A Queuing Modeling Approach for Load-Aware Route Selection in Heterogeneous Mesh Networks*

Raffaele Bruno, Marco Conti, Antonio Pinizzotto
IIT - CNR
Via G. Moruzzi, 1 - 56124 Pisa, Italy
{*r.bruno,m.conti,a.pinizzotto*}@*iit.cnr.it*

Abstract

Wireless mesh networks are primarily used to provide Internet access by sharing the Internet connections of a limited number of gateways. If traffic is routed in the mesh without considering load distribution, unevenly network congestion may arise and some gateways may rapidly get overloaded, which causes a significant reduction of the network capacity. To address this issue, in this paper we firstly develop a queuing network model that accurately predicts the residual capacity of paths in heterogeneous mesh networks, and precisely identify network bottlenecks. By taking advantage of this model, we design a novel Load-Aware Route Selection algorithm, named LARS, which improves the network capacity. This objective is obtained by allowing each mesh node to distribute the traffic load among multiple gateways in order to ensure evenly utilization of Internet connections. Simulation results show that LARS significantly outperforms shortest path routing using contention-aware link costs, achieving throughput improvements of up to 210% *in the considered network scenarios.*

1. Introduction

802.11-based wireless mesh networks are emerging as a key technology to provide cost-effective ubiquitous access to the Internet [12]. Normally, in mesh networks only a subset of routers, referred to as *gateways*, has a high-speed Internet connection, while Internet access is shared among all the other mesh nodes by exploiting the ad hoc routing capabilities of the mesh routers [3]. However, this vision is rapidly changing. Real-world mesh networks have been recently deployed, which are used to share a potentially large number of low-speed Internet connections (i.e., DSL fixed lines) available at the customers' premises. Examples of such networks are Meraki-based deployments in urban areas [15], or the Ozone's network in Paris, which is composed of 400 mesh routers, most of them using standard DSL links as Internet backhaul, while only ten gateways are provided with an ISP-owned fiber link [16]. In a broader sense, wireless mesh networks are evolving into a *converged infrastructure* used to share the Internet connectivity of sparsely deployed fixed lines with *heterogeneous capacity*, ranging from ISP-owned broadband links to subscriber-owned low-speed connections [18].

Being mesh networks primarily used for Internet access, both traffic routing and Internet gateway selection play a crucial role in determining the overall network performance, and in ensuring the optimal utilization of the mesh infrastructure [21]. For instance, if too many mesh nodes select the same gateway as egress point to the Internet, congestion may increase excessively on the wireless channel, or the Internet connection of the gateway can get overloaded. This is especially important for the heterogeneous mesh networks considered in this study, because low-speed Internet gateways may easily become a *bottleneck*, limiting the achievable capacity of the entire network. In addition a load-unaware gateway selection can lead to an unbalanced utilization of network resources.

Routing protocols in wireless mesh networks have been extensively studied in last years, but primarily focusing on the design of more efficient routing metrics. One of the first examples of such routing metrics is ETX [7], which exploits per-link frame loss rates to find paths with high throughput. Starting from the ETX concept, many variants have been proposed to capture, for instance, the effect of diverse transmission rates [2, 8], or intra-flow and inter-flow interference [10]. Although these metrics have been demonstrated to work very well in mesh networks, and to provide significantly higher throughput performance than simple hop count, they are completely unaware of the available resources at the gateways. On the contrary, significant performance improvements might be obtained by consider-

*This work was supported in part by the European Commission in the 7FP through project EU-MESH, ICT-215320, and by the Italian National Project WORLD, under grant number 2007R989S.

978-1-4244-4439-7/09/$25.00 ©2009 IEEE

ing residual capacity of gateways' Internet connections, as well as load distributions, when routing traffic flows. Some schemes for load balancing in wireless mesh networks proposed to restrict load balancing decisions only at the gateway side [6,17]. Other recent studies have explored the benefits of introducing load-dependent information in the design of routing heuristics. For instance, in [20] authors proposed the Interference-Aware Resource Usage (IRU) routing metric, which captures inter-flow interference (i.e., interference between neighboring nodes) to facilitate load-balanced routing. In [14] load balancing is incorporated in the routing metric by taking into account the average queue length of mesh routers. An approach more similar to our work is adopted in [1], which introduces a simple heuristic to determine the residual capacity of a node. However, there is a complex interdependence between the way traffic flows are routed in the network and the utilization of network resources, which makes very difficult to precisely estimate the remaining capacity of a path or a gateway.

To address the above problem, in this paper we make the following two main contributions. First of all, we develop a queuing-based model of a heterogeneous mesh network, which incorporates the interdependencies between random access MAC protocol, traffic routing and load distribution. This model is used to estimate the network capacity, and to identify network bottlenecks, due to either congestion on the wireless channels or overloading of fixed lines. Simulation results are shown to validate the accuracy of this model using shortest path routing algorithms. Then, we propose a novel *Load-Aware Route Selection* algorithm, named *LARS*, which integrates traffic routing with gateway selection. The goal of LARS is to improve the network capacity and to avoid underutilization of gateways' resources. The idea behind the LARS design is to allow each mesh node to distribute the traffic load among multiple gateways to ensure evenly utilization of Internet connections. To this end, mesh nodes select the routes towards the gateways taking into account the residual capacity of the paths, and the utilization of the gateways' fixed lines. We exploit the proposed queuing model to accurately predict the residual capacity of each path, and to discard paths or gateways that cannot accept additional demands. Simulations performed with various network configurations show that the proposed route selection algorithm significantly outperforms shortest path routing using contention-aware link costs [10], providing up to 210% throughput improvement in some network scenarios.

