Multipath Load Balancing in Multi-hop Wireless Networks

Evan P. C. Jones, Martin Karsten, and Paul A. S. Ward, *University of Waterloo, Canada*

***Abstract*—Multi-hop wireless networks have the potential to dramatically reduce the cost of deploying communication infrastructure. However, the nature of this technology limits the capacity of radio links. Thus, it is important to utilize them as efficiently as possible. In this paper, we investigate load balancing across multiple paths as a possible mechanism to improve performance in multi-hop wireless networks. Given the inherent interference of multi-hop transmissions in a single radio channel, it is generally assumed that single-channel multipath routing cannot provide any benefits, but in fact would have detrimental effects on resource efficiency. However, a careful investigation of the issue reveals that under certain theoretic conditions, significant gains are possible. In fact, we show throughput improvements of 80-100% in some scenarios. We present a novel interference metric to assess the quality of a set of disjoint paths. We further present a heuristic path selection algorithm to find appropriate routing paths in structured networks, which is a first step towards the application of our basic results in realistic scenarios.**

***Index Terms*—Communication systems, Computer network performance, Wireless LAN, Routing**

I. Introduction

Multi-hop wireless networks have been a popular research topic for a number of years because they have the potential to dramatically reduce the cost and effort of deploying networks. The vision is that nodes are placed within radio range of each other and they automatically form a network. This eliminates the need for careful planning. One common scenario is to use these types of networks for "last mile" Internet access, as an alternative to traditional wired technologies like cable or DSL. Unfortunately, multi-hop wireless networks have a low throughput, which could prevent them from being useful in many of these scenarios. Two issues cause this low capacity.

The first issue is that the wireless spectrum is a precious and limited resource, so the data rates of digital radios are limited. Currently, the highest possible throughput available with commodity 802.11 radios is 54 Mbps, and that rate is not possible at the maximum transmission range. This is significantly less than the rate of 100 Mbps Ethernet, the most common local networking technology, though more than cable and DSL connections.

The second issue is that the interference from subsequent hops limits the throughput. After sending a packet, a node must wait until the data has been relayed outside of its interference range, otherwise the simultaneous transmissions could interfere. Li *et al.* show that due to this issue, the best theoretical throughput for a single flow in a single-channel multi-hop wireless network is one-third of the channel capacity [1]. This assumes that transmissions are perfectly scheduled and only interfere within their transmission range. However, the interference range is typically longer than the transmission range, which aggravates the problem. For example, if the interference range is twice as long as the transmission range, the best possible throughput falls to one-quarter of the data rate. This low throughput is caused by the fact that a packet must be forwarded over four hops before the next packet can be forwarded, as shown in Fig. 1. In reality, as a path gets longer the throughput falls even further due to inefficiencies in the 802.11 protocol. Therefore, we must make the most of this limited capacity.

Multi-hop wireless routing protocols, such as DSDV [2] or AODV [3], generally select paths with the fewest hops. With this metric, most paths pass near the center of the network [4]. Thus, the center is the bottleneck and becomes congested. In wired networks, load balancing can distribute traffic across multiple links, avoiding this kind of congestion. However, it is unclear if multipath load balancing can be effectively used in multi-hop wireless networks, because the transmission properties are very different. In a wired network, it is sufficient to send data along paths that do not share links, since transmissions along each link occur independently. Wireless transmissions, on the other hand, interfere with

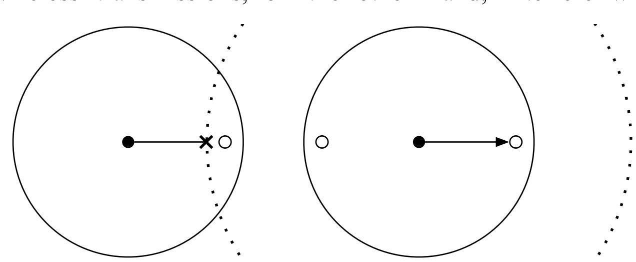

Fig. 1. Interfering transmissions in a chain of nodes. The transmission at the left hand side is unsuccessful due to the interference from the right hand transmission.

Manuscript received January 10, 2004. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada and the Software Telecommunications Group at the University of Waterloo.

