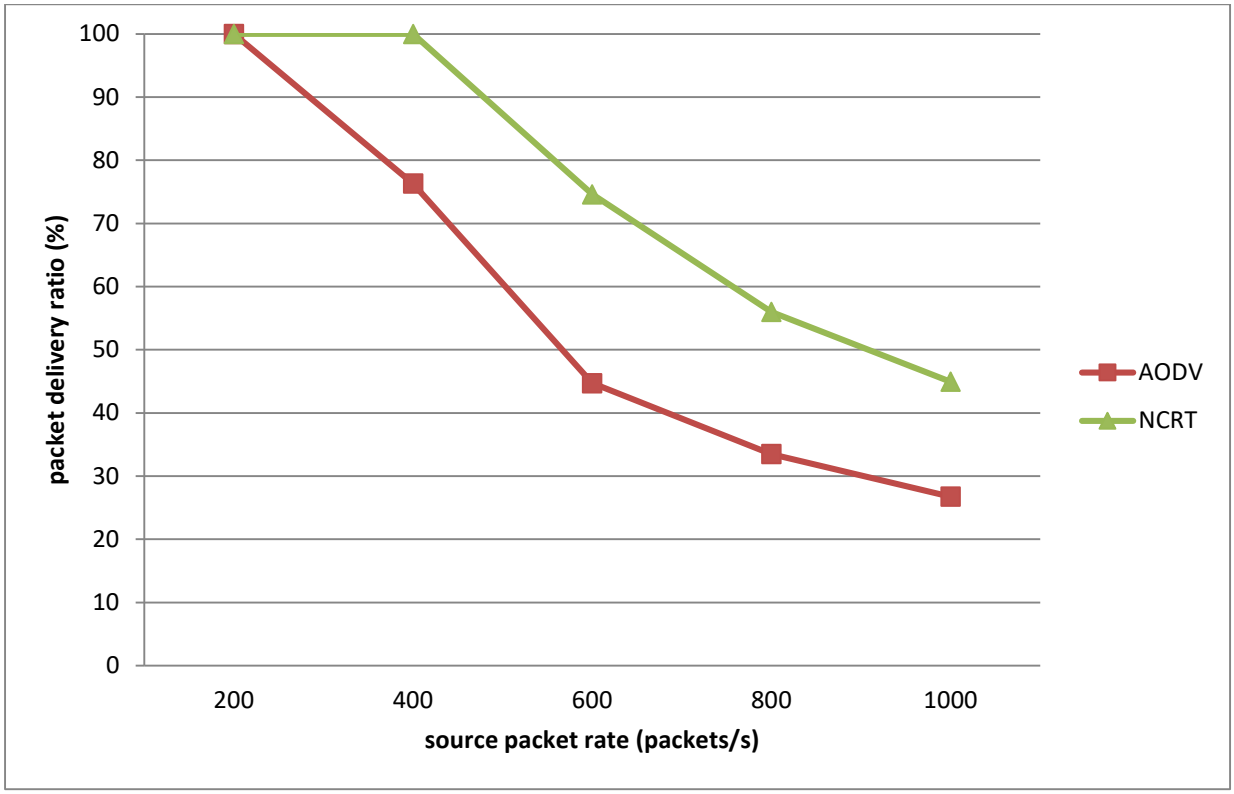


(d)

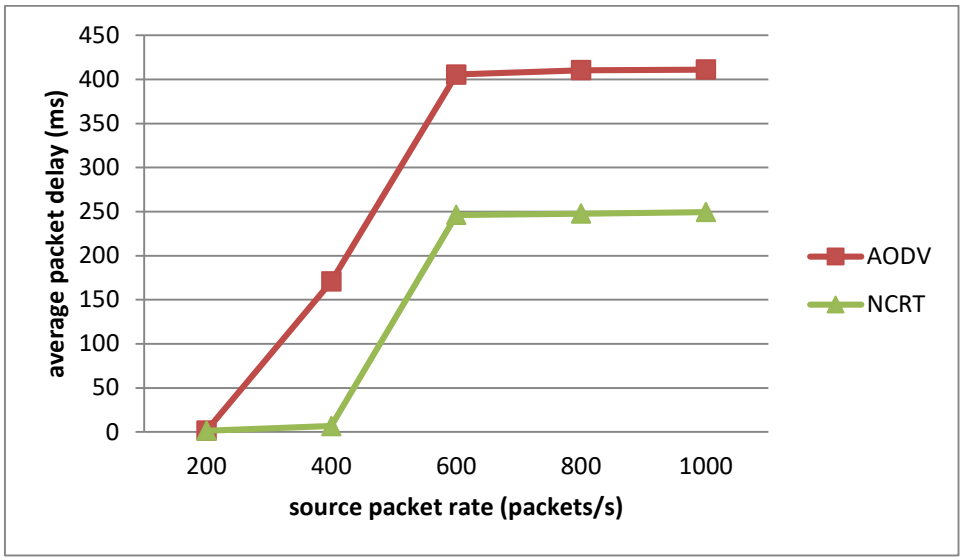
Figure 9. Three-hop chain: (a) topology (b) packet delivery ratio (c) average packet delay (d) average network throughput


nam: /home/gx/Desktop/test.nam
File
Views
Analysis
/home/gx/Desktop/test.nam
0.000000
Step: 2.0ms
EDIT
TIME

