Cross-Layer Support for Group-Communication Applications in MANETs*

Marco Conti, Franca Delmastro
CNR, IIT Institute
Via G. Moruzzi, 1 – 56124 Pisa, Italy
{firstname.lastname}@iit.cnr.it

Jon Crowcroft, Andrea Passarella
University of Cambridge, The Computer Laboratory
15 JJ Thomson Avenue – Cambridge CB3 0FD, UK
{firstname.lastname}@cl.cam.ac.uk

Abstract

P2P systems are a natural way of supporting group-communication applications in MANETs. In this paper we discuss our experiences in developing such an application in the real world. We highlight limitations of legacy P2P systems, and show that solutions based on cross-layer optimisations are very promising.

1 Introduction

One of the most interesting class of applications that can be envisaged for MANETs is represented by group-communication applications. In the framework of the MobileMAN Project [6], we are investigating the viability of developing such kind of applications on real ad hoc networks. To this end, we developed the Whiteboard application (WB), which implements a distributed whiteboard among MANET users. WB usage is very intuitive (see Figure 1). Each MANET user runs a WB instance on her device, selects a topic she wants to join, and starts drawing on the canvas. Drawings are distributed to all nodes subscribed to that topic, and rendered on each canvas. We believe that these simple, "Plug&Play" applications will be of great value for MANET users.

Developing this kind of applications in MANETs is a challenging task. In this paper we present the networking solutions we have studied and tested to this end. We present alternative networking frameworks for supporting WB-like applications (Section 2). Then, we compare a standard P2P system (Pastry [8]) with CrossROAD [5], the P2P system optimised for MANETs that we have designed within these frameworks (Section 3). Advantages of the CrossROAD approach are presented by means of experimental results in Section 4. Finally, Section 5 concludes the paper.

2 WB integration in MANETs

Group-communication applications such as WB are distributed, self-organising, decentralised in nature. Designing them on top of P2P systems guarantees a great flexibility and optimised performances exploiting P2P policies to distribute and recover information. Figure 2 depicts the ab-

*This work was partially funded by the FET-IST Programme of the European Commission, IST-2001-38113 MOBILE-MAN project.

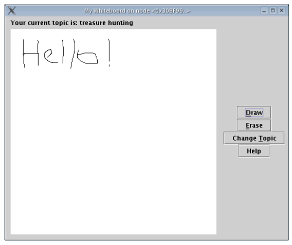

Figure 1. The WB application interface

stractions we have used to support WB. The network level provides basic connectivity among nodes through IP-like routing and transport protocols. On top of them, a structured overlay network, comprising nodes that participate in the WB application, is built. The overlay abstraction is the fundamental substrate for any P2P application, providing functionalities such as logical node addressing (instead of topological, IP-like addressing) and subject-based routing. Finally, an additional multicast level is used to efficiently distribute contents generated by application users to all nodes in the overlay. These abstractions make quite straightforward develop group communication applications. They hide the complexity of low-level communications, group management, and data distribution, and provide a robust, flexible, self-organising networking environment.

Figure 3 shows the complete networking solutions we have used to support WB in real-world MANETs. We have defined a first architecture (referred to as *legacy*), that uses state-of-the-art components to implement the abstractions in Figure 2. Specifically, it uses either AODV [1] or OLSR [7] at the network level, Pastry [8] at the middleware level, and Scribe [2] at the multicast level. While AODV and OLSR represent standard models for ad hoc reactive and proactive routing protocols, Pastry and Scribe have been designed for wired networks. The evaluation of the "legacy solution" indicates weaknesses of these components, and ways to improve them. In order to optimize the entire system performances, a *cross-layer* architecture, as depicted on the right-hand side of Figure 3, has been proposed in [4]. Specifically, the NeSt module allows cross-layer interactions between protocols at different layers. To this aim, NeSt provides well-defined interfaces and data abstractions to protocols [4], joining the advantages of cross-layering

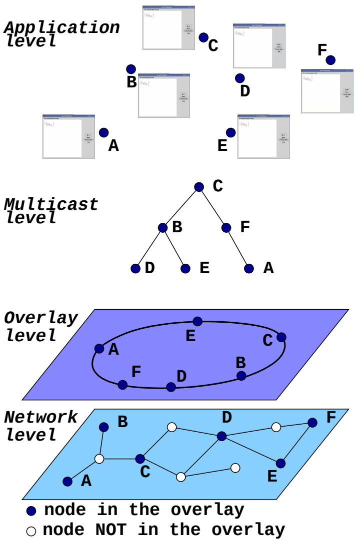

Figure 2. Abstractions supporting WB

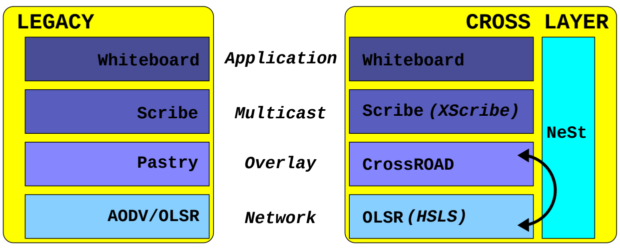

Figure 3. Network solutions: legacy (left) and cross layer (right)

