Network Coding Opportunities for Wireless Grids Formed by Mobile Devices

Karsten Fyhn Nielsen, Tatiana K. Madsen, and Frank H.P. Fitzek

Dept. of Electronic Systems, Aalborg University
{kfyhn,tatiana,ff}@es.aau.dk

Abstract. Wireless grids have potential in sharing communication, computational and storage resources making these networks more powerful, more robust, and less cost intensive. However, to enjoy the benefits of cooperative resource sharing, a number of issues should be addressed and the cost of the wireless link should be taken into account. We focus on the question how nodes can efficiently communicate and distribute data in a wireless grid. We show the potential of a network coding approach when nodes have the possibility to combine packets thus increasing the amount of information per transmission. Our implementation demonstrates the feasibility of network coding for wireless grids formed by mobile devices.

Keywords: Wireless grids, network coding, implementation, mobile phones.

1 Introduction and Motivation

Grid computing and grid topologies are attracting attention as sharing resources among devices has proven to bring benefits to the overall system performance [1]. A virtual pool of resources may consist of computing power, storage capacity or communication bandwidth. Despite of the variety of applications, the underlying concept for grid technologies is cooperation among devices and willingness to share resources. This concept has expanded into the world of wireless communication and by *wireless grids* we understand wireless devices forming cooperative clusters. Mobile devices within a wireless grid can use short-range links in addition to their cellular communication interfaces to share and combine resources and capabilities.

To exploit the benefits that device cooperation potentially offers, proper solutions should be found to a number of technical challenges. Compared with their wired counterpart, wireless grids are characterized by device mobility and fluctuating capacity of wireless links. However, both wireless and wired grid technologies have to overcome the following common set of challenges [2]:

- Efficient routing protocols
- Discovery semantics and protocols
- Security
- Policy management

P. Primet et al. (Eds.): GridNets 2008, LNICST 2, pp. 187–195, 2009.

In this work we focus on the first item that can be also formulated as *how to efficiently distribute data among devices in a cooperative cluster?* Wireless grids formed by mobile devices are known to be constrained by limited communication bandwidth. Additionally, mobile devices are battery powered and thus, energy limited. Most of the energy consumption of a mobile device is due to sending and receiving operations.

Under these conditions, efficient data distribution translates into minimization of the number of transmissions required to distribute the data. An extensive amount of research has been focused on power conservation techniques and energy-efficient routing algorithms for wireless grids (survey can be found e.g. in [3]). In this work we investigate another approach than traditional data forwarding. It is based on network coding and we demonstrate its benefits for wireless grids.

Using traditional routing techniques, intermediate nodes between the source and the destination are relaying and replicating data messages. With network coding, intermediate nodes code incoming messages, e.g. by using exclusive-ORs for packet combinations. The packets for coding are chosen in the way such that the destination node is capable of decoding information. The more coding opportunities an intermediate node can find, the greater is the amount of information combined in one coded packet and the less overall number of transmissions is required.

Many analytical studies and simulation evaluations advocate the usefulness of a network coding approach in terms of network throughput improvement. However its practical applicability for off-the-shelf devices with standard protocols can be demonstrated only by experimental studies. Experimental evaluation allows us to observe the influence of the realistic operation conditions, including error-proneness of the wireless channel and processing delays. Additionally, for wireless grids an extra challenge is the distributed nature of a network. There exist a few implementations that address the issue of network coding for wireless mesh networks. One should mention COPE [4] where devices perform opportunistic coding using XOR operation, and a recent work [5] presenting CLONE algorithms for unicast wireless sessions and [6] with a lightweight localized network coding protocol BFLY.

The focus of this paper is on understanding the potential of network coding implemented on small hand-held mobile devices forming a wireless grid. The implementation platform is Nokia N810, where a built-in WLAN interface is used for the local packet interchange. We limit our investigations to one example, a well-known "Alice and Bob" example [7] (explained in details in the next section). The performance of network coding is compared with a traditional routing approach based on a reliable broadcast implementation. We present a detailed study on how each individual transmission increases an accumulated knowledge in the whole network.