The remaining of this paper is organized as follows. Section 2 introduces the system model and develops the capacity analysis. Section 3 describes the LARS algorithm. In Section 4 we validate the analysis and present a simulation-based comparison of LARS against shortest path routing algorithms. Finally, conclusions and future extensions are discussed in Section 5.

2. Network capacity analysis

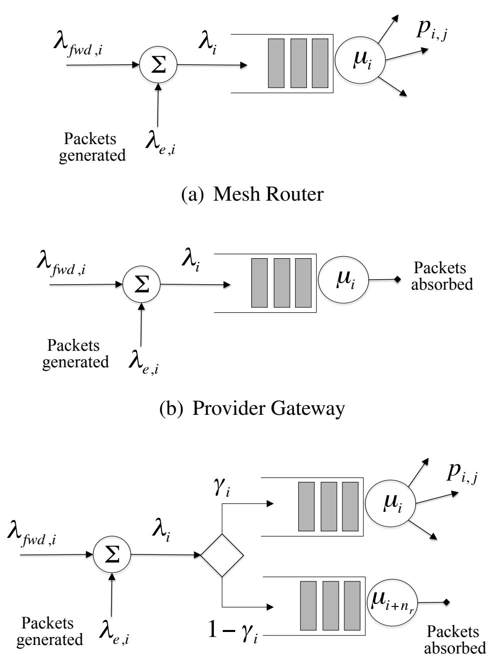
In this section we describe a queuing model for a heterogeneous mesh network using a random access MAC protocol, which is used to estimate the network capacity.

2.1. The network model

As discussed in Section 1, we consider an heterogeneous mesh network consisting of three classes of nodes: mesh routers, provider gateways and residential gateways. Each class of nodes supports different communication services, and uses wireless and/or wired communications technologies. To model wireless communications we adopted the Protocol Model [11]. Thus, transmission range r_{tx} of each station is fixed, and all stations transmit at the same power. Moreover, a wireless transmission between neighboring nodes i and j is successful only if $d_{k,j} > (1+\Delta) \cdot r_{tx}$ for any other node k that transmits simultaneously with node i. The constant $\Delta > 0$ is a parameter taking into account the interference level.

In this study we are concerned with *uplink* traffic, i.e., traffic generated by the mesh clients for hosts/servers in the Internet. Each mesh node receives packets from the mesh clients associated to it. In the considered scenario, packet generated by mesh clients can be forwarded to any of the available Internet gateways. We assume that the inter-arrival time of the external traffic arriving at mesh node i is exponentially distributed with parameter $1/\lambda_{e,i}$, which implies that the packet generation process is a Poisson process. However, as we will explain in Section 2.2, the analysis is applicable to any traffic model, whose inter-arrival distribution is a renewal process. Regarding the modeling of the forwarding strategy implemented by mesh nodes, we use a *forwarding matrix* $\mathcal{P} = \{p_{i,j}\}$, where $p_{i,j}$ is the probability that a packet received by mesh node i is forwarded to mesh node j. In this work we analyze routing strategies that lead to state-independent forwarding matrices (i.e., with constant forwarding probabilities).

To model the mesh network as a queuing network it is necessary to characterize each mesh node with a correspondent queuing system, which should capture the most important aspects of the queuing and forwarding processes. Since we consider mesh networks composed of three types of mesh nodes, we have introduced three different classes of stations in our equivalent queuing network, which are shown in Figure 1. First of all, let us assume that in the mesh network there are n_O mesh routers, n_P provider gateways, and n_R residential gateways, being $n = n_O + n_P + n_R$ the total number of nodes in the mesh network. Each mesh router will receive user-generated traffic with an average rate $\lambda_{e,i}$, and packets from the other stations of the queuing network with an average rate $\lambda_{fwd,i}$. Hence, the resulting *effective* *arrival rate* λ_i at station i is equal to $\lambda_{fwd,i}+\lambda_{e,i}$. Since the mesh routers are not connected to the Internet, they behave as pure relay stations forwarding the packets they received to their neighbors according to the pre-computed forwarding probabilities. We assume that the transmission rate of each mesh router is equal to W_O bps, and the packet size is constant and equal to L bits. However, the time needed by station i to serve the packet is generally greater that the packet transmission time (L/W_O), and depends of the number of its interfering neighbors. In Section 2.2.1 we present a stochastic model to determine the mean and second moment of the service times when the stations in the queuing network use a random access MAC protocol to coordinate their transmissions. The queuing representation of the



(a) Mesh Router

(b) Provider Gateway

(c) Residential Gateway

Figure 1. Representation of the nodes of the mesh network as equivalent stations in the queuing network.

provider gateway is shown in Figure 1(b). The primary difference between mesh routers and provider gateways is that the latter are connected to the Internet with a fixed access line of bandwidth W_P bps, with $W_P >> W_O$. Note that the Internet access line is a point-to-point connection. As a consequence, the time taken by the provider gateway to serve the packet is constant and equal to L/W_P seconds. Moreover, being the final destination of the packet, the gateway *absorbs* the packet, which departs from the queuing network.

