Enhancing Application-Layer Multicast Solutions by Wireless Underlay Support

Christian Hübsch and Oliver P. Waldhorst

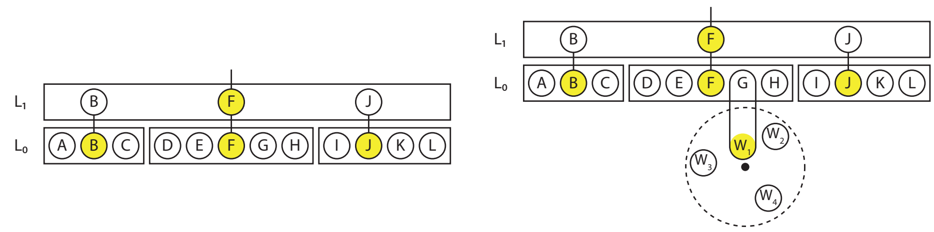
Institute for Telematics, University of Karlsruhe (TH), Germany

Abstract. Application Layer Multicast (ALM) is an attractive solution to overcome the deployment problems of IP-Multicast. We show how to cope with the challenges of incorporating wireless devices into ALM protocols. As a first approach we extend the NICE protocol, significantly increasing its performance in scenarios with many devices connected through wireless LAN.

1 Introduction

Novel Internet applications such as radio and TV broadcasting, live video streaming, video conferencing and multiplayer games increase the demand for multicast communication. Unfortunately, IP-multicast is far from global deployment due to administrative, technical and bussiness issues. Application-layer multicast (ALM) is an attractive solution to cope with these issues, since it implements multicast functionality on end systems, where it can easily be installed as a piece of software. Several approaches for ALM have been proposed in recent years [5], including popular approaches like *Narada* [6] and *NICE* [1]. The latter is described shortly in Section 2. Recent work focuses on building resilience and efficiency in the ALM topologies by considering heterogeneous and short-lived devices [7] or building inner-node disjoint multicast trees [8]. Other recent work has shown how to transparently employ ALM for applications based on IP-Multicast [9].

ALM protocols face new problems due to an increasing number of devices that connect to the Internet using wireless technology like wireless LAN or cellular connections. Typically, such devices decrease ALM performance since they have lower connection bandwidth or may be located in a shared medium with limited capacity. While some existing approaches like [2] address ALM in wireless ad-hoc domains, most protocols only refer to member node capacities, not considering heterogeneous underlay properties. In this paper, we propose to incorporate awareness for wireless devices into ALM protocols. We present an extension of NICE denoted as *NICE with WireLess Integration* (*NICE-WLI*) (Section 3). As major features, in NICE-WLI (1) all device within the same wireless LAN are represented by a single gateway node within the NICE ALM-topology and (2) gateway nodes are not assigned demanding tasks like cluster leadership. We present initial performance results showing that NICE-WLI significantly reduces the traffic load in the wireless network (Section 4). The work presented



(a)The original NICE cluster hierarchy (b)Logical protocol parts in NICE-WLI

Fig. 1. The NICE cluster hierarchy and the NICE-WLI extension

in this paper is part of the Spontaneous Virtual Network (SpoVNet) project (http://www.spovnet.de) and constitutes a step towards providing a SpoVNet group communication service.

2 Nice

The NICE protocol [1] is an overlay-based application-layer multicast (ALM) protocol that has been developed in the context of the correspondent NICE project. NICE arranges its members in a hierarchical structure of clusters, aiming to provide communication in large scale groups. Members only exchange protocol maintenance messages inside the clusters they reside in, thus providing good scalability properties. Clustering is accomplished based on a distance metric between nodes (network latency in the original proposal). Each cluster in layer L_i elects a leader (starting with L_0 as the lowest layer), located in its graph-theoretic center. Cluster leaders form new logical clusters in a higher layer L_{i+1} which again elect leaders, resulting in a hierarchical structure of layered clusters. For data dissemination, cluster members send their data to all cluster neighbors, including the specific cluster leaders. The latter send the data to all nodes located in clusters they are leader of. Fig. 1(a) shows an exemplary hierarchy. A detailed descriptions can be found in [1].