2 Problem Statement

We consider the following scenario consisting of three nodes: *A*, *B* and *C*. Nodes *A* and *C* are located far apart and can not directly communicate with each other. All packets have to be relayed through *B*, see Figure 1. We say that *A* and *B* are outside communication range, however they are within an interference range, thus the hidden terminal problem is eliminated.

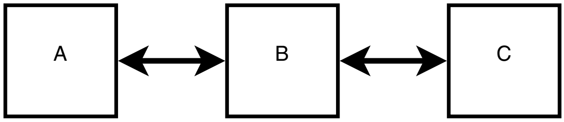

Fig. 1. The configuration of the three nodes for the test

It is assumed that each of the nodes has a unique part of a set of data, which all nodes need the full set of. When node *B* broadcasts its packets, they will be received by both *A* and *C*. Packets from *A* (and respectively from *C*) will be received by *B*, stored in its memory and forwarded to another node. Traditional routing approach consists in forwarding at the interlineate node *B*, resulting in one transmission pr. packet. With a network coding approach node *B* may XORs two packets received from *A* and *B* and broadcast that, resulting in one saved transmission.

For the simplicity of the further explanations, we consider that all data to be exchanged consists of 240 packets, 80 different packets on each node. The network is expected to behave as shown in Figure 2. The outcome of both distribution methods are separated into two phases.

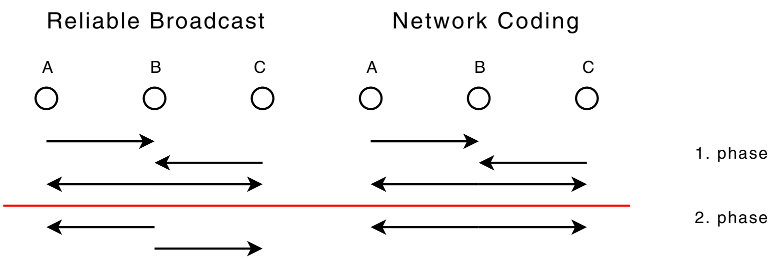

Fig. 2. The expected distribution of packets in the test

The first phases are identical. The two outer nodes send one packet to the middle node, and the middle node broadcasts one of its own packets to the two outer nodes. This results in four satisfied nodes using three transmissions. This can be used to denote the *"speed"* of the distribution. Here we say the speed is 4/3, because four pieces of information were received using three transmissions. The speed is larger than one, because of the broadcast. This phase is repeated until the two outer nodes have sent all of their packets to the middle node, and the middle node has broadcasted all of its own packets. This is the end of the first phase, and it occurs when 77,78 % of the packets are distributed in the system on average, because each outer node has 2/3 of the packets and the middle node has all the packets. Then the second phase begins. With reliable broadcast the middle node must transmit 160 packets in 160 transmissions, because only one node can use each packet. This results in a speed of one, i.e. a lower speed than in the first phase. With network coding the middle node can code together two packets and broadcast the result, thereby sending 160 packets in 80 transmissions. This gives a speed of 2, i.e. a higher speed than in the first phase. Assuming no collisions and thus no retransmissions, this results in reliable broadcast

Fig. 3. The implementation platform and GUI

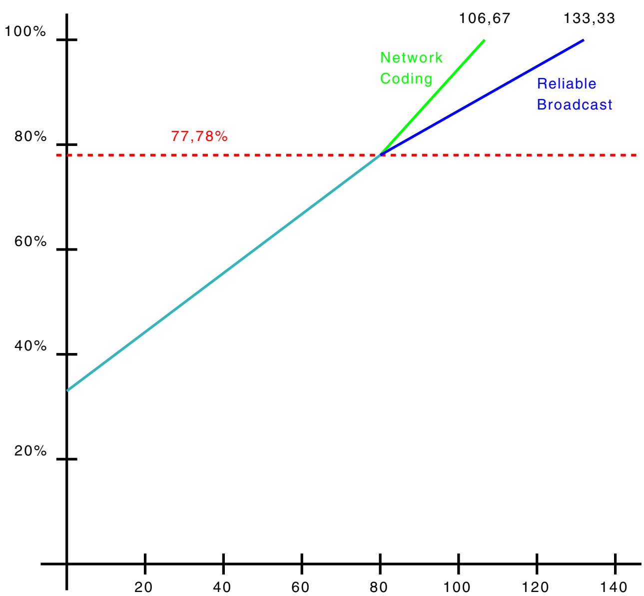

Fig. 4. The expected outcome of the test. The dotted line separates the two phases of the distribution at 77,78 %.