Similarly to provider gateways, residential gateways have the primary purpose of providing Internet access, but their queuing representation is different and more complicated. Specifically, let us denote with W_R the bandwidth, in bits per second, of the fixed access between the residential gateway and the Internet. In this case, the Internet access line of the residential gateway may rapidly become a bottleneck as the traffic received on the wireless interface builds up, limiting the achievable capacity of the whole mesh network. To make this limitation less severe, the residential gateway may take advantage of the available wireless bandwidth to behave as a relay node, and further forwarding the traffic to one of its neighbors, which may be less congested, or closer to a provider gateway. As shown in Figure 1(c), this can be modeled using two separate queues. Then, when a residential gateway i receives a packet, it decides to absorb the packet with probability γ_i, or to forward the packet to one of its neighbors with probability $1-\gamma_i$. The design of the γ_i function is influenced by the routing and resource allocation strategies implemented in the mesh network.

In summary, our model of a mesh network of n nodes consists of an equivalent $G/G/1$ queuing network of n stations with $n+n_R$ queues, in total. The connections between these queues is determined by the forwarding matrix. Finally, we assume that each queue in the network has infinite size, and the packets are served according to a FCFS discipline.

2.2. Diffusion approximation for an open queuing network

It is intuitive to note that, even assuming a Poisson model for the external packet generation process at each station of the equivalent queuing system, both service times and overall packet inter-arrival times are generally not exponentially distributed. Unfortunately, the problem of deriving closed-form expressions for the state probabilities of a $G/G/1$ queuing network is generally mathematically intractable. Nevertheless, using the diffusion approximation as proposed in [13], we can obtain an approximate solution for the joint distribution of the queue lengths. For the sake of clarity, in the following we review the main results of this method.

The underlying assumptions of the diffusion approximation method are the following. First of all, the external arrival process is a renewal process with mean inter-arrival time $1/\lambda_e$ and coefficient of variation c_E. Secondly, the service times at queue i have an arbitrary distribution with known mean $1/\mu_i$ and coefficient of variation c_{B_i}. Finally, all queues in the network are single server with FCFS service strategy. By applying classical reasoning on the statistical equilibrium between the rate of departures from a queue and the rate of arrivals, the *effective* arrival rate of packets at queue i can be computed as

$$\lambda_i = \lambda_e \cdot p_{e,i} + \sum_{j=1}^{m} p_{j,i} \cdot \lambda_j \ , \qquad (1)$$

where $p_{e,i}$ is the probability that an external packet first enters the network at queue i (i.e., $\lambda_{e,i} = p_{e,i} \cdot \lambda_e$), and m the number of queues in the network. By definition, the *relative* arrival rate at queue i, hereafter denoted with *visit ratio* e_i, is equal to λ_i/λ_e, and it represents the average number of times a packet is forwarded by the station i. Finally, the utilization factor ρ_i for queue i is simply λ_i/μ_i.

According to [13] the approximated joint distribution of queue lengths have a product-form solution with marginal probabilities

$$\hat{\pi}_i(k) = \begin{cases} 1-\rho_i & k=0 \\ \rho i(1-\hat{\rho}_i)\hat{\rho}_i^{k-1} & k>0 \end{cases} , \qquad (2)$$

where $\hat{\rho}_i$ is a correction factor for the queue utilization computed as follow

$$\hat{\rho}_i = exp\left(\frac{2(1-\rho_i)}{c^2_{B_i} + \rho_i \cdot c^2_{A_i}}\right) \ . \qquad (3)$$

In Equation (3), the parameter c_{A_i} denotes the coefficient of variation of the inter-arrival times at node i. This term depends on the complex interplay between the service processes of other stations in the queuing network. In [13] the square c_{A_i} value is approximated using the following expression

$$c^2_{A_i} = 1 + \sum_{j=0}^{m}(c^2_{B_i} - 1) \cdot p^2_{j,i}\frac{e_j}{e_i} \ , \qquad (4)$$

where we set $c^2_{B_0} = c^2_E$.

2.2.1 Analysis of service times

In this section we derive analytical expressions for μ_i and c_{B_i} values, which are necessary to compute the queue utilization factor and queue length distributions, respectively. For the sake of notation brevity, we introduce three sets of integers as follows:

$$I_O = \{1, \ldots, n_O\}$$
$$I_P = \{n_O + 1, \ldots, n_O + n_P\}$$
$$I_R = \{n_O + n_P + 1, \ldots, n\} \ .$$

Then, in the following development we assume that the station i in the queuing network has the structure shown in Figure 1(a) for $i \in I_O$, the structure shown in Figure 1(b) for $i \in I_P$, and the structure shown in Figure 1(c) for $i \in I_R$. This permits to easily associate the μ_i parameter to the service time of a corresponding node in the mesh network.

As discussed in Section 2.1, the queues modeling Internet access lines are characterized by a constant service time, which depends only on the packet size and the fixed channel bandwidth. Consequently, it is easy to derive that $\mu_i = W_P/L$ for $i \in I_P$, while $\mu_{i+n_R} = W_R/L$ for $i \in I_R$ (see Figure 1(c)), and the corresponding coefficients of variation are equal to zero. On the contrary, the derivation of the service time distributions for the wireless buffers is more involved because it is necessary to take into account the collision avoidance mechanisms of the random access MAC protocol used to coordinate the packet transmissions. In our analysis we assume that the propagation delay is null, so that each station can have an instantaneous knowledge of the wireless channel status.

