Gateway Control of Wireless Mesh Networks

Kamran Jamshaid, Lily Li and Paul A.S. Ward
Shoshin Distributed Systems Group
Department of Electrical and Computer Engineering
University of Waterloo, Waterloo, Ontario N2L 5N1
Email: {kjamshai, l7li, pasward}@shoshin.uwaterloo.ca

***Abstract*— The use of multi-hop wireless networks based on 802.11 technology is extensive and growing. The primary advantages of this approach are ease of deployment and lower cost. However, such networks typically exhibit poor fairness properties, often starving nodes if they are too many hops distant from the gateway. Research efforts to address this problems have largely focused on notions of fairness (*e.g.*, time fairness *vs.* bandwidth fairness ; proportional fairness *vs.* max-min fairness, *etc.*), with little attention being paid to how the desired fairness might actually be achieved.**

MAC-layer approaches have been shown to not extend fairness properties across the network, and thus do not produce the desired network-layer fairness. The only extant technique we are aware that has been shown to achieve network-layer fairness is source-rate limiting. The problem with such an approach is that it requires telling the mesh routers what their fair share is, and having them enforce that rate.

In this paper we propose an alternate approach, exploiting the traffic patterns inherent in mesh networks. Since all traffic is expected to traverse the gateway, we enforce rate control there, anticipating that the sources will react to limit their traffic to their gateway-limited capacity. We use a fair-share computational model to determine the appropriate rate for the various sources. Our approach does not require any additions or changes to the basic mesh network protocols, and works well with 802.11-based systems. We evaluate the performance of our algorithm using simulation over various mesh topologies.

I. Introduction

Wireless Mesh Networks (WMNs) are a type of multi-hop wireless network that have received significant recent attention as an alternative technology for last-mile broadband Internet access [1], [3]. These networks are composed of regular *mesh nodes* that act as both data sources/sinks and as routers, and *gateway nodes* that bridge traffic between the mesh and the wired network (usually the Internet). Also referred to as community or infrastructure wireless networks, these networks have the following properties:

- Fixed location: The mesh nodes in a community wireless network are usually located on rooftops or other fixed locations. As a result, the topology is mostly static, with topological changes occurring only through the addition or removal of mesh nodes.
- Powered: Given a fixed location, mesh nodes can be powered from the electricity grid. As such, power use is not a significant issue.
- Traffic pattern: Unlike the general peer-to-peer paradigm of *ad hoc* networks in which any two nodes can communicate with each other, the traffic in WMNs is between mesh nodes and some distant server on an external network, *via* one of the gateway nodes in the WMN.

WMNs based on the IEEE 802.11 MAC require mesh nodes to contend for access to the wireless medium. This contention for medium access produces a structural asymmetry in the network. Data from flows traversing multiple hops has to contend for the medium at each intermediate hop, as compared to data from flows that originate in the vicinity of the gateway. This means that current WMNs based on the IEEE 802.11 MAC and standard network-layer protocols cannot provide fairness to each node in the network. In particular, it has been demonstrated that nodes close to the gateway can starve those that are more hops away [3], [8]. Although significant research has been done to address fairness issues over MAC-layer flows within a single-hop, very little research has been done to address the problem at the network layer in multi-hop wireless networks.

It has been shown that network-layer fairness can be achieved by knowing the fair-share bandwidth each node can receive, and limiting the nodes to that rate [9]. However, while this approach is possible, it requires that all mesh routers be modified to operate the relevant source-rate-limiting protocol. In this paper we propose an implicit feedback-based mechanism that is enforced at the gateway and restricts traffic flows to their fair share. Our rate-control mechanism allows us to implement in a single, central location any feasible bandwidth allocation policy, where the policy is feasible if does not violate the constraints imposed by the network topology.

This remainder of this paper is organized as follows. In Sect.II, we describe the unfairness phenomenon observed in a simple mesh topology, and demonstrate how it can be fixed through traffic rate limiting. We then present our simple gateway rate control algorithm that enforces implicit flow control by dropping or delaying excess traffic at the gateway. In Sect.IV we provide simulation results that show the performance of our proposed solution for chain, mesh, and random network topologies of varying sizes. We conclude by observing what issues remain open.