3 NICE-WLI Protocol Design

This chapter describes the design of *NICE-WLI (NICE with WireLess Integration)* and its modifications to the original NICE protocol.

For employing NICE-WLI, we assume the existence of an out-of-band mechanism for discovering a node's local network context. This means every node is capable of knowing its connectivity properties, in case of NICE-WLI either the underlay connection via fixed network or its attendance in a particular wireless (WiFi) domain. In the latter case, also parameters like signal-to-noise ratio (SNR) to the access point as well as the used SSID is known. The protocol enhancement is based on partitioning the overlay instance into two logical parts, the core part (which is identical with the conventional NICE) and the WiFi part. The WiFi part may split into multiple WiFi domains, each uniquely identified by the specific SSID and the access points' MAC address. In NICE-WLI, WiFi domains are handled as logical entities that are represented by exactly one L_0 cluster member in the overlay hierarchy (Fig. 1(b)). This member encapsulates the whole domain, both if it is the only WiFi node and if there are other nodes residing in the WiFi domain. This allows for minimizing wireless network traffic while keeping the wireless underlay drawbacks and properties out of the fixed network structure. The transition between the two protocol parts is accomplished by Gateway Nodes.

3.1 Gateway Nodes

A Gateway Node in NICE-WLI operates hybridly by communicating with the fixed network structure and also with the nodes in the connected WiFi domain. Being member of the WiFi domain itself, it is the only node in the WiFi domain that is also embedded in the NICE cluster hierarchy: Every Gateway Node is also a non-cluster-leader member in a L_0 cluster. A Gateway Node therefore represents all NICE members that are part of this WiFi domain and manages most protocol mechanisms and data forwarding. In Fig. 1(b), the Gateway Node is marked as W1. NICE-WLI always keeps Gateway Nodes out of the cluster leader disposition process. This is crucial, as one of the main design targets in NICE-WLI is communication efficiency and a cluster leadership may increase communication overhead in the Gateway Node up to $O(k\ log\ N)$ (k being the cluster size parameter in NICE) in the worst case, which is not desireable in shared mediums like WiFi domains. Also, cluster leaders occupy higher responsibility in terms of protocol stability, while WiFi domain members rather tend to be prone to errors. When more than one node resides in the same wireless domain, NICE-WLI aims at assigning the Gateway role to the one that is next to this domain's access point (in terms of SNR). Gateway Nodes define a new role a member can take in the overlay and are a key aspect in NICE-WLI.

3.2 Protocol Operation

This section describes the NICE-WLI protocol operation and how it handles to connect WiFi domains to an overlay instance efficiently. As long as members only join the overlay through fixed line connections, the overlay protocol operates as NICE does in its original proposal. As soon as a WiFi node wishes to join the overlay, it first checks the existence of a Gateway Node in the wireless domain. This is accomplished by broadcasting a *Gateway Discovery* message. Should no *Gateway Response* be received in a certain period of time, the node assumes to be the first overlay member in the specific WiFi domain and takes the role of the Gateway Node. Thereby it connects to the overlay and joins the L_0 cluster it belongs to in terms of network latency. If a Gateway Node responds to the *Discovery* in time, the joining node remains loosely coupled to the Gateway Node instead of joining the fixed network structure (W2, W3, W4 in Fig. 1(b)). Just like in the original NICE, nodes in NICE-WLI's core part exchange protocol information by sending periodic state messages (Heartbeats) to their cluster neighbors. This enables the current Gateway Node in a WiFi domain to inform the L_0 cluster members (especially the L_0 cluster leader) of its special role. By this, the cluster leader is capable of excluding the Gateway Node from the cluster leader determination process. Due to the central role of the Gateway Node, it should be chosen to have the best connectivity to the wireless network's access point. Because wireless nodes may be subject to mobility and varying conditions, NICE-WLI refines its roles periodically. Every node broadcasts its SNR per Heartbeat, enabling the Gateway Node to compare all participating members' quality of connection. Should one SNR exceed a specific difference to the current Gateway's (based on a Weighted Moving Average estimation), the Gateway role is transferred to this node, causing him to join the L_0 cluster. In case a node leaves the wireless domain, it has to be distinguished between Gateway Nodes and non-Gateway Nodes. In the latter case, the node may simply leave without any further protocol signaling, for no logical relationship had been established that should be released. If a Gateway Node leaves, it first has to sign off its L_0 cluster (like in NICE); additionally, it has to elect a new Gateway Node based on the current SNRs. If a Gateway Node leaves 'ungraceful', meaning the Gateway Node leaves unpredictably, both the NICE overlay and the wireless NICE-WLI part detect this by not receiving any further Heartbeats. The NICE overlay reacts like described in [1], while the wirless nodes begin to elect a new Gateway Node based on the SNRs they know from each others Heartbeats.

