

On the Scalability of Carrier-grade Mesh Network Architectures

Johannes LESSMANN¹, Antonio DE LA OLIVA², Cigdem SENGUL³,
Andrés GARCÍA², Mathias KRETSCHMER⁴, Sean MURPHY⁵, Paul Patras^{2,6}

¹NEC Laboratories Europe, Network Research Division, 69115 Heidelberg, Germany

²Universidad Carlos III de Madrid, 28911, Leganés, Madrid, Spain

³Deutsche Telekom Laboratories, D-10587, Berlin, Germany

⁴Fraunhofer FOKUS, D-10589, Berlin, Germany

⁵University College Dublin, Belfield, Dublin 4, Ireland

⁶Institute IMDEA Networks, 28918, Leganés, Madrid, Spain*

Abstract: The increasing popularity of multi-hop wireless mesh networks, augmented with specific mechanisms required to support carrier-grade services, makes them an attractive alternative to classical backhaul solutions for network operators. However, current mesh deployments are typically small-sized, and, under real-time service requirements, it is yet unclear if they could scale to realistic network sizes. In this paper we present a carrier-grade mesh network architecture and conduct a thorough study of its scalability. In particular, we identify the main bottleneck problems and evaluate by means of simulations and practical experiments the costs induced by the modules that provide self-configuration, resource handling, routing and mobility support capabilities. The obtained results confirm that the control overhead of the proposed modules is low enough to permit mesh deployments spanning a network size of 100 nodes. Furthermore, our architecture and all proposed modules support dividing the whole network into sub-domains, which allows operating even larger networks.

Keywords: wireless mesh networks; carrier-grade; scalability

1. Introduction

With the decreased cost of wireless devices and their increasing flexibility in management and deployment, heterogeneous wireless mesh networks (WMNs) have become an interesting candidate solution for providing Internet access to mobile clients. Several multi-hop deployments in university campuses, hotels or airports have already proved to support applications without stringent QoS requirements [1, 2], while several works have proposed architectures that address problems such as overhead minimization, optimal channel assignment, throughput optimization, etc. [3, 4, 5]. However, in order to persuade network operators that WMNs could provide a feasible alternative to classical wired backhaul solutions, WMNs need to be enhanced to support the carrier-grade requirements imposed by real-time applications. Under this scenario, which involves more complex resource management, routing and mobility mechanisms, an inherent question that arises is whether carrier-grade mesh networks will scale with the network size.

In this paper, we present the case study of a pilot carrier-grade mesh network architecture that, in contrast to previous proposals, has been specifically designed to support the QoS requirements of real size deployments and conduct a detailed analysis of the scalability of the involved functionalities necessary to fulfill these requirements. Our architecture enables WMNs comprised of heterogeneous radio technologies, hiding many

*The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement N° 214994 (CARMEN).

of the complexities of the radio interface and making it easier to develop solutions for self-configuration, routing and network management, with the goal of delivering carrier-grade services.

We identify the bottlenecks inherent to mesh topologies and we undertake thorough simulation- and experiment-driven analyses of the modules involved by such architectures, specifically the adopted self-configuration, resource management, routing and mobility solutions. Our results show that managing networks with approximately 100 nodes does not pose scalability issues on any of the aforementioned modules. For larger topologies the proposed mechanisms support dividing the network into smaller, independently managed subnetworks with no practical impact on the overall performance.

2. Carrier-grade Mesh Network Architecture

In this section we provide a high level overview of the envisioned network architecture from a functional perspective, summarizing the interfaces and interactions between the modules comprising the designed system. The proposed architecture provides an integrated solution to enable wireless mesh networks comprised of heterogeneous radio technologies to deliver carrier-grade services.