II. Fairness Issues in WMNs

To demonstrate unfairness effects in WMNs, we performed a series of simulations with the Network Simulator ns-2 [11]. We use the default ns-2 radio model, which results in a transmission range of 250 m. and an interference range of 550 m. We consider a simple chain topology (Fig.1) where the nodes

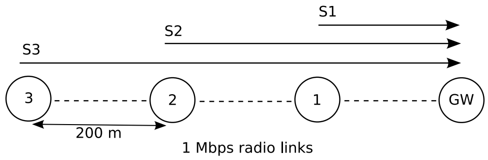


Fig. 1. Simple chain topology for demonstrating the unfairness between different flows in the network. All nodes are 200 m. apart, thus allowing only the neighboring nodes to directly communicate with each other.

are placed 200 m. apart. This means that while each radio node can directly communicate only with its one-hop neighbor, a successful transmission is possible only when the two-hop neighbors of the receiver do not attempt a transmission at the same time. We use a MAC data rate of 1 Mbps to simulate the radio link between two adjoining nodes. We tested this topology with both TCP and UDP streams. For TCP we used an infinite file transfer to simulate the traffic flows, and for UDP we simulated constant bit rate (CBR) traffic.

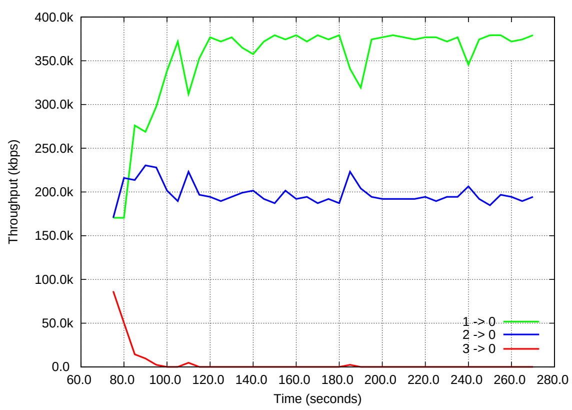


Fig. 2. Throughput over time corresponding to TCP flows for the topology shown in Fig.1. Node 3 that is 2 radio hops away from the gateway starves, while Node 2 gets almost half as much throughput as Node 1.

Fig.2 shows how the TCP throughput varies over time for this simple chain topology. We observe that Node 3 starves throughout the duration of the experiment. Analysis of the trace data shows that this is because it experiences exponential backoff far more frequently and in greater degree than either Node 1 or Node 2. The cause of this backoff is the hidden terminal problem [12] resulting from the transmission of ACKs from the gateway, exacerbated by the binary exponential backoff algorithm of 802.11. This problem has been observed before (*e.g.*, [2], [10]), though typically in the context of single-hop wireless networks, where no source-rate limiting was imposed. In addition, Node 2 only gets about half as much throughput as Node 1. This is likely because TCP is proportionally fair, and traffic from Node 2 requires use of the medium twice to reach the gateway, while that from Node 1 requires only a single access to the medium.

If we consider absolute fairness, the fair share corresponding to each flow for this topology is around 155 kbps. In Fig.3 we plot the offered load *vs.* the network throughput for CBR traffic. The plot shows that the throughput for each flow scales linearly with increasing traffic load until we reach the fair-

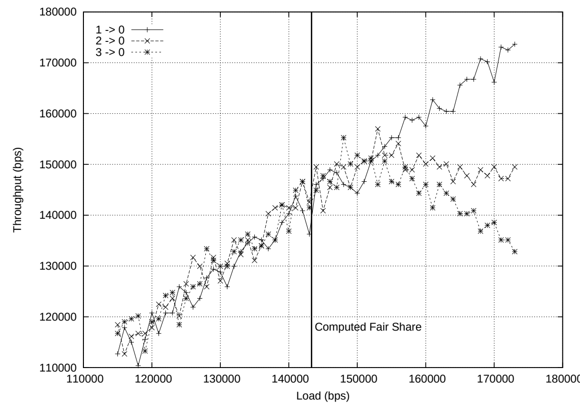


Fig. 3. A plot illustrating how the throughput of the network topology shown in Fig.1 varies with increasing traffic load at each node.

share point. Beyond this point, the throughput for Node 3 starts decreasing and eventually drops to 0 at a high-enough traffic load. Jun and Sichitiu [7] show that the unfairness exhibited by Node 1 is partly caused by the fact that both the locally generated traffic as well as the relayed traffic are queued together at the node. When the traffic load increases and the node cannot transmit all of its data, its queue overflows, being filled with only the locally generated CBR traffic.