finishing in (80+80+80+80+80)/3=133,33 transmissions, whereas network coding should be able to finish in (80+80+80+80)/3=106,67 transmissions. The expected results of the test are shown in Figure 4.

In the following we compare the measured data exchange process with the presented theoretical one. But first, we describe implementation details.

3 Implementation

As a platform for the implementation the Nokia N810 Internet Tablet is chosen. This is chosen because of its large screen, which makes it useful for visualization, because of its build in WLAN interface, and because of it's operating system, which is a Linux distribution, making it easy to develop for because of the many readily available tools. The platform has the following specifications:

- *Processor* - TI OMAP 2420, 400 MHz ARM11.
- *Memory* - 128 MB + 128 MB swap.
- *WLAN* - IEEE 802.11 b/g (only b in ad-hoc).
- *Operating System* - Maemo[1] OS2008 (Linux kernel 2.6.21-omap1)

Programs must be compiled for the ARM processor, so a special environment platform is set up for the development. This platform is called Scratchbox[2] and is a cross-compilation toolkit, which may be used with Maemo SDK for software development for the Maemo platform.

The developed implementation consists of the following four levels:

- *Test Application*: At this level, a GUI has been implemented to show the distribution of packets (Figure 3). This level also manages a logging facility.
- *Framework*: This level is responsible for placing the implementation of network coding as an extra protocol between the IP and MAC layers. This has one very important advantage. It relies only on the information in the IP-header, which may be retrieved using raw sockets directly after the MAC layer. By also placing the implementation beneath the normal IP layer, it becomes possible to use a virtual network interface, as the interface between the first and the second level. Thereby communication between the two levels can happen solely through standard Berkely socket calls. The framework also provides functions for sending packets through the socket, and for sending special packets, that start and stop the lower levels. With very few changes this level could be run as a daemon on the OS instead, making it a completely separate entity from the application or applications using it.
- *Logistics Platform*: It contains all the data structures and functions for the logistics of network coding. In the implementation of network coding for distributed wireless grids, this especially is the knowledge of which packets the local node has, and the knowledge of which packets all remote nodes have.
- *Schemes*: This level is the algorithms for encoding and decoding. These may be different as well, even while using the same logistics. For this implementation two schemes have been implemented. One scheme for reliable broadcast, and one for network coding.

The two schemes are further explained in the following, but first some important data structures are introduced, which are heavily used in the schemes:

[1] http://www.maemo.org

[2] http://www.scratchbox.org

- *NC packet*: When a node sends out a transmission, we refer to it as an NC packet. An NC packet may contain zero, one or a combination of many IP packets, depending on what the scheme has found. But always present in an NC packet is an NC header. This header is used to distribute knowledge between the nodes, e.g. the nodes reception vector.
- *Reception vector*: The reception vector linked to a node is a bitarray with as many bits as there are packets in the set.
- *Packet pool*: The reception vectors are closely intertwined with the packet pool. For a scheme to be able to minimize the number of transmissions, by sending combinations of IP packets in each transmission, the IP packets must be readily available in some form of persistent store. Packets are therefore not only forwarded between the application on the first level and the network, but are stored in the logistics platform, to provide the scheme with more coding opportunities.

The usefulness of these data structures becomes clear in the following, where the two implemented schemes are explained.