Based on the interference model introduced in Section 2.1, when a station has a packet to transmit on the wireless channel it is allowed to transmit only if none of the interfering neighbors of the stations are transmitting simultaneously (i.e., the channel is sensed idle), and its transmission will not interfere with nearby receivers. It is important to note that the number of active interfering stations is not constant, but it depends on the queues' utilization. To simplify the analysis, we assume that a station does not back off before transmitting, but it transmits as soon as there are not interfering or hidden stations, and the MAC protocol grants it with the permission to transmit. Note that the focus of our analysis is to model the effect of exposed and hidden station problems, which prevent station from sending packets to other nodes due to neighboring transmitters and receivers, rather than exactly modeling all the features of the IEEE 802.11 MAC protocol. Indeed, in multi-hop environments the location-dependent contention issues due to differences in the number of interfering neighbors can be the predominant source of the channel access inefficiency.

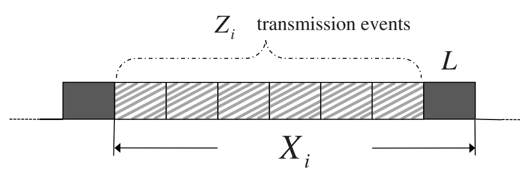


Figure 2. Evolution of service time process X_i between two consecutive successful transmission of station i.

To derive the service time distribution we consider a tagged wireless queue i (with $i \in I_O \cup I_R$), and analyze the events occurring during the time X_i needed to serve a packet (i.e., the time needed to successfully transmit the packet at the head of the transmission buffer), as shown in

Figure 2. Due to the contention process, station i may wait for Z_i transmissions from other interfering stations before being able to transmit its packet. Hence, the X_i period can be expressed as

$$X_i = \frac{L}{W_O}(1 + Z_i) \ . \tag{5}$$

To compute the X_i distribution we assume that, at the end of each transmission of any interfering node, station i transmits with a fixed and constant probability q_i, which takes into account that the MAC protocol randomly assigns a permission to transmit to one of the contending stations. With this assumption we have that $\Pr\{Z_i = k\} = (1 - q_i)^k q_i$. Note that $E[X_i] = 1/\mu_i$ and $c^2_{B_i} = (E[X_i^2] - E[X_i]^2) \cdot \mu_i^2$. Owing to the geometric assumption for the distribution of the Z_i process, it holds that

$$E[X_i] = \frac{1}{\mu_i} = \frac{L}{W_O} \cdot \frac{1}{q_i} \ , \tag{6a}$$

$$c^2_{B_i} = (1 - q_i) \ . \tag{6b}$$

To ease the derivation of the q_i expression we introduce the $q_{i,j}$ parameter, defined as the probability that, at the end of a transmission of an interfering neighbor, station i transmits a packet, given that this packet is for its neighbor j. Using the forwarding matrix, and applying the law of total probability, we can write

$$q_i = \sum_{j \in \mathcal{N}_i} p_{i,j} \cdot q_{i,j} \ , \tag{7}$$

where $\mathcal{N}_i$ is the set of neighboring wireless queues for station i. Formally, $\mathcal{N}_i = \{j : d_{i,j} \leq r_{tx} \ , \ j \in I_O \cup I_P \cup I_R\}$.

The following lemma provides an explicit expression for the $q_{i,j}$ probability.

Lemma 1 *In the considered queuing network system, under the assumption that reception and transmission events are mutually independent, it holds that*

$$\begin{aligned} q_{i,j} &= \alpha_{i,j} \cdot \beta_{i,j} \\ &= \prod_{h \in \mathcal{E}_i} (1 - \phi_h \cdot \eta_{h,j}) \cdot \prod_{k \in \mathcal{E}_j} (1 - \psi_k) \ , \end{aligned} \tag{8}$$

where

- ϕ_h *is the long-term fraction of time spent by station* h *receiving packets from its neighbors;*
- ψ_k *is the long-term fraction of time spent by station* k *forwarding packets to its neighbors;*
- $\eta_{h,j}$ *is the fraction of wireless queues that are neighbors of station h, but they are not interferers for station* j*;*
- $\mathcal{E}_j$ *is the set of interfering nodes for station* j *(formally,* $\mathcal{E}_j = \{k : d_{k,j} \leq (1+\Delta) r_{tx} \ , \ k \in I_O \cup I_R\}$*).*

Proof: Reported in [4] due to space limitations. □

It is worth pointing out that the utilization factor is a very important property of a queue, because it can be used to determine the set of traffic loads that are sustainable by a given routing policy. More precisely, our primary goal is to ensure that the network is *stable*. We provide the following formal definition of network stability.

Definition 1 *A queue is strongly stable if it has a bounded time-average number of packets in the system.*

Definition 2 *A network is strongly stable for a specific arrival process if all individual queues are strongly stable.*

In conclusion, to determine the *network capacity* under a specific routing policy, we will search for the minimum arrival rate λ_e that makes the network unstable, and the *network bottleneck* is the first queue to become unstable.

3. Route selection for improving network capacity

In this section we describe a *Load-Aware Route Selection* algorithm, named *LARS*, which aim at improving network capacity by avoiding underutilization of gateways' resources. The idea behind the design of the LARS algorithm is to allow each mesh node to distribute the traffic load among multiple gateways to ensure evenly utilization of Internet connections. Traditional shortest path routing algorithms enhanced with contention-aware link costs [1, 10] are not easily applicable to this problem because the link costs depend on the routing configuration itself. On the contrary, the key feature of our algorithm is to take advantage of the queuing model developed in Section 2 to *predict* if the assignment of additional traffic to a path will generate a bottleneck in the mesh network. If the network becomes unstable an alternate path to the same gateway, or to a different gateway, can be tried to satisfy the flow demands.