Fig.3 is illustrative of the more-general result that fairness can be achieved in a WMN by limiting data sources to their fair share [9]. Prior approaches to enforce such source-rate limiting use explicit protocols, which can result in considerable overhead from protocol traffic exchanged between nodes.

III. Gateway Rate-Control Algorithm

We observe that with the traffic flow predominantly directed to and from the gateways, gateways become a natural choice for enforcing congestion-control policies and fair bandwidth-allocation mechanisms. The gateway has a unified view of the entire network, and thus is better positioned to manage allocation of network resources more fairly. We propose a simple, efficient algorithm that takes advantage of the centralized gateway node for indirectly enforcing source rate-control.

Our gateway rate-control protocol requires the gateway to perform flow classification of all the traffic entering the gateway. The gateway can then enforce traffic policing so that the rate allocated to each flow is limited only to the fair share corresponding to that flow. If the stream source is using an adaptive transport protocol (like TCP), it would register this delay or dropping of packets as an indication of congestion, and therefore slow down by reducing its congestion window size. This slow down of aggressive sources frees up the wireless medium, thus providing an opportunity for starving flows to transmit their packets.

Our gateway rate-control protocol consists of three steps:

1) Gather information required to compute the fair-share bandwidth
2) Compute the fair share for each stream
3) Enforce the computed rate for each stream at the gateway

We now describe the three steps.

A. Information Gathering

The information that must be gathered by the gateway will be a function of the fair-share computation used. In general, however, it will consist of the following. First, some notion of the network topology will be required. This is necessary to compute what rates are feasible, which is determined in part by which nodes interfere with which other nodes. While many fair-share computation models require precise link-interference data (*e.g.*, [2], [3], [10]), others simply need neighbour knowledge [8]. For our experiments, we use the simple model of Li *et al.* [9] that only requires neighbour information. If we moved to a more-sophisticated model, we can require mesh nodes to report radio transmissions on the same channel as the node interface.

Determining simple topology information can be achieved directly if link-state routing is used (*e.g.*, OLSR) or *via* a utility such as `traceroute`, thus obviating the need to change mesh routers. The information needed by more-sophisticated models likely requires explicit feedback from the mesh nodes, and therefore incurs an overhead and a change requirement on existing equipment. In either approach, because of the static nature of the WMN topology, the information is expected to remain constant for large durations of time, and thus the cost incurred in its collection can be ignored.

In addition to knowing what is feasible, the gateway must also know which mesh nodes have active flows, since there is no need to reserve bandwidth for nodes that are not transmitting. Since all flows have to pass through the gateway, it is trivial for the gateway to determine which flows are currently active by performing per-packet inspection.

B. Fair-share Computation

The second problem is how to efficiently compute the fair-share capacity of the network. This problem has been addressed by a number of authors (*e.g.*, [4]), with the answer changing depending on the definition of fair sharing (*e.g.*, some look at proportional fair-sharing of bandwidth [10], some at absolute [8]; [3] looks at equal-time sharing, *etc.*). As the focus of this paper is not on the fair-share capacity model, but rather on the practical matter of achieving fairness by rate limiting in the gateway, we adopt a restricted version of the model developed by Li *et al.* [9]. Specifically, our simplifications constrain us to absolute fair sharing of the bandwidth, with single-rate routers. We will extend this to more-complex definitions of fairness in future work.

We now briefly describe our simplified version of the Li *et al.* fair-share computation model. The Li *et al.* model treats the network as a graph, with mesh nodes as vertices connected *via* bi-directional wireless links. A link interferes with another link if either endpoint of one link is within transmission range of either endpoint of the other link. Thus, the set of all links that interfere with a given link, referred to as the *collision domain* of that link, are all those within two hops of either endpoint of the link. It is assumed by the model that the links within a collision domain cannot transmit simultaneously. This actually over-estimates link contention. However, given that link interference, defined by transmission range rather than interference range, is under-estimated, the presumption (born out by detailed simulation studies) is that the overall model is approximately correct.