Finally, it may in some cases appear that two nodes take the role of the Gateway Node in a WiFi domain coincidentally. This may happen e.g. due to lost responses to *Gateway Discoveries*. In such cases, the Gateway Nodes recognize this situation by means of their mutual Heartbeat messaging and dissolve it. In phases of such duplicate Gateway Nodes, loops may occur in data forwarding (because both Gateway Nodes forward data they receive in the wireless domain to the fixed connection NICE part). To solve this, data packets hold a flag that marks them as being sent by a Gateway Node. If another Gateway Node receives such a packet, it does not forward the data, instead intiating the process of Gateway election to resolve the duplicate Gateway Nodes. In case data from wireless protocol parts reaches the core part anyhow, the specific L_0 cluster leader is able to detect duplicate packets based on their origin from Gateway Nodes from the same wireless domain.

3.3 Data Dissemination

While 3.2 described how NICE-WLI arranges nodes, this section discusses how data is disseminated in the overlay. Inside the core part, data is forwarded like desribed in [1], where cluster leaders send packets to all nodes located in clusters they are leader of. As soon as a data packet reaches a Gateway Node in a L_0 cluster, this Gateway Node has to disseminate the data in its WiFi domain, for there is no knowledge about the wireless part in the core parts. Additionally, the Gateway Node has to minimize wireless network traffic s.t. the medium is

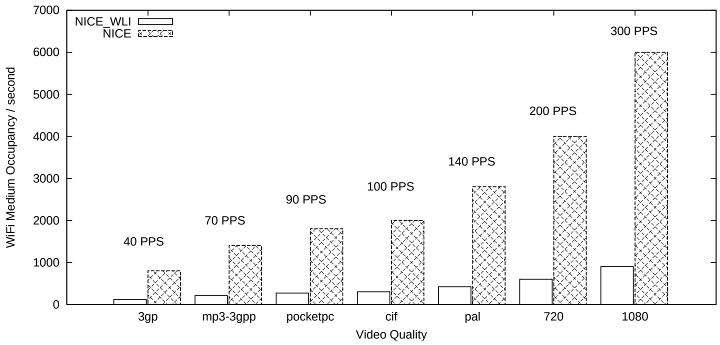


Fig. 2. Wireless Medium Occupancy in NICE-WLI

not occupied more than necessarily required. Therefore, upon reception in the L_0 cluster, the Gateway Node broadcasts the data packet in its WiFi domain, reaching all protocol members with a single transmission.

Nodes in the wireless network wishing to send group-related data also broadcast their packets, reaching other wireless members including the Gateway Node. The Gateway Node then forwards the data to its L_0 cluster. While in the original NICE, cluster members unicast their data to all other cluster members and the cluster leader, this is different in NICE-WLI. To minimize the use of the shared medium, Gateway Nodes send data from the wireless network only to their L_0 cluster leader, allocating the medium only once. Knowing that the data's origin is a Gateway Node, the L_0 cluster leader disseminates the data to the remaining L_0 cluster members and all other nodes in clusters it is leader of.

4 Evaluation

In this section we analyze how much transmission overhead can be saved in WiFi domains by using NICE-WLI compared to NICE. In our estimation we assume an overlay of 1000 nodes of which are 5% connected through one wireless domain.