E. P. C. Jones is with the Department of Electrical and Computer Engineering at the University of Waterloo, Waterloo, ON, N2L 3G1, Canada (phone: 519-725-3358; fax: 519-746-3077; e-mail: ejones@uwaterloo.ca).

M. Karsten is with the School of Computer Science at the University of Waterloo (e-mail: mkarsten@uwaterloo.ca).

P. A. S. Ward is with the Department of Electrical and Computer Engineering at the University of Waterloo (e-mail: pasward@ccng.uwaterloo.ca).

communication at all nearby nodes. If two transmissions in the same frequency band arrive at a receiver, their signals are combined and it may not be possible to separate them. This means that unlike wired networks, using paths with no links in common is not sufficient to guarantee an improvement.

In this paper, we evaluate the potential for multipath load balancing to improve the throughput of paths in multi-hop wireless networks. In order to isolate the impact of load balancing from other factors, we only consider topologies without mobility. First, we review the previous work in this area. Next, we discuss a simple theoretical model of wireless network interference, and use it to evaluate the potential for load balancing in two scenarios. Finally, we present simulation results that support our analysis.

II. Related Work

Much of the research in multi-hop wireless networks has investigated routing protocols. Many protocols find multiple paths [5, 6, 7], but most do not use load balancing. Instead, they send data exclusively over the best path, and fall back to the alternative paths if it fails. This has been shown to reduce routing overhead and improve reliability, particularly in scenarios where path failure is common [5]. These protocols find link- or node-disjoint paths, since that is sufficient to improve reliability. However, link- or node-disjoint paths can still interfere with each other, hence there may be no performance benefit when using load balancing with these protocols.

Jain *et al.* present an analytical model to compute the upper and lower bounds on the optimal throughput for a specific topology [8]. The model uses a graph formulation of the network connectivity and interference. While the model supports multipath load balancing, they do not compare it with single-path routing.

Some papers have proposed load balancing in combination with other improvements, for example, with different routing metrics [9], packet caching [10], and directional antennas [11]. All of them show some improvement over shortest-single-path routing. However, these studies do not compare the improvements without load balancing, and so it is not possible to determine how much, if any, of the improvement is because of load balancing.

Pearlman *et al.* studied the benefits of load balancing in mobile networks [12]. While it reduces the delay for multiple-channel networks, the improvements are "negligible" for single-channel networks. They conclude that the coupling between single-channel paths severely limits the gains.

Wu and Harms define the *correlation* between two node-disjoint paths as the number of links between nodes on the separate paths [13]. Their results show that as the correlation increases, the end-to-end delay along both paths increases. They introduce a routing protocol that balances traffic across the least-correlated paths, in order to decrease the delay. They present no throughput results, and the delay results seem to show no improvement without mobility.

Pham and Perreau provide an analysis of the traffic distribution of shortest-path routing [4]. Their results indicate that nodes near the centre of the network must forward more traffic, and therefore the center is the bottleneck. They show that given a perfect load-balancing algorithm that distributes traffic evenly through the network, congestion is reduced and the overall throughput improves. However, naively using a number of shortest paths has been shown to not be effective, unless hundreds of paths are used [14]. This indicates that new routing metrics are needed in order to effectively spread the load. One proposal uses the concept of electric field lines to select routes that are physically separated [15]. Unfortunately, this idea relies on location-based routing, which requires special hardware in each node. While the paper discusses load balancing as a potential application of the protocol, they do not investigate its performance.

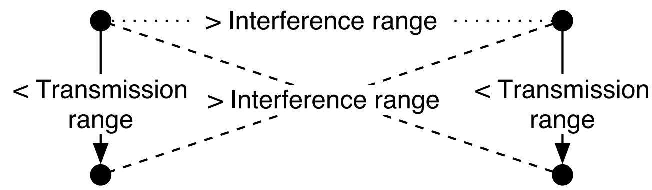

Fig. 2. Geometric requirements for simultaneous transmissions. If the inequalities are satisfied, both transmissions will succeed, according to the protocol model of interference. The dotted line represents an additional requirement for systems that use physical carrier sensing.