It is then sufficient to determine the bottleneck collision domain, which will be a function of the usage of the links within each collision domain. Link usage is determined by routing and demand. For the work in this paper, we presume that routing is relatively static. That is, it changes infrequently compared with traffic demand changes. This is generally true, though it would not be difficult to remove this assumption, by simply recomputing the feasibility as routing changed. We consider network demand to be binary. That is, either a node is silent or it demand is insatiable. This corresponds to TCP behaviour, which either is not transmitting or will increase its transmission rate to the available bandwidth. We are currently extending this model to incorporate QoS flows. Given the stream activity, we can then compute the load over each link, and in turn compute the load in each collision domain. Given the single-rate assumption, the bottleneck collision domain is simply that domain with the greatest load, and the fair share is determined simply by dividing the rate by the load.

C. Fair-share Enforcement

To enforce the fair-share rate, the gateway node sorts all incoming packets by stream, placing them into a leaky bucket corresponding to their stream. Each bucket has an adjustable rate, releasing a packet after a delay of $\frac{lastPacketSize}{lastRate}$ from when it last released a packet. On release from its leaky bucket, a packet is placed in the queue of the outgoing interface corresponding to its destination.

The size of the leaky buckets determines how rigidly the fairness is enforced. If it is small, fairness is enforced very rigidly, with little allowance for deviation above the fair share. If large, it will allow bursts above a long-term average. This is, to some degree, a policy decision, though upper bound on the size is limited by the capacity of the mesh to absorb the temporary unfairness that might ensue.

IV. Performance Evaluation

We evaluated the performance of our gateway rate-control algorithm on a number of different WMN topologies with the ns-2 simulator and the radio model described in Sect.II. To provide a quantitative measure of fairness, we use Jain's Fairness Index (JFI) [5] that provides an aggregate measure of fairness between flows, as well as $\frac{min_{throughput}}{avg_{throughput}}$ ratio to identify any starving flows in the network.

We used a number of chain, grid, and random topologies to test the performance of the algorithm. Chain topologies are easy to analyze, and thus serve as a useful tool for checking the observed results. We tested the algorithm on 5-, 10-, and 15-hop chain topologies with a gateway at one end of the chain. Grid topologies are more complex than chain topologies, yet still maintain a regular structure with equidistant separation between neighboring nodes, and thus amenable to analysis. We tested the algorithm on 3x3 and 4x4 grid topologies,

TABLE I

QUANTITATIVE FAIRNESS ANALYSIS OF TESTED TOPOLOGIES

Topology	**JFI**	$\frac{min_{throughput}}{avg_{throughput}}$
5-Hop Chain	0.99977	0.987
10-Hop Chain	0.99960	0.964
15-Hop Chain	0.99966	0.960
3x3 Grid	0.99802	0.914
4x4 Grid	0.99488	0.799
5-Node Random	0.99989	0.979
10-Node Random	0.99982	0.967
15-Node Random	0.99958	0.930

with the gateway located at one corner of the network. In both chain and grid topologies, the separation between any two communicating nodes was maintained at 200 m, per our original experiments described in Sect.II. Finally, as mesh deployments are unlikely to be as structured as either chain or grid topologies, we tested the algorithm on 5-, 10-, and 15-node random mesh topologies. The position of the nodes was randomly chosen on a flat topographical grid of size of 1000 m. x 1000 m. for 5-node, 1500 m. x 1500 m. for 10-node, and 2000 m. x 2000 m. for 15-node random networks.

While we have tested with a mixture of upstream and downstream traffic (*i.e.*, to and from the gateway), we only show the results here for traffic to the gateway. Downstream traffic corresponds effectively to source-rate control, which has already been shown to work [6], and thus only the upstream results are novel. The traffic was TCP with infinite file transfer.

Results for chain, grid, and random topologies are shown in Fig.4, Fig.5, and Fig.6, respectively. JFI and $\frac{min_{throughput}}{avg_{throughput}}$ corresponding to the three topologies is shown in Table I.

These results show that our algorithm prevents flow starvation and provides a fair distribution of bandwidth between all active flows. However, these experiments were limited to static flows. That is, the flows existed for the entire experiment. We, therefore, extended our experiments to consider TCP flows that started and stopped during the course of the experiment, thus requiring re-computation and adjustment of the flow rate on the leaky buckets during the execution. Our preliminary results in this area show that such dynamic traffic is handled well with respect to stream activation, but is somewhat problematic when dealing with deactivation. Specifically, new streams are rapidly identified, and the fair-share rate is re-computed and the algorithm is able to provide bandwidth for the new stream quickly. However, identifying stream deactivation is difficult as substantial packet re-ordering is occurring within the mesh. As a result, a stream may appear silent at the gateway for quite some time, even though it is still an active stream.