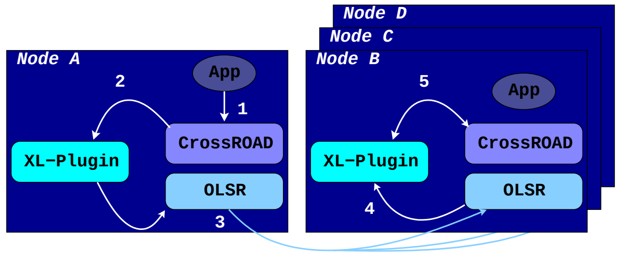

Figure 4. Cross-layer interactions between CrossROAD and OLSR

and the scalability of traditional layered approach. CrossROAD represents an optimised solution at the middleware layer that exploits cross-layer interactions with a proactive routing protocol (OLSR in this case) in order to optimize the creation and management of the overlay network. In this paper we do not discuss any other MANET-optimised solutions that could be integrated into the cross-layer architecture. However, in the framework of the MobileMAN project, other such components both at the routing level (Hazy Sighted Link State [9]), and at the multicast level (X-layer Scribe) are being studied and currently under development.

3 Pastry vs. CrossROAD

Pastry represents the P2P computing model on which CrossROAD has been designed to obtain great optimisations on ad hoc networks. It generates an overlay network by organising nodes in a circular logical address space (ring). Specifically, it assigns to each node a *logical* identifier by hashing, for example, the node IP address. Logical identifiers determine the node position in the ring. In addition, messages are routed over the ring by following a subject-based policy, rather than a topology-based one. Specifically, an application wishing to send a message m has to provide a key k linked to m. The k value is hashed to obtain an identifier in the same space of nodes' logical ids that is used to select the best destination for that message. The subject-based routing of Pastry sends the message to the node in the ring whose id is numerically closest to the key hashed value. This policy represents the basis for several distributed services; for example, Scribe exploits subject-based routing to build and maintain multicast distribution trees.

To implement subject-based routing, Pastry builds at each node a middleware routing table storing a subset of other nodes' ids. This table is initialised (during a bootstrap phase) and updated (periodically) by exchanging information with the other nodes. When adopted in MANETs, this approach generates quite a lot of network overhead. CrossROAD [5] provides the same Pastry functionalities through the P2P commonAPI [3], but it drastically reduces the overlay management traffic by exploiting cross-layer interactions with a proactive routing protocol. Specifically, CrossROAD implements a Service Discovery protocol, that exploits the broadcast flooding of routing packets to distribute services information. An example of cross-layer interaction between CrossROAD and OLSR is shown in Figure 4. Each application running on CrossROAD has to register itself by specifying a *service id* (step 1). The list of service ids registered at the local node (Node A in the figure) is maintained by the Cross-Layer Plugin (XL-Plugin), which can be seen as a portion of the NeSt module (step 2). The XL-Plugin embeds the list of local service ids into periodic Link-State Update packets generated by OLSR (step 3). On the other nodes of the network (nodes B, C, D in the figure), upon receiving LSU packets containing such list, the routing level notifies XL-Plugin to store the list in its internal data structures. This way, each CrossROAD node has a complete knowledge of all the other nodes providing the same service in the MANET, and it is able to autonomously build the overlay network without generating any further management traffic (step 5). Furthermore, in case of topology changes, the status of the overlay network converged as quickly as the routing protocol does.

4 Experimental Results

The networking solutions described in Figure 3 have been implemented and tested in a read-world multi-hop ad hoc network. Specifically, the testbed consisted of 8 homogeneous laptops, out of which 6 run the WB application, and the remaining 2 were used just as routers. Experiments that have been run, which mimic the behavior of WB users concurrently drawing strokes on their canvas. Users are represented by software agents that continuously interleave active phases (during which they draw a burst of strokes), and idle phases (during which they just receive others' bursts). Idle phase durations and burst sizes are exponentially distributed. A traffic load of 100% is defined as the load generated by a user drawing – on average – 1 stroke per second.