The global architecture is depicted in Fig. 1. The main components are organized around an *Abstract Interface* that hides technology-specific aspects from higher level functions by providing them a unified and technology-independent API. The key modules required to support carrier-grade service in a mesh network include the following:

- **MAC Adapters (MAd)** implement the functionality required by the upper layer modules in a technology specific manner, providing support for interface setup and traffic handling. Nodes have one MAd per radio interface, which communicate with upper modules using the Abstract Interface primitives provided by the IMF.
- **Monitoring System** supplies other modules with timely information regarding the status of network in both technology-dependent and technology independent manner. It consists of three submodules: MeM (Measurement Module), MoMa (Monitoring Module aggregator), and MoMs (Monitoring Module storage), each of them being designed for different operation time-scale.
- **Interface Management Function (IMF)** coordinates the message flow between higher layer modules and the radio interfaces, and between local and remote nodes. IMF adopts the IEEE 802.21 architecture [6], but enables heterogeneous wireless network management in addition to supporting handovers between heterogeneous technologies.
- **Routing Function (RtF)** ensures connectivity between the APs and the gateways. RtF is also responsible for the overall capacity management within the mesh. In the proposed architecture, RtF is centralized, i.e., all link state information of the whole mesh is gathered and available in a single path computation element, that can therefore provide more optimal routing decisions.
- **Capacity Handling Function (CHF)** handles admission control and flow shaping in the access part, and reserves per aggregate capacity within the mesh that will be used afterwards to accommodate flows of the same traffic class.
- **Mobility Management Function (MMF)** is mainly responsible for maintaining connectivity of User Terminals (UTs) moving within the mesh network, while preserving associated QoS for the UTs associated flows.

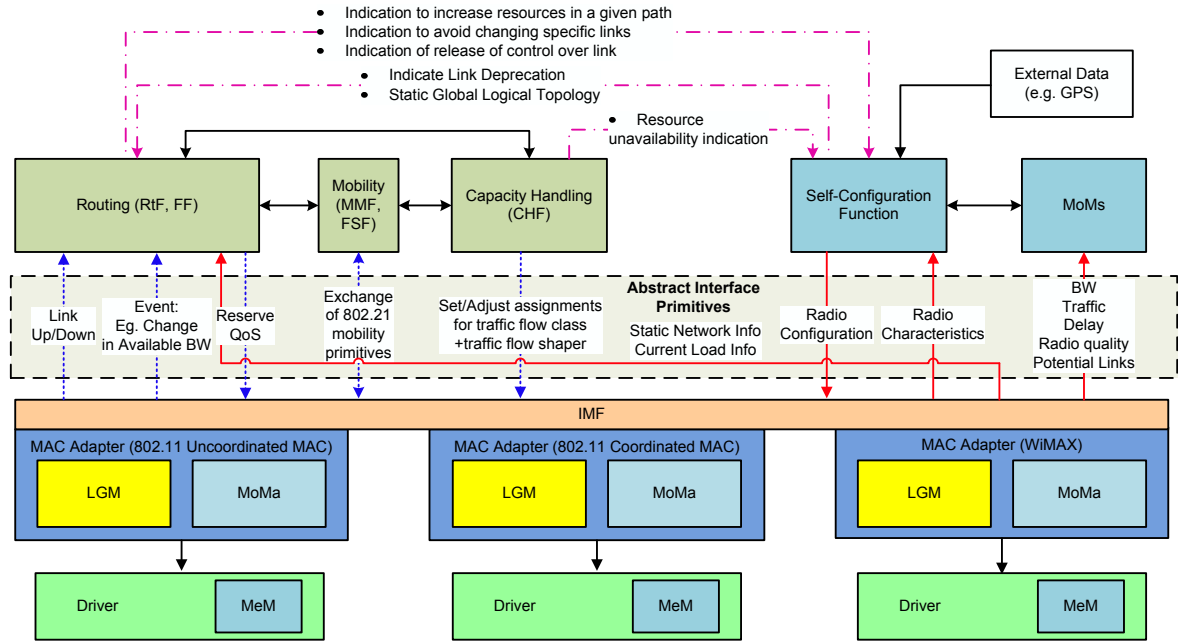


Figure 1: Functional Architecture

- **Self-Configuration Function (SCF)** provides the intelligence to autonomously perform network setup and configuration, especially node discovery and initial topology formation, and optimization during regular operation as well as during network failure. Like RtF, SCF takes a centralized approach.

3. Bottleneck Problems

The proposed architecture defines a centralized control by gateway nodes within the mesh network. This might raise certain scalability and bottleneck issues, which we address in this section. We consider two main use-cases: communications in emergency scenarios and backhauls for mobile networks. In the first case, a significant share of user traffic is sent between different nodes within the mesh itself. In the second case, the majority of user traffic flows between a node in the mesh and an external network, such as the Internet, therefore traffic concentrates around the mesh gateway, which implies that the latter can become a capacity bottleneck.