III. Load Balancing in Multi-Hop Wireless Networks

In order to improve the throughput by using multiple paths, transmissions along those paths must be able to occur simultaneously. When does this happen in a multi-hop wireless network? In order to answer this question, we need to model the wireless interference. We use a very simple model called the protocol model of interference [8].

A. Protocol Model of Interference

In this model, a node, n_i, has a radio with a transmission range of T_i and an interference range $I_i \geq T_i$. The distance between node n_i and n_j is given by d_{ij}. Node n_j can successfully receive a transmission from node n_i if the following conditions are satisfied:

1. $d_{ij} \leq T_i$
2. Any node n_k such that $d_{kj} \leq I_k$ is not transmitting

If physical carrier sensing is used the sender tries to detect any ongoing communication before it begins to transmit. This attempts to avoid corrupting an existing transmission. This means that the transmitter must also be out of range of other transmissions, adding an optional third condition:

3. Any node n_k such that $d_{ki} \leq I_k$ is not transmitting

A graphical representation of two transmissions that satisfy the requirements is shown in Fig. 2. This simple model treats the interference caused by a transmitter as either completely

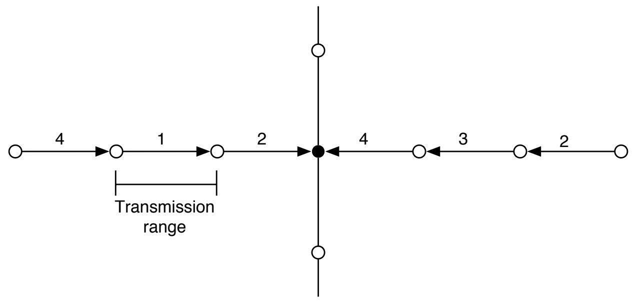

Fig. 3. Cross network topology. Data flows in to and out of the middle node. The edge labels show the optimal schedule for using two paths towards the middle node. The interference range is double the transmission range.

TABLE I
CROSS THROUGHPUT WITH THE PROTOCOL MODEL OF INTERFERENCE

Dir.	C/S	Paths (Global Schedule)				Paths (Rate Limited)			
		1	2	3	4	1	2	3	4
Out	No	¼	½	¾	⅘	¼	½	½	½
	Yes	¼	½	½	½	¼	⅓	⅓	⅓
In	No	¼	½	¾	⅘	¼	½	½	½
	Yes	¼	½	¾	⅘	¼	½	½	½

destroying the signal, or not interfering at all. This is not an accurate reflection of reality, but it is a useful tool for reasoning about multipath load balancing. In real systems, it is very likely that the interference range will be greater than the transmission range. For the remainder of the paper, we assume that it is twice the transmission range. This is a close match for the simulation model presented later.

B. Cross Topology

As an initial load balancing example, consider the cross topology shown in Fig. 3, but imagine that the arms of the cross continue to infinity. The nodes are placed one transmission range apart to form a connected network. This configuration has multiple paths if the midpoint sends or receives along multiple arms of the cross at the same time. In this case, all the interference is around the middle node.

In the remainder of the paper, we express throughput as fractions of the link data rate. If we assume that the interference range is twice the transmission range, then the optimal throughput along a single path is 0.25. This is because the middle node must wait for the fourth node to finish relaying before the second node can receive another packet, as shown in Fig. 1. However, the middle node could take advantage of other paths during this period. For example, adding one more path allows the middle node to send at 0.5. The same procedure works when data flows towards the middle. The edges in Fig. 3 are labeled with the optimal transmission schedule for sending data towards the middle with two paths. It shows that data is delivered on every even-numbered slot, for a throughput of 0.5.

This procedure can be repeated for any number of paths. The complete results for up to four paths are summarized in Table 1. The results show that adding multiple paths can improve the throughput from 0.25 up to 0.8. An important note is that if carrier sensing is used, the best possible throughput when sending out from the center over two paths is limited to 0.5. This is because the middle node is no longer able to transmit while the third node is transmitting. For the rest of the paper, our results assume carrier sensing is used.