Note that estimating the residual capacity of all the possible paths towards all the available gateways has a complexity that increases exponentially with the number of mesh nodes. To reduce the algorithm complexity, LARS algorithm explores a path subset consisting of the least-cost paths between mesh nodes and available gateways. Thus, we assume that in parallel to our proposed route selection algorithm, a proactive link-state routing is running to build the set of available paths. Starting from this set, LARS decides to which available path to assign the new traffic flow so as to avoid the creation of network bottlenecks.

In general, the definition of the resource allocation problem may also change depending on the users' fairness models we want to enforce. In [9] various fairness models and sharing objective functions have been proposed for multi-hop wireless backhaul networks. In this work we are concerned with eliminating *spatial bias* [9] so as to ensure that nodes one hop away from a wired egress point do not receive a greater share of resources than the nodes multiple hops away. Furthermore, to maximize network capacity we consider a finite set of flows and we try to maximize the cumulated data rate by progressively increasing each flow until all gateways get congested.

Algorithm 1: Pseudo-code of the LARS algorithm.

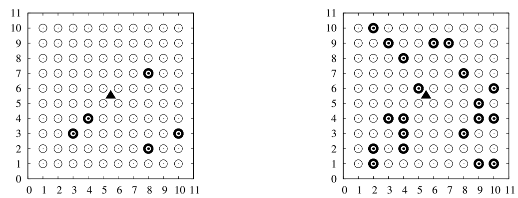
Input: A map $\mathcal{M}$ of the mesh network topology.
Output: The forwarding matrix.

1 stable $\leftarrow true$
2 $\mathcal{P} \leftarrow \{0\}$
3 **while** stable **do**
4   **foreach** s *in* A **do**
5     $\lambda_{e,s} \leftarrow \lambda_{e,s} + \Delta\lambda$
6     $\lambda_e \leftarrow \lambda_e + \Delta\lambda$
7   **foreach** i *in* $\mathcal{A}$ **do** $p_{e,i} \leftarrow \frac{\lambda_{e,i}}{\lambda_e}$
8   allocated $\leftarrow false$
9     $\mathcal{Q} \leftarrow \mathcal{G}_s$
10     $\mathcal{M}' \leftarrow \mathcal{M}$
11   **while** *(*$\mathcal{Q} \neq \emptyset$ **or** ! allocated*)* **do**
12       $\mathcal{P}' \leftarrow \mathcal{P}$
13       $g \leftarrow$ `ExtractMin(`$\mathcal{Q}$`)`
14       $path_{s,g} \leftarrow$ `ShortestPath(`$s, g, \mathcal{M}'$`)`
15   **foreach** *node* $k \in path_{s,g}/\{g\}$ **do**
16   **if** $k \neq s$ **then** $\lambda'_k \leftarrow \lambda_k + \Delta\lambda$
17   **foreach** *neighbor* r *of* k **do**
18           $m \leftarrow$ `Next(`$path_{s,g}, k$`)`
19   **if** $m = r$ **then** $\lambda'_{k,r} \leftarrow \lambda_{k,r} + \Delta\lambda$
20           $p'_{k,r} = \frac{\lambda'_{k,r}}{\lambda_k}$
21   **end**
22   **end**
23   **if** `IsStable(`$\lambda_e, p_e, \mathcal{P}'$`)` **then**
24   **foreach** $k \in \mathcal{A}$ **do** $\lambda_k = \lambda'_k$
25   allocated $\leftarrow true$
26         $\mathcal{P} \leftarrow \mathcal{P}'$
27   **else**
28   remove g's Internet connection from $\mathcal{M}'$
29   **end**
30   **end**
31   **if** $\mathcal{Q} = \emptyset$ **and** ! allocated **then**
32   stable $\leftarrow false$
33   **end**
34   **end**
35 **end**

Let $\mathcal{A}$ be the set of mesh nodes in the network, and $\mathcal{M}$ the mesh network topology. Let us assume that each mesh node i (i.e., both routers and gateways) is the source of an upstream Internet flow. Then, we denote with $\mathcal{G}_i$ the set of gateways that node i can use to access the Internet. In a classical single-path routing a mesh node ca use one Internet gateway, but *splitting* policies may be applied to allow a mesh node to use all, or a subset, of the available Internet gateways. For instance, residential gateways can become congested when the user traffic increases around them, and they may prefer to relay their traffic to other less congested gateways.

As explained above, the LARS algorithm tries to maximize the network capacity by progressively increasing each flow bandwidth by a constant factor $\Delta\lambda$, visiting the nodes in a round robin fashion so as to guarantee the same share of network capacity to each node s in the network. To some extent, this is equivalent to add a new *micro-flow* of bandwidth $\Delta\lambda$ to each node. Specifically, let s be the mesh node trying to activate a new micro-flow. First of all, we add the $\Delta\lambda$ demand to the arrival rate of the external traffic at node s (line 5) and we update accordingly the overall network load (line 6). Then, these values are used to update the probabilities $p_{e,i}$ that a packet entering the network from outside first enters the i^{th} node (line 7).