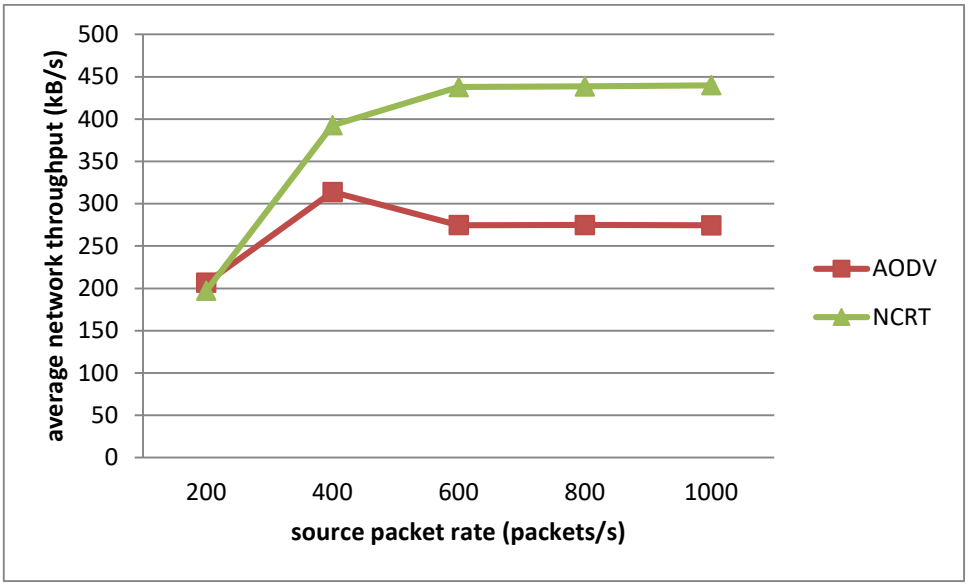
(a)


100
90
80
70
60
50
40
30
20
10
0
packet delivery ratio (%)
200
400
600
800
1000
source packet rate (packets/s)
AODV
NCRT

(b)



(c)



(d)

Figure 10. Cross topology: (a) topology (b) packet delivery ratio (c) average packet delay (d) average network throughput

Conclusions

In this paper we proposed SafeNet, a framework for disaster mitigation networks using programmable sensor nodes. We discussed about various aspects that we deemed helpful for such networks such as: i) placement of nodes, ii) different modes of operations, and iii) the required components. As would any other types of network, two different packet distribution methods are required i.e.: i) unicast, and ii) broadcast. For unicast routing, we proposed to incorporate network coding into the conventional AODV routing protocol resulting in the AODV-Network Coding (AODV-NC) routing protocol to save bandwidth and improve network throughput. For broadcasting, we proposed to adopt the Improved Partial Dominant

Pruning (IPDP) scheme to reduce broadcast redundancy. We also identified the need of and proposed the following: i) a service discovery scheme, and ii) centralized maintenance scheme. We evaluate the performance of AODV-NC in several scenarios and found that it outperforms AODV.

Acknowledgement

This project is funded by the AUN/SEED-net under the Special Research Program for Disaster Prevention and Mitigation (UM SDM 1201).

References

[1] G.X. Kok, C.O. Chow, and H. Ishii, "Reducing broadcast redundancy in wireless Ad-Hoc networks with implicit coordination among forwarding nodes," *Wireless Personal Communications*, October 2014. doi: 10.1007/s11277-014-2127-y

[2] C.E. Perkins, and E.M. Royer, "Ad-hoc on-demand distance vector routing," In: *The Proceedings of WMCSA'99 Second IEEE Workshop on Mobile Computing Systems and Applications*, Vol. 6, No. 3, pp. 90 – 100, 1999.

[3] D.B. Johnson, D.A. Maltz, and J. Broch, "DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks," *Computer Science Department, Carnegie Mellon University, The Monarch Project*, pp. 1–25, 2001.

[4] C. Ibm, and W.E. Perkins, "Highly dynamic (DSDV) for mobile computers routing," In: *SIGCOMM Computers Communication Review*, pp. 234–244, 1994.

[5] R. Ahlswede, and N. Cai, "Network information flow," *IEEE Transaction on Information Theory*, Vol. 46, No. 4, pp. 1204–1216, 2000.

[6] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *IEEE/ACM Transaction on Networking*, Vol. 16, No. 3, pp. 497–510, 2008.

[7] W. Lou, and J. Wu, "Toward broadcast reliability in mobile Ad Hoc networks with double coverage," *IEEE Transaction on Mobile Computing*, Vol. 6, No. 2, pp. 148–163, Feb. 2007.

[8] "The Network Simulator - ns-2", (n.d.) [Online]. Available: http://www.isi.edu/nsnam/ns/ [Accessed: Nov, 2012]