Due to space constraints, we cannot provide here detailed measurements. Therefore, we discuss the outcomes of some selected experiments, that allow us to highlight several benefits introduced by CrossROAD[1]. Table 1 shows the aggregate throughput (in the sending and receiving directions) of each node during the Pastry 80% and CrossROAD 100% experiments, respectively[2]. These results account for the traffic generated from the routing up to the application level. We mark node C as "C(R)" since it was the root of the Scribe tree. Finally, the last two rows show the average throughput computed over the nodes running WB including and excluding C, respectively. Overall, when CrossROAD is used instead of Pastry, the throughput is drastically reduced. The average value over all nodes in the CrossROAD setup is about one third of the average value in the Pastry setup. It should be noted that, due to Scribe mechanisms, the root node has to handle a far greater amount of application-level traffic than other nodes. Therefore, the throughput reduction due to CrossROAD can be better emphasised by focusing on the last row of the table. If we exclude node C, the average throughput in the CrossROAD setup is about *one fourth* of the average throughput in the Pastry setup. Finally, we found that CrossROAD also improves the stability of the Scribe tree. Table 2 shows the number of sub-trees that are generated in Pastry and CrossROAD setup, respectively. When Pastry is used, the Scribe tree is often partitioned in several isolated sub-trees, resulting in nodes to be isolated from the rest of the network. Instead, this misbehavior is always avoided when CrossROAD is used. It can be shown that it is a byproduct of the Pastry network overhead and bootstrap procedure.

Node	Pastry	CrossROAD
A	16529	2966
B	21278	5069
C(R)	48542	21146
D	29066	7819
E	18047	5993
F	14964	4313
avg	24738	7884
avg (no C)	19977	5232

Table 1. Throughput (Bps) in the Pastry 80% and CrossROAD 100% setup.

Load	Pastry	CrossROAD
20%	1	1
50%	2	1
80% (100%)	3	1

Table 2. Number of sub-trees at the Scribe level.

5 Conclusions

In order to support P2P group-communication applications in MANETs, legacy network architectures designed for *wired* networks are not the real solution. Specifically, such solutions require too much management traffic, and tend to saturate the scarce MANET resources. Optimising the network stack components through cross-layering is a very promising way. In this paper, we have highlighted drastic performance improvements by replacing Pastry with CrossROAD, a cross-layer optimised P2P substrate. Further improvements might be envisaged if also the other P2P components (e.g., Scribe) are optimised according to the cross-layer paradigm.

[1] In the Pastry case, we herafter show only results from OLSR experiments, since OLSR generally allowed to achieve better performances than AODV.

[2] We were not able to run Pastry experiments at 100% traffic load, because the testbed crashed due to excessive network load.

References

[1] AODV, Dept. of Information technology at Uppsala University (Sweden), http://user.it.uu.se/ henrikl/aodv/.

[2] M. Castro, P. Druschel, A-M. Kermarrec and A. Rowstron, "SCRIBE: A large-scale and decentralised application-level multicast infrastructure", IEEE Journal on Selected Areas in Communication (JSAC), Vol. 20, No, 8, October 2002.

[3] F. Dabek and B. Zhao and P. Druschel and J. Kubiatowicz and I. Stoica, "Towards a common API for Structured Peer-to-Peer Overlays", Proc. of the the 2nd International Workshop on Peer-to-peer Systems (IPTPS'03), Berkeley, CA, Feb. 2003.

[4] M. Conti, G. Maselli, G. Turi, "Design and evaluation of a flexible cross-layer interface for ad hoc networks", Proc. of the Fourth Annual Mediterranean Ad Hoc Networking Workshop (MedHocNet 2005), June 2005.

[5] F. Delmastro, "From Pastry to CrossROAD: Cross-layer Ring Overlay for Ad hoc networks", in Proc. of Workshop of Mobile Peer-to-Peer 2005, in conjuction with the PerCom 2005 conference, Kauai Island, Hawaii, Mar. 2005.

[6] "Mobile Metropolitan Ad hoc Network (MobileMAN)", IST-2001-18113 Project, funded by the EC FET-IST Programme, `http://cnd.iit.cnr.it/mobileMAN/`

[7] OLSR, Andreas Tonnesen, Institute for informatics at the University of Oslo (Norway), http://www.olsr.org.

[8] A. Rowstron and P. Druschel, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems", Middleware 2001, Germany, November 2001.

[9] C.A. Santivanez, I. Stavrakakis, and R. Ramanathan, "Making link-state routing scale for ad hoc networks". In Proceedings of the 2nd ACM Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC'01), 2001.