The algorithm for network coding finds the IP packets to send in one NC packet using exclusive-ORs for packet combinations. This is done by iteratively finding IP packets to code with, until all coding opportunities have been explored. Coding opportunities is defined as the opportunity for a local node to combine one or more packets together for a single transmission. Because each NC packet is only useful to a node if it contains only one unknown IP packet, it is important to take this into account in the algorithm. If an IP packet is chosen for transmission with node A as receiver, no IP packet unknown to node A must be chosen in any of the following iterations. To avoid this, a *codingvector* is used to contain knowledge of IP packets, which can be used for coding. The codingvector must for each iteration contain the ids of IP packets, which all receivers found in previous iterations have in common. Therefore, if one of these is chosen for transmission, all receivers can decode it. It is also important that when an IP packet A is chosen for transmission in an iteration, none of the IP packets found in previous iterations must be unknown to the receiver(s) of A. In the implementation of a scheme for network coding, this has been solved by AND'ing the codingvector with the vectors of the receiving nodes in each iteration.

The scheme for reliable broadcast is somewhat similar and much simpler. At each iteration a node has to choose an IP packet for transmission. The algorithm for reliable broadcast runs through all available IP packets on the local node, to find the first IP packet missing on a remote node. Exchange of reception vectors plays a role of acknowledgements.

4 Results

In the experimental evaluation we focus on the average number of packets available on the nodes as a function of the average number of transmissions for each node. The results are averaged over five test runs. The result is normalized to show the number of packets in percent of the total set.

The result of the test is shown in Figure 5. As can be seen, the test run is separated into two phases, the first running from start and until the system in average has 77 % of the set, and the second from then and until all nodes have the entire set. The first phase is when all nodes are sending with a speed of 4/3. This is the case for both transmission methods, and they are therefore similar. In the second phase, reliable broadcast only sends with a speed of 1, and therefore the curve declines, whereas the speed of network coding becomes 2, so its curve increases.

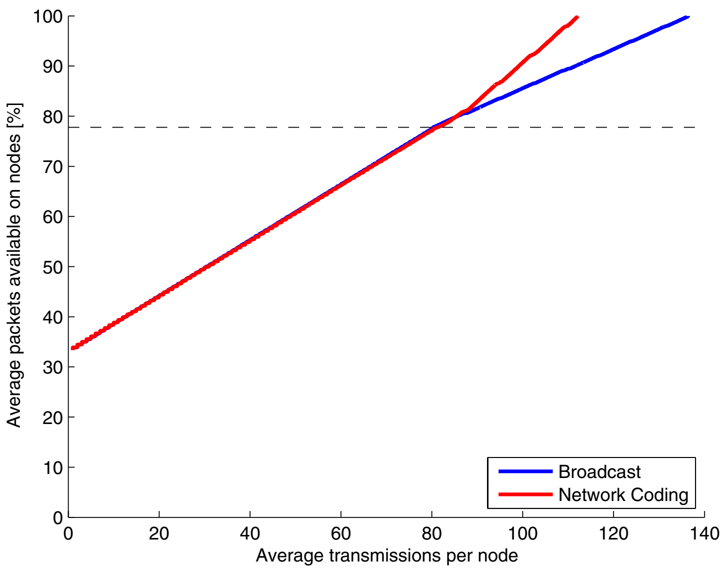

Fig. 5. The measured results of the test. The dotted line separates the two phases of the distribution at 77,78 %.

The measured and theoretical transmission rates are compared in Table 1. The reliable broadcast scheme distributes the set with 136,4 transmissions on average, where the theoretical limit was 133,33, giving only three redundant transmissions per node, and network coding finishes with 112,2 transmissions compared to the theoretical 106,67,meaning six redundant transmissions per node.

Table 1. Average number of transmissions per node required to distribute data for the two transmission schemes

	Measurements	**Theory**
Reliable Broadcast	136,4	133,33
Network Coding	112,2	106,67

These redundant transmissions are caused by an initial transmission, when nodes do not have knowledge about the status of other nodes. Additionally, collisions, retransmissions and acknowledgements are other reasons for redundant transmissions. The behavior of the three individual nodes is shown in Figures 6 and 7.