To appraise the savings in data dissemination, we have to presume an application that uses the overlay to send its data. In our case we look at video streaming in different quality levels, ranging from low-quality (3gp) to high-quality (HDTV) video. Fig. 2 shows the theoretic number of wireless media allocations, meaning how often the medium has to be allocated to transmit a packet to a wireless node, distinguished between the original NICE and NICE-WLI. Analysis with VLC [3] has shown that those qualities are connected to characteristic packet rates in data transfer (packets per second, approximated by PPS). We assume the video source to be a fixed overlay member node. In NICE, whenever a data packet has to be disseminated in the overlay (especially in the wireless domain), the packet has to be unicasted to every member node, allocating the shared medium. In NICE-WLI, the allocation is limited by broadcasting every data packet, reaching all wireless nodes with one transmission. We abstract from potential collisions and retransmissions in MAC layer. Clearly visible, the number of medium allocations in the WiFi domain is only a fractional amount of what it would be setting aside underlay awareness. While the evaluative estimation here is based on analytical considerations, we were also able to verify it based on simulations. For this purpose we implemented NICE and NICE-WLI in OverSim [4], an overlay simulation environment.

5 Conclusion and Outlook

In this paper we presented NICE-WLI, an extension for NICE to efficiently support overlay members in wireless domains. NICE-WLI preserves the wireless medium in such domains by forbidding cluster leader roles for wireless nodes while limiting protocol messaging and data dissemination overhead. This is promising, as more and more wireless domains arise, allowing connection to the Internet and its services, including application-layer solutions at a global scale. In the future we plan to evaluate the benefits in more detail, including exact MAC layer considerations and media allocation savings in cluster leadership prescription. Also, we look at how to minimize the drawbacks that come up with broadcasting (e.g. higher loss ratios in wireless environments).

This work is supported by the Landesstiftung Baden-Württemberg within the SpoVNet Project and by the "Concept for the Future" of the Karlsruhe Institute of Technology within the framework of the German Excellence Initiative. Additionally, we would like to thank Daniel Schmidt for his ideas and collaboration.

References

[1] Banerjee, S., Bhattacharjee, B., Kommareddy, G.: Scalable Application Layer Multicast. *Proc. SIGCOMM 2002*, Pittsburgh, PA, USA, 205–217, 2002.

[2] Bloedt, S.: Efficient End System Multicast for Mobile Ad Hoc Networks. *Proc. 2nd IEEE Annual Conf. on Pervasive Computing and Communications Workshops*, Orlando, FL, 75, 2004.

[3] VLC media player - The Cross-Platform Media Player and Streaming server. http://www.videolan.org/vlc/. September 2008.

[4] OverSim: The Overlay Simulation Framework. http://www.oversim.org/. September 2008.

[5] Hosseini, M., Ahmed, D.T., Shirmohammadi, S., Georganas, N.D.: A Survey of Application-Layer Multicast Protocols. *IEEE Communications Surveys & Tutorials* 9(3),58–74, 2007.

[6] Chu, Y., Rao, S., Seshan, S., and Zhang, H.: A case for end system multicast. *IEEE Journal on Selected Areas in Communications* 20(8), 1456–1471, 2002.

[7] Fu, X., Hogrefe, D., and Lei, J,: DMMP: A New Dynamic Mesh-based Overlay Multicast Protocol Framework. *Proc. 4th IEEE Consumer Communications and Networking Conf. (CCNC 2007)*, Las Vegas, NV, 1001–1006, 2007.

[8] Strufe, T., Wildhagen, J., and Schäfer, G.: Towards the construction of Attack Resistant and Efficient Overlay Streaming Topologies. *Electronic Notes in Theoretical Computer Science* 179, 111–121, 2007.

[9] Brogle, M., Milic, D., Braun, T.: Supporting IP Multicast Streaming Using Overlay Networks. *Proc. Int. Conf. on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine 2007)*, Vancouver, Canada, 2007.