The second stage of LARS operations is the selection of the *best* gateway for s in the set $\mathcal{Q}$ of potential Internet gateways. Initially, $\mathcal{Q} = \mathcal{G}_s$ (line 9). More precisely, LARS searches for the gateway g that has the *least* distance from the node s (line 13), and then removes that gateway from the set $\mathcal{Q}$. Note that any isotonic routing metric [19] can be used to measure the path length, and for simplicity here we adopt the hop count. Then, the function $ShortestPath(s, g, M')$ gives the set, denoted with $path_{s,g}$, of mesh nodes on the shortest path between source s and gateway g in the tested topology $\mathcal{M}'$. Note that $\mathcal{M}'$ is the original network topology $\mathcal{M}$ *pruned* of bottlenecked links, as better explained in the following. For each mesh node k in $path_{s,g}$ we update the arrival rate (from lines 16 to 19), and we recompute the forwarding probabilities to each of its neighbors (line 20). Finally, the algorithm checks if the *tentative* forwarding matrix $\mathcal{P}'$ obtained after adding this new flow from s to g generates a new bottleneck in the mesh network. If the stability test is positive, the algorithm confirms the flow allocation (line 24), and moves on considering the next mesh node in the set $\mathcal{A}$ (line 25). On the other hand, if gateway g is a bottleneck it can not be used as an egress point to the Internet for this new flow, and the link connecting g to the Internet is removed from the tested network topology $\mathcal{M}'$ (line 28). If there are still available gateways in the set $\mathcal{Q}$, LARS will repeat the steps from line 13 to line 30, testing a new gateway g. If $\mathcal{Q} = \emptyset$ then all the paths towards the gateways in $\mathcal{G}_s$ are bottlenecked. This implies that no new flow can be admitted in s, and the flow allocation procedure terminates. The last forwarding matrix that ensures a stable mesh net-



(a) $n_R/n = 5\%$ (b) $n_R/n = 20\%$

Figure 3. Illustrative network topologies. Bold circles represent residential gateways, while filled triangles are provider gateways.

work is the final outcome of the route selection algorithm (line 26). Note that the computed forwarding probabilities provide the fraction of uplink traffic that each node should transmit to each of its neighbors.

Concerning the implementation requirements of the LARS scheme, it is worth pointing out that it should run on a centralized entity. For instance, we can envision that one of the provider gateways collects the flow demands, computes the forwarding probabilities, and distributes them to the mesh routers through special routing messages (e.g., using HNA messages provided by OLSR to distribute gateway-related information [5]).

4. Performance Evaluation

In this section we use computer-based simulations to validate our analysis and evaluate the performance gains of LARS over a shortest-path routing algorithm. To this end we have developed a discrete-event simulator of an heterogenous mesh network.

In our tests we deploy the mesh nodes over a square area of side equal to 1Km. In the center of this area we place one provider gateway (i.e., $n_P = 1$) with a high-speed Internet connection ($W_P = 1$ Gbps). We have chosen a n-point grid layout for the mesh topology, where the grid points are separated by 100m. Then, we randomly pick up n_O grid points where we place mesh routers, and in the remaining n_R grid points we place residential gateways ($n = n_O + n_R = 100$). Therefore, diverse levels of network heterogeneity can be simulated by varying the percentage of residential gateways over mesh routers. For each setting of the n_R/n ratio, we consider twenty random instances of network topologies. For the sake of clarity, two illustrative network topologies are plotted in Figure 3. The transmission rate W_O of the wireless medium is set to 50 Mbps, while the bandwidth W_R of the low-speed Internet connections available at the residential gateways is assumed equal to 5 Mbps, which is consistent with typical uplink DSL lines.

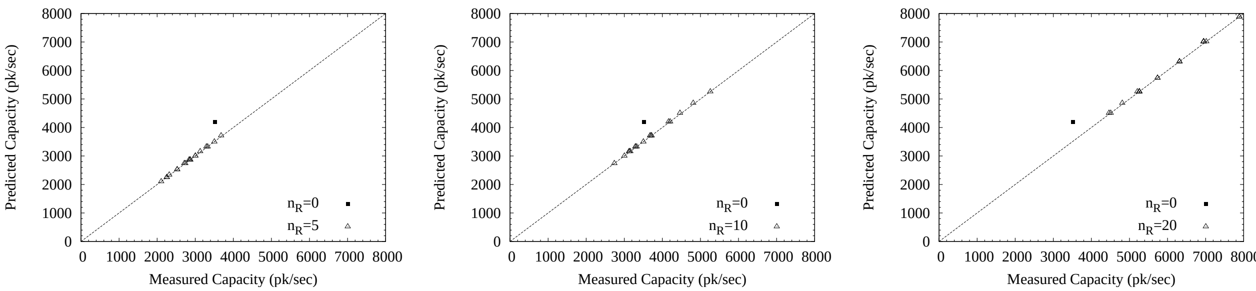
In the simulator we implemented a simplified physical layer model based on the Protocol Model [11], where the transmission ranges and interference ranges of each station are fixed and equal to 100m and 200m, respectively. Regarding the MAC protocol, our simulator implements a collision-free version of a classical CSMA access scheme without backoff. More precisely, the nodes have complete information on other nods' state and coordinate their channel transmissions so as to avoid collisions. This model is suitable to represent the location-dependent contention issues due to differences in the number of contending neighbors. A more realistic physical model including collision effects will be considered in future work.

The following graphs show the average results obtained by replication of simulation runs, and the 95% confidence intervals of collected statistics, which are very tight.