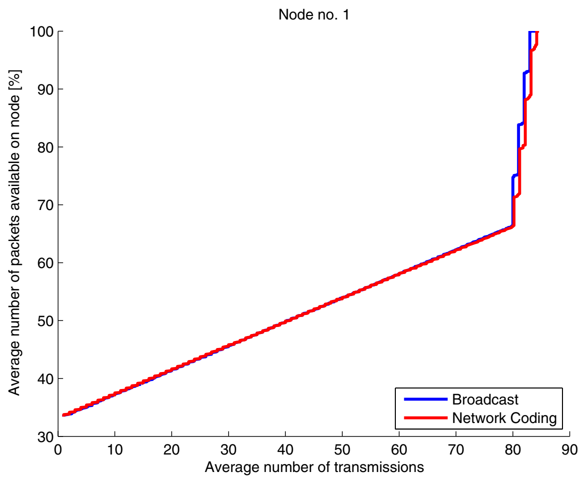

Fig. 6. The result of the test for the outer nodes A and C

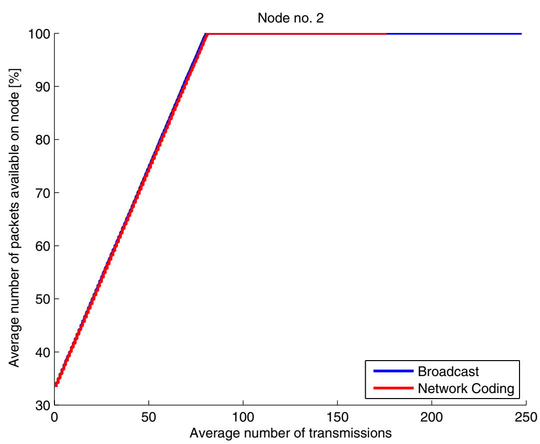

Fig. 7. The result of the test for the middle node B

5 Conclusions

This paper demonstrates the practical potential of network coding for wireless grids using a simple three node example. As can be seen from the test bed results, network coding increases the throughput gained from sending the same information with fewer transmissions with approx. 20 % compared to reliable broadcast. A closer look at system behavior shows that after some time network coding is "speeding up" distributing information in the network, while a traditional approach based on broadcast is slowing down. The measured number of required transmissions is very close to the theoretical one and suggests the feasibility of network coding implementation even on small hand-held devices such as mobile phones. A slight difference in the results can be explained by redundant transmissions and acknowledgements and most probably it can be illuminated or at least improved by using *intraflow* network coding [7] instead of sending single packets. This would greatly reduce the need for acknowledgements, and thereby reduce the required number of transmissions further.

References

1. Fitzek, F., Katz, M.: Cooperation in Wireless Networks: Principles and Applications - Real Egoistic Behavior is to Cooperate! Springer, Heidelberg (2006)
2. Agarwal, A., Norman, D., Gupta, A.: Wireless Grids: Approaches, Architectures and Technical Challenges. MIT Sloan Working Paper No. 4459-04 (2004)
3. Ahuja, S.P., Mayers, J.R.: A Survey on Wireless Grid Computing. The Journal of Supercomputing 37(1), 3–21 (2006)
4. Katti, S., Rahul, H., Hu, W., Katabi, D., Medard, M.: The Importance of Being Opportunistic: Practical Network Coding for Wireless Environments. In: Allerton Conference (2005)
5. Rayanchu, S., Sen, S., Wu, J., Sengupta, S., Banerjee, S.: Loss-Aware Network Coding for Unicast Wireless Sessions: Design, Implementation, and Performance Evaluation. ACM SIGMETRICS Performance Evaluation Review 36(1), 85–96 (2008)
6. Omiwade, S., Zheng, R., Hua, C.: Butterflies in the Mesh: Lightweight Localized Wireless Network Coding. In: Proc. of Fourth Workshop on Network Coding, Theory, and Applications (2008)
7. Fragouli, C., Katabi, D., Markopoulou, A., Medard, M., Rahul, R.: Wireless Network Coding: Opportunities & Challenges. In: Military Communications Conference, MILCOM 2007, pp. 1–8 (2007)