Second, we extended the computation model to a multi-channel model. While the details of that model are the subject of another paper, we do observe here that our results with the multi-channel model showed that gateway rate control worked as effectively in that environment as in the single-channel system. This is extremely significant, as current source-rate control methods require packet snooping, which is not feasible in a multi-channel system.

V. CONCLUSIONS AND FUTURE WORK

WMNs, especially those based on the IEEE 802.11 MAC, exhibit extreme fairness problems, requiring existing deployments to limit the maximum number of hops to the gateway to prevent distant nodes from starving. In this paper we have proposed a gateway rate-control algorithm that induces source rate-control for network traffic using adaptive transport protocols like TCP. We achieve this without any overhead associated with explicit co-ordination between the nodes.

Our method consists of three components: gathering controller-specific information required for computing the fair-share bandwidth, computing the fair-share bandwidth, and enforcing that fair-share at the gateway. Simulation results over various topologies demonstrate that our approach is effective in providing fair sharing of the network resources. It works with both single- and multi-channel systems, and with both static and dynamic traffic, though detecting deactivation of dynamic streams remains problematic.

In the future we expect to extend our algorithm to provide any feasible fairness policy enforcement for both upstream and downstream traffic, where where a policy is feasible if it does not violate the constraints imposed by the network topology. We are also studying feedback-control-based approaches, to compensate for deficiencies in the computation model.

REFERENCES

[1] J. Bicket, D. Aguayo, S. Biswas, and R. Morris. Architecture and evaluation of an unplanned 802.11b mesh network. In *ACM International Conference on Mobile Computing and Networking*, pages 31–42, August 2005.

[2] Z. Fang and B. Bensaou. Fair bandwidth sharing algorithm based on game theory frameworks for wireless ad-hoc networks. In *IEEE INFOCOM*, Hong Kong, 2004.

[3] V. Gambiroza, B. Sadeghi, and E. Knightly. End-to-end performance and fairness in multihop wireless backhaul networks. In *ACM International Conference on Mobile Computing and Networking*, pages 287–301, September 2004.

[4] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. In *ACM Mobicom*, September 2003.

[5] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. Technical report, DEC Research Report TR-301, Digital Equipment Corporation, Hudsosn MA, September 1984.

[6] S. Jakubczak, A. T. Lau, L. Li, and P. A. Ward. Feasibility-based rate-control fairness algorithm for wireless mesh networks. Technical report, Department of Electrical and Computer Engineering, University of Waterloo, August 2004.

[7] J. Jun and M. L. Sichitiu. Fairness and QoS in Multihop Wireless Networks. In *IEEE Vehicular Technology Conference*, Orlando, FL, October 2003.

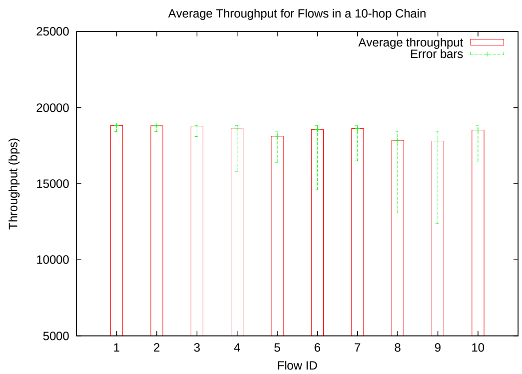
[8] J. Jun and M. L. Sichitiu. The nominal capacity of wireless mesh networks. *IEEE Wireless Communications*, pages 8–14, October 2003.

[9] L. Li, S. Jakubczak, A. T. Lau, and P. A. Ward. Achieving fairness in 802.11-based wireless mesh networks. *Ad Hoc Networks*, Submitted, May 2006.

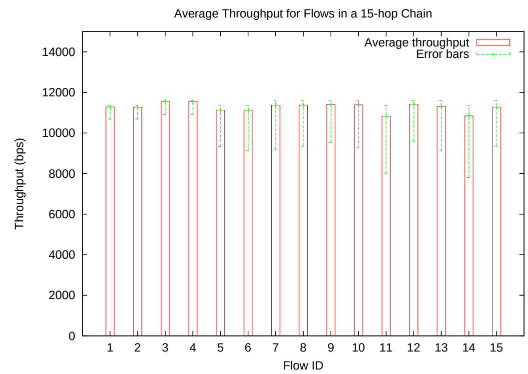
[10] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan. Achieving MAC layer fairness in wireless packet networks. In *Proc. of The 6th Annual International Conference on Mobile Computing and Networking*, pages 87–98, 2000.