4.1. Validation of model accuracy

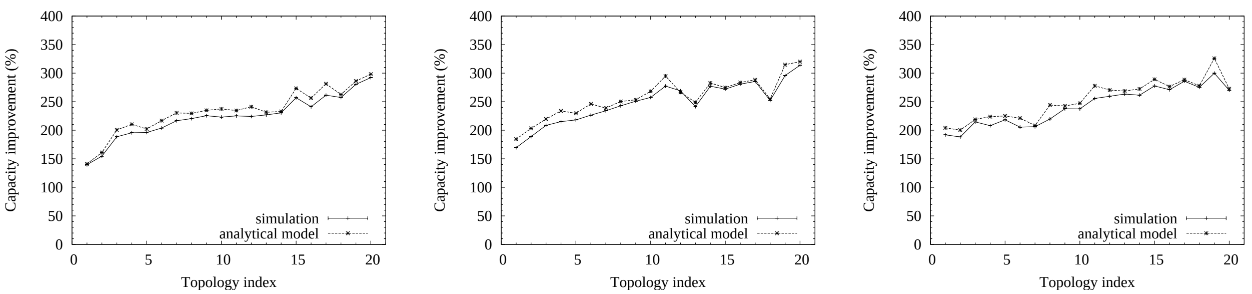
To validate our analysis we consider a shortest-path routing algorithm using a contention-aware routing metric derived form the IRU heuristic proposed in [20]. More precisely, in the route computation, the link cost of a gateway Internet connection line is equal to the inverse of fixed line bandwidth, while the cost of a wireless link l is n_l/W_O, where n_l is the number of other wireless links whose transmission can interfere with the transmissions of link l. Regarding the traffic model we assume that each mesh node produces UDP packets of size $L = 1000$ bytes with an exponentially distributed packet-interarrival time equal to λ. In other words, all mesh nodes generate the same amount of uplink traffic to the Internet.

Figure 4 shows a series of scatter plots comparing the network capacity predicted by our model and the one measured with the simulator, under different percentages of residential gateways in the mesh network. Twenty topologies are investigated per each network scenario. The network capacity is estimated according to the formal definition provided in Section 2.2.1, i.e., the maximum load that can be injected in the network without making the network unstable. The shown results indicate that our queuing model is very accurate in predicting the network capacity, and a small error is introduced only in a network scenario without residential gateways. This also implies that our model can be used to precisely identify which gateway or mesh router in the network becomes the bottleneck for the overall network performance. Similar results have been obtained using a minimum hop-count routing algorithm and are not reported here for space limitations. Furthermore, we can observe that adding additional gateways in the network not necessarily improves the network capacity. On the contrary, when the percentage of residential gateways in the network is 5% the network capacity can be significantly lower than the one obtained in a mesh without residential gateways. This is due to the load-unaware behavior of the shortest path rout-



(a) $n_R/n = 5\%$ (b) $n_R/n = 10\%$ (c) $n_R/n = 20\%$

Figure 4. Comparison of the model predictions with the simulation results for the shortest path routing algorithm using contention-aware link costs.



(a) $n_R/n = 5\%$ (b) $n_R/n = 10\%$ (c) $n_R/n = 20\%$

Figure 5. Capacity gain of LARS over shortest path routing using contention-aware link costs. The topologies in each graph are ordered from the one with maximum network capacity to the one with minimum network capacity.

ing algorithm that may easily lead to overuse gateways with congested or low-speed Internet connections. Only when the percentage of residential gateways over mesh routers is sufficiently high (in our scenarios 20%) there is an appreciable benefit from sharing the low-speed Internet connections. These results motivate the need for load-aware route selection algorithms in order to improve system performance.

4.2. Performance of proposed route selection algorithm

In this section we compare the performance of the LARS algorithm and a shortest path routing protocol using the contention-aware routing metric defined above. Specifically, for each of the network topologies considered in Figure 4, we employ our model to estimate the network capacity achievable using either the LARS algorithm or the shortest path routing. Then, in Figure 5 we show the capacity improvement provided by LARS against the shortest path routing, measured both analytically and through simulations. For the sake of clarity, in the graphs we sorted the topologies in decreasing order of network capacity for shortest path routing algorithm. In the considered scenarios, LARS significantly outperforms shortest path routing ensuring throughput improvements that ranges from 35% up to 210%. It is important to note that the performance gain is higher for the most disadvantaged topologies, i.e., for the topologies where the shortest path routing obtained the lowest performance. Furthermore, the shown results confirm that our model accurately predict the network capacity also for the LARS scheme.

The remarkable throughput improvements provided by LARS can be explained by considering the ability of this algorithm to evenly distribute the network load among all the available gateways. To confirm this feature, in Figure 6 we compare the utilization of the residential gateways' Internet connections obtained, by simulation, using LARS and the shortest path routing, under saturated conditions. The graph refers to an illustrative topology with $n_R/n = 20\%$. As shown in the figure, the LARS algorithm ensures to fully utilize the available bandwidth of the gateways' fixed lines, while the shortest path routing underutilizes many gateways. This leads to the inefficient usage of the network resources, and, consequently, lower network performance. On the contrary, LARS is able to optimally utilize the net-

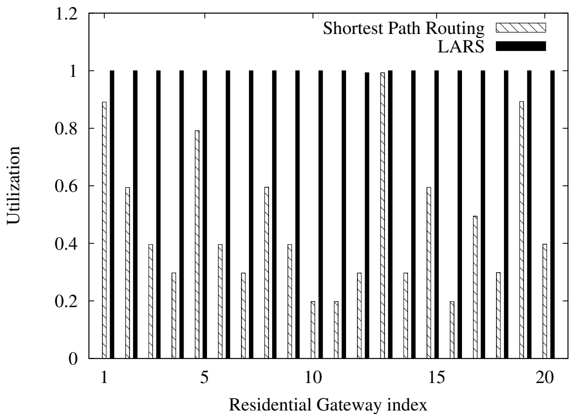


Figure 6. Comparison of the utilization of Internet fixed lines for residential gateways using LARS and shortest path routing with $n_R/n = 20\%$.

work resources, maximizing the network capacity.