To prevent capacity bottlenecks, multiple gateways can be deployed based on a gateway placement algorithm that takes into account mesh routers, clients, gateway locations and the traffic demands from the clients [5]. Our proposed mesh architecture indeed allows for multiple gateway and, conceptually, there is no architectural constraint on either the number of possible gateways or the number of SCF, RtF and MMF modules. The algorithms run by SCF explicitly assume the presence of multiple gateways when bootstrapping the network. RtF, while having only one logical instance of a path computation element (called PiCE), also supports multiple gateways. All path setup related primitives allow specifying arbitrary source and destination nodes, i.e., any desired gateway or even no gateway at all for intra-mesh traffic. MMF also supports multiple gateways. UTs can be associated to different mobile access gateways (MAGs) located on different gateways and even inter-gateway handover is supported

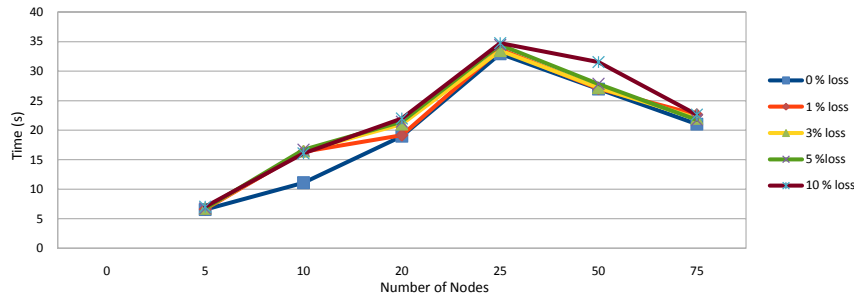


Figure 2: SCF Scalability Analysis

via proxy mobile IP (PMIP) mobility management in the core network.

4. Scalability Analysis

Scalability problems mostly occur whenever some functionality is centralized instead of decentralized. On the other hand, decentralized approaches may sacrifice optimality for the sake of scalability. A good solution for carrier-grade mesh networks must find a reasonable balance between these two goals. In the following we present the results of the conducted scalability study and insights gained from the analyses we carried out both in real networks, and by means of simulations with emulated wireless links. The experiments conducted in real networks were performed on unoccupied channels to minimize the interference with other networks, while the simulations did not particularly consider the impact of interference, since our main goal was rather to determine if our mechanisms can handle large number of nodes even under high loss probabilities, which can be caused by various types of channel impairments.

4.1 Self-Configuration Scalability

In order to evaluate SCF scalability for varying network sizes, we turned to a test network in which the physical wireless links are emulated. On top of the link emulator, the actual modules of the proposed architecture are running. We evaluate the scalability of the centralized SCF approach depending on the number of nodes to be configured and the stability of the links between the nodes. This evaluation has been performed using the SENF NetEMU¹ real-time network emulation component, which does not focus on a particular MAC protocol but can emulate links with different delays and loss rates, and upon which the IMF is implemented. NetEMU can be configured with different topologies wrt. the total number of nodes, the number and type of radio interfaces per node, as well as the geographical position of the nodes.

In order to evaluate the SCF scalability, randomly generated topologies consisting of 5, 10, 15, 20, 25, 50 and 75 nodes have been used. For each case the Topology Discovery Function component of SCF was run one hundred times with 0%, 1%, 3%, 5% and 10% loss on each link. Due to the ring-based approach (first direct neighbors are configured, then one-hop neighbors, followed by two-hop neighbors), the total time to detect and configure a scenario rather depends on the max hop-distance between the nodes and the gateway instead of the total number of nodes. The generated topologies comprised

¹<http://senf.berlios.de>

paths of lengths 1, 3 and 5 hops, while the node degree varied between 1 and 6 nodes. Note that the examined scenarios were generated randomly and a very favorable node placement may have a significant impact on resilience towards packet loss.

Fig. 4.1 depicts the total time required for a gateway to join all nodes in a given topology for a varying number of nodes and packet loss. This includes the configuration of the wireless interfaces of each node and the setup of a management pipe² between the gateway and each node. The emulated scenario does not include inter-radio interference measurements, but rather focuses on signaling load and latency measurements.