[11] The Network Simulator. ns-2. `http://www.isi.edu/nsnam/ns/`.

[12] F. A. Tobagi and L. Kleinrock. Packet switching in radio channels: Part II—The hidden terminal problem in carrier sense multiple-access and busy-tone solution. *IEEE Transactions on Communication*, 23(12):1417–1433, December 1975.

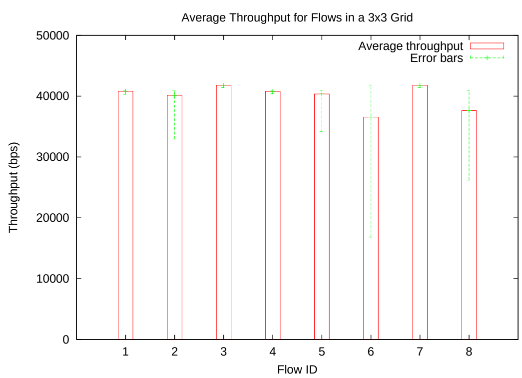


(a) 10-hop chain

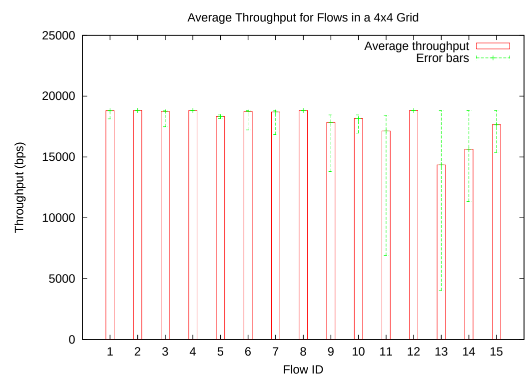


(b) 15-hop chain

Fig. 4. Average throughput of a 10-hop chain and a 15-hop chain topology with upstream TCP traffic from mesh nodes to the gateway node 0 at one end of the chain. The experiment was repeated with 50 simulation runs, and the corresponding error bars show the maximum and the minimum throughput recorded in any of the 50 runs

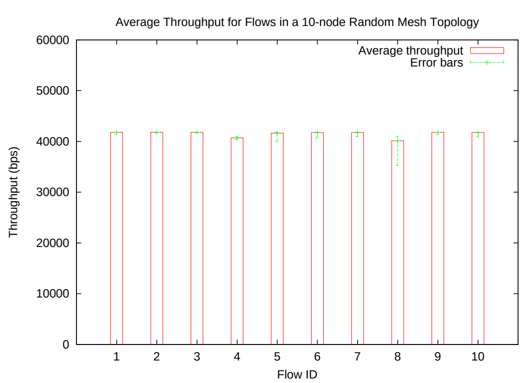


(a) 3x3 grid

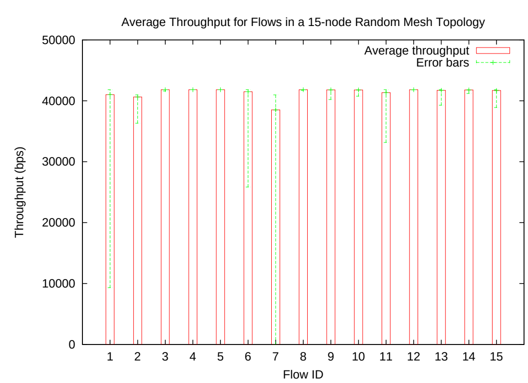


(b) 4x4 grid

Fig. 5. Average throughput of a 3x3 grid topology and a 4x4 grid topology with upstream TCP traffic from mesh nodes to the gateway node 0 at one corner of the grid. The experiment was repeated with 50 simulation runs, and the corresponding error bars show the maximum and the minimum throughput recorded in any of the 50 runs.



(a) 10 node random



(b) 15 node random

Fig. 6. Average throughput of a 10 node random topology in a 1500 m x 1500 m topographical grid and a 15 node random topology in a 2000 m x 2000 m topographical grid with upstream TCP traffic from mesh nodes to the gateway node 0. The experiment was repeated with 50 simulation runs. The error bars show the maximum and the minimum throughput observed in any of the 50 runs.