5. Conclusions

Focusing on heterogeneous mesh networks where gateways with low-speed Internet connections may exist, in this paper we have developed a queuing network model that accurately predicts the path residual capacity, and precisely identify network bottlenecks. By exploiting this predictive tool, we have designed LARS, a Load-Aware Route Selection algorithm that improves the network capacity by evenly distributing the network load among available gateways. Simulation results show that LARS significantly outperforms shortest path routing using contention-aware link costs, providing up to 210% throughput improvement in the considered network scenarios.

The presented results have been obtained assuming an idealized MAC protocol that tries to capture primarily location-dependent contention issues. However, the extension of our analysis to a real 802.11 MAC protocol is an ongoing activity. Furthermore, LARS design does not consider end-to-end delays in the selection of feasible network paths. Thus, integrating end-to-end delay constraints in the routing scheme is an interesting and open research area.

References

[1] H. Aiache, L. Lebrun, V. Conan, and S. Rousseau. A load dependent metric for balancing Internet traffic in Wireless Mesh Networks. In *Proc. IEEE MeshTech 2008*, pages 629–634, Atlanta, GA, USA, September 29, 2008.

[2] B. Awerbuch, D. Holmer, and H. Rubens. The Medium Time Metric: High throughput Route Selection in Mult-rate Ad Hoc Wireless Networks,. *Mobile Networks and Applications*, 11:253–266, 2006.

[3] R. Bruno, M. Conti, and E. Gregori. Mesh Networks: Commodity Multihop Ad Hoc Networks. *IEEE Commun. Mag.*, 43(3):123–131, March 2005.

[4] R. Bruno, M. Conti, and A. Pinizzotto. A Queuing Modeling Approach for Load-Aware Route Selection in Heterogenous Mesh Networks. Technical report, IIT-CNR, April 2009.

[5] T. Clausen and P. Jaquet. Optimized Link State Routing Protocol (OLSR). RFC 3626, October 2003.

[6] N. D., L. Santhanam, N. Nandiraju, and D. Agrawal. Achieving Load Balancing in Wireless Mesh Networks Through Multiple Gateways. In *Proc. of IEEE MASS'06*, pages 807–812, Vancouver, Canada, Oct. 9–12 2006.

[7] D. De Couto, D. Aguayo, J. Bicket, and R. Morris. A High-Throughput Path Metric for Multi-Hop Wireless Routing. In *Proc. of ACM MobiCom*, pages 134–146, San Diego, CA, USA, September, 14–19 2003.

[8] R. Draves, J. Padhye, and B. Zill. Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks. In *Proc. of ACM MobiCom'04*, pages 114–128, Sept. 26–Oct. 1 2004.

[9] V. Gambiroza, B. Sadeghi, and E. Knightly. End-to-End Performance and Fairness in Multihop Wireless Backhaul Networks. In *Proc. of ACM MobiCom*, pages 287–301, Philadelphia, PE, Sept. 26–Oct. 1 2004.

[10] M. Genetzakis and V. Siris. A Contention-Aware Routing Metric for Multi-Rate Multi-Radio Mesh Networks. In *Proc. IEEE SECON 2008*, pages 242–250, San Francisco, CA, USA, June 16–20, 2008.

[11] P. Gupta and P. Kumar. The Capacity of Wireless Networks. *IEEE Trans. Inform. Theory*, 46(2):388–404, March 2000.

[12] R. Karrer, A. Sabharwal, and E. Knightly. Enabling Large-Scale Wireless Broadband: The Case for TAPs. *ACM SIGMOBILE Comp. Comm. Review*, 34(1):27–34, 2004.

[13] H. Kobayashi. Applications of the diffusion approximation to queuing networks, Part I - Equilibrium distributions. *Journal of ACM*, 21(2):316–328, April 1974.

[14] L. Ma and M. Denko. A Routing Metric for Load-Balancing in Wireless Mesh Networks. In *Proc. of IEEE AINAW '07*, volume 2, pages 21–23, May 21–23, 2007.

[15] Meraki Networks Inc. Residential & MDU Case Studies. http://meraki.com/.

[16] Ozone. French Wireless ISP. http://wwwo.zone.net/.

[17] K. N. Ramachandran, M. Buddhikot, G. Chandranmenon, S. Miller, E. Belding-Royer, and K. Almeroth. On the Design and Implementation of Infrastructure Mesh Networks. In *Proc. of WiMesh*, Santa Clara, CA, USA, September, 26 2005. IEEE Press.

[18] V. Siris, I. Askoxylakis, M. Conti, and R. Bruno. Enhanced, Ubiquitous and Dependable Broadband Access using MESH Networks. *ERCIM News*, 73:50–51, April 2008.

[19] Y. Yang, J. Wang, and R. Kravets. Designing Routing Metrics for Mesh Networks. In *Proc. of IEEE WiMesh, 2005*, Santa Clara, CA, USA, September, 26 2005.

[20] Y. Yang, J. Wang, and R. Kravets. Load-balanced Routing For Mesh Networks. *ACM Mobile Comp. and Comm. Review (M2CR)*, 1, 2006.

[21] P. Zou, X. Wang, and R. Rao. Asymptotic Capacity of Infrastructure Wireless Mesh Networks. *IEEE Trans. Mobile Comput.*, 7(8):1011–1024, August 2008.