4.2 Resource Management Scalability

Our WMN architecture introduces a novel concept for managing resources called Link Group. We say that there is a link between two interfaces if these two interfaces operate on the same channel and can establish direct communication. We further define a link group as the set of such links. We focus on a scenario where the channels are carefully assigned in order to avoid undesired interferences. This is typical in planned networks, such as an operator-owned network where channels are centrally assigned in a way that overall interference in the network is minimized. In our system, SCF configures all the interfaces within a Link Group, including the modulation rates of the links, which are chosen as the highest possible value, considering the channel conditions. Once this configuration has been set, we compute the *production cost* (PC) of a link as the time an interface will be using the channel to transmit a frame with the configured modulation rate, related to the time it would have used the channel if it was configured with the maximum rate. Once the bootstrap process finishes, a set of metrics (e.g., frame loss) are measured on links and used to update the PCs. Knowing the link PCs, the effective available capacity in a link group, is computed as the difference between the maximum achievable capacity and the sum of the cost and employed rate products of all the links.

The routing scheme we employ relies on the link group parameters (production cost and overall capacity) for each link group in the network. This computation has to be performed at boot time, when nodes are configured, in order to let the routing module compute the best paths. Since PCs already account for typical small-scale link variations, a new computation has to be triggered only in case of severe degradations, which are reported by MoMa, which constantly monitors the link conditions.

Computation time for uncoordinated MAC: To analyze the impact and to identify potential bottleneck from this computation, we prototyped in Matlab a non-optimized version of the algorithm to compute link group capacity and costs. Then, we measured the time required to compute them, for a varying number of N nodes in the link group –note that in a link group with N nodes, there are $N \times (N - 1)$ links to consider. Fig. 3(a) shows the time required to compute the link group model as a function of the number of nodes that must be considered. We observe that for link groups of realistic sizes, this computation should not impact network operation. Indeed, even for the case of 10 nodes the computational time will be around 1s, which does not constitute an excessive burden for the bootstrapping process.

Upon a new allocation request received by the Link Group Manager (LGM) of an uncoordinated MAC scheme, e.g., IEEE 802.11 EDCA [7], based on the information gathered by the measurement modules and an analytical model of the link group performance, the LGM computes the new set of MAC parameters that should be used within

²We refer with pipe to a reserved end-to-end allocation of a traffic aggregate.

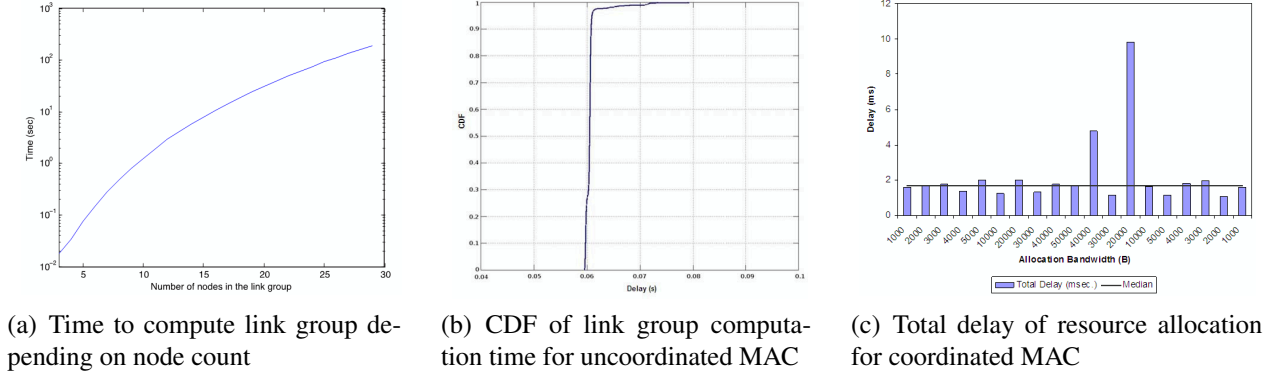


Figure 3: Resource Management Scalability Analysis

the link group. To do so, the LGM sweeps a range of transmission probability values related to the requested rates and for each of the different traffic classes assigned. The theoretical global throughput achieved for each of the previous values is then calculated, and the maximum point is searched using a golden search algorithm. The computation of this new configuration could result in a non-negligible computational cost, given the limitations imposed by the devices used in a real deployment.

To evaluate the computational complexity, we performed the following experiment. We installed inexpensive COTS devices, namely Soekris net4826 boxes with two Atheros-based wireless interfaces and we issued a request for a resource reservation, corresponding to a WLAN scenario with 10 pipes installed. We measured the elapsed time until the new set of parameters was distributed to the nodes. In Fig. 3(b) we plot the sCDF of this computation time, obtained after repeating the measurement 4500 times. As the result shows, the time required for such operations is approximately 60 ms, and rarely above 80 ms. Therefore, we conclude that even using inexpensive COTS devices the computation of the EDCA parameters is achievable within affordable time.

Computation time for coordinated MAC: Next, we evaluate resource allocation delay with a coordinated MAC scheme, i.e. SoftToken TDMA[8], where the LGM uses a token-based scheme over WiFi MAC to grant access to channel. A node can only access the channel when it has the token, and as much as it is allowed to transmit. The node can request for additional resources when returning the token. Note that the node learns whether its request has been granted or not in the next token cycle. To understand the resource allocation of this scheme, we conducted experiments to assess the time required to serve allocation requests. Essentially understanding these delays helps evaluate how fast the changes can propagate within a link group. The link group consisted of 2 nodes employing IEEE 802.11a PHY, and configured in ad-hoc mode on 5.3 GHz carrier frequency and 54 Mbps transmission rate. The allocation request starts with an initial value of 1000 B, then it is increased up to 50000 B in 10 steps, and finally reduced back to 1000 B in 10 steps. The delay is calculated as the time difference between a resource allocation request and the corresponding response.

In Fig. 3(c) we plot the time required for Coordinated MAC in the user space to get a response back from SoftToken, which resides in the kernel space, plus the one hop communication delay between the node initiating the request and the LGM. The results show that the median delay is around 2.18 ms and the maximum observed delay is around 9 ms. The main reason behind the high peaks is token losses, which delay

the response until the next token cycle. However, as SoftToken generates tokens every 2 ms, when there is no traffic to poll the slaves about their traffic requests, we believe that the delay is dominated by this SoftToken token generation period.

4.3 Routing Scalability

Several routing algorithms have been implemented and evaluated searching for the perfect balance between complexity and optimal path placing. The result of this research has been a set of routing algorithms with different characteristics, all of them using the Link Group Parameters (production cost and overall link group capacity) as metrics.³

The schemes evaluated correspond to a Multi-Path algorithm, which solves a linear optimization problem to find the optimum routing, but with the particularity of splitting the flows through several paths. We also consider the Single-Path algorithm, which, through solving an Integer Linear Problem, is able to find a good approximation to the optimal while keeping each flow as an atomic entity. Finally the Widest Path algorithm uses the Link Group Parameters as metric to solve a Widest Shortest Path problem.

To evaluate the performance of the Multi-Path algorithm, we performed a set of several tests with random topologies of 100 nodes in order to measure the time required to compute a pipe from any node to the gateway. Fig. 4(a) shows the results obtained from these experiments. The x axis corresponds to the number of paths computed in a certain execution of the algorithm. The code used to compute the paths is the actual code used in the implementation that is programmed completely in C and optimized for fast execution. As can be seen in the figure, the time required for computing a maximum of 100 paths simultaneously falls below 45ms. Note that this result is also significant for rerouting of pipes in cases of link failures. In such cases, multiple pipes might be affected and must be rerouted at the same time. Our results indicate that this should not cause scalability concerns even for a large number of affected pipes.

4.4 Mobility Management Scalability

Our architecture adopts a two-level hierarchy to provide UTs with mobility management. On one hand, PMIPv6 [9] is used as mobility management solution to enable users to move between different gateways (that play the role of PMIPv6 MAGs). On the other hand, a PMIPv6-alike solution is used within the mesh network, allowing UTs to roam across APs associated to the same GW, with a per-flow granularity. These movements are transparent to the PMIPv6 entities of the upper level.

For the conducted scalability analysis we considered a worst case scenario of having one pipe per flow and assessed the PMIPv6 signaling overhead within the mesh. Fig. 4(b) shows the results of our analysis of the signaling overhead (load) within the mesh network, for different scenarios. On one hand, we analyze UTs that have a range of 1 to 10 active flows. Besides, different handover frequencies have been studied. We note here that since the signaling depends - for a given number of flows being handed off - on the number of handovers, from an overhead perspective having two UTs moving each minute is equivalent to having one UT moving each 30 seconds. Therefore, using the frequency of handovers as parameter allows us to evaluate how the scalability of the mobility management solution is affected by large numbers of UTs managed by the same CGW, and how it is affected by highly mobile UTs. The obtained results show that even for extreme scenarios the overall mobility signaling load is below 100kbps.

³Due to space constraints we omit the details of the set of routing algorithms studied.

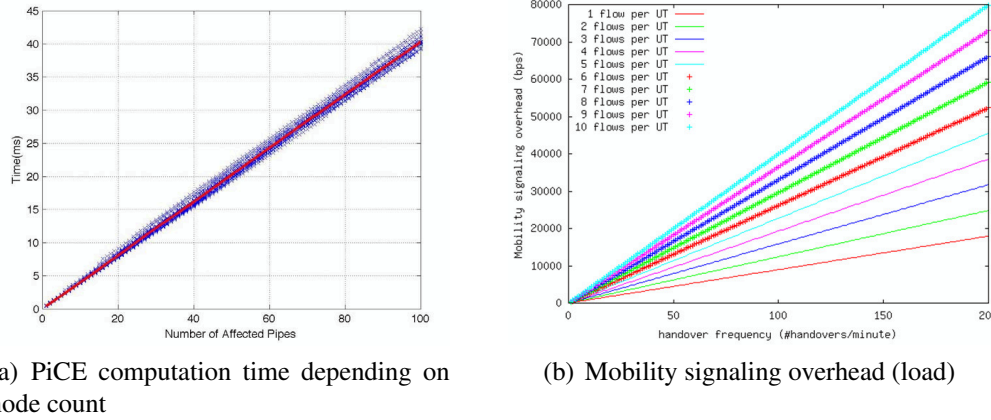


Figure 4: Mobility Management Scalability Analysis

5. Conclusions

In this paper we presented a scalability study of the mechanisms involved in a WMN supporting carrier-grade services. The key findings of our evaluation are summarized as follows: *(i)* self-configuration scalability is not critical, since this process primarily runs at network boot time, at which point it lacks real-time requirements, *(ii)* from a routing perspective, managing networks with about 100 nodes is not problematic, the solution being fast enough for all practical purposes, *(iii)* from a mobility management viewpoint, a network of 100 nodes should also pose no issue, even at rates of 2 handovers per node per minute, and *(iv)* from a resource management perspective the link group computation times are extremely small and hence do not pose major implications on the maximum practical network size. Additionally, we remark that the envisioned architecture and mechanisms support dividing the whole network into smaller independently managed subnetworks, therefore scaling to any practical network size.

References

- [1] Y. Takahashi, Y. Owada, H. Okada, and K. Mase, "A wireless mesh network testbed in rural mountain areas," in *Proc. ACM WinTECH*, pp. 91–92, 2007.
- [2] T. He, S. Chan, and C. Wong, "Homemesh: a low-cost indoor wireless mesh for home networking," *Communications Magazine, IEEE*, vol. 46, pp. 79–85, Dec. 2008.
- [3] B. Wehbi, A. Laouiti, and A. Cavalli, "A reactive wireless mesh network architecture," in *Advances in Ad Hoc Networking*, vol. 265 of *International Federation for Information Processing*, pp. 203–214, 2008.
- [4] A. Raniwala and T. cker Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. IEEE INFOCOM*, pp. 2223–2234, August 2005.
- [5] P. Zhou, X. Wang, B. S. Manoj, and R. Rao, "On optimizing gateway placement for throughput in wireless mesh networks," *EURASIP Journal on Wireless Communications and Networking*, 2010.
- [6] IEEE 802.21, "Media Independent Handover Services," 2009.
- [7] IEEE 802.11, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*. Revision of Std. 802.11-1999, 2007.
- [8] P. Djukic and P. Mohapatra, "Soft-tdmac: A software tdma-based mac over commodity 802.11 hardware," in *Proc. IEEE INFOCOM*, pp. 1836–1844, 2009.
- [9] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil, "Proxy Mobile IPv6." RFC 5213 (Proposed Standard), Aug. 2008.