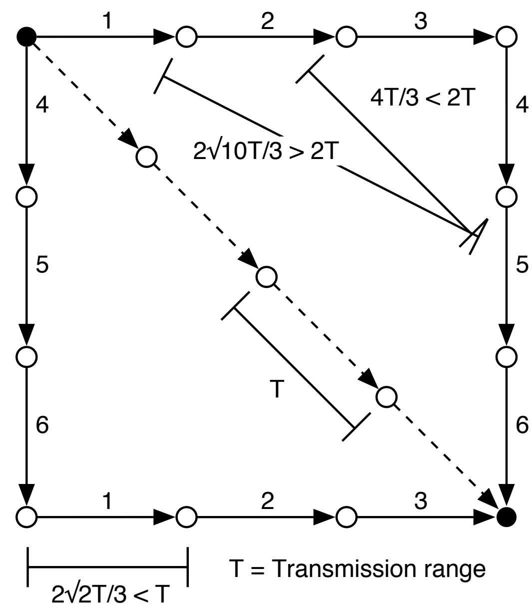

Fig. 4. The rate-limited transmission schedule for a simple 4×4 grid topology.

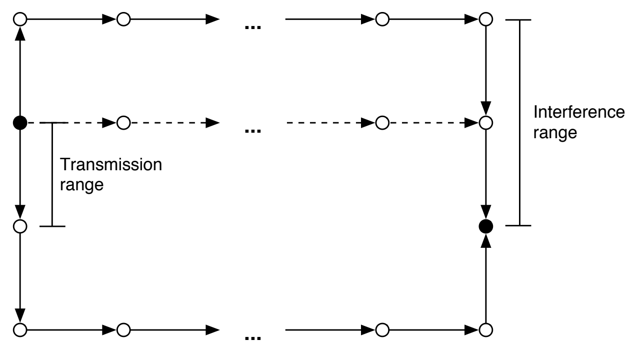

Fig. 5. Load balancing in a large grid network. The paths are forced away from the shortest path, shown with the dashed line, to ensure that the two paths are out of interference range of each other.

Unfortunately, a perfect global schedule is not easy to achieve in practice, as it is difficult to achieve the accurate clock synchronization that is required. However, it is easy to limit the sending rate. If we assume that the sender transmits at a constant rate, and all other nodes immediately forward packets we can still benefit from multiple paths. As can be seen from the results for a rate-limited sender in Table 1, the gain is less than with the global schedule. Additionally, there is no improvement with more than two paths.

This example shows that there is a potential performance benefit when using multiple paths. This topology represents examining one end of a path in isolation. Thus, the results represent an upper bound on the performance possible when using multiple paths simultaneously.

C. Grid Networks

A simple end-to-end path is shown in Fig. 4. This topology has nodes around the perimeter of a 4×4 grid and nodes placed diagonally between the source in the upper left corner and the destination at the bottom right. This results in a shortest path with four hops and two alternate paths with six hops. The same optimal schedule analysis can be done on this topology. Along any single path, the best achievable throughput is 0.25. However, using the two edge paths at the same time gives an optimal throughput of 0.5, or a rate-limited throughput of 0.333, a 33% improvement. The labels on the edges in Fig. 4 show the schedule for a rate-limited sender. This shows that gains are possible for end-to-end paths in a network that has some structure in its topology.

This technique can easily be extended to large grid networks. The destination must be at least three hops away in one direction in order to take advantage of load balancing. Any less than that and there is not enough separation between paths. The natural way to set up the paths is to use the edges of the rectangle formed with the source in one corner, and the destination in the opposite corner. If this rectangle has less than 3 hops of separation along its shortest edge, we move the paths apart. An example path is shown in Fig. 5. The two nodes have only one hop of separation along the shortest edge, so the rectangle formed by the path is extended vertically.

D. Determining Paths for Effective Load Balancing

In the previous section routes were determined using complete knowledge of the node positions. While this may be viable for some networks, in general this information is not available. One of the fundamental principles of multi-hop networks is that they must be self-configuring. To do this, we must find paths using only information that can be gathered in the network.

In order to find paths that do not interfere, Wu and Harms define a metric called correlation [13]. The *correlation* of two paths is the number of links between the nodes on the separate paths. Wu and Harms show that as the correlation between two paths increases, the average delay of flows using both paths increases. While they do not report any throughput results, this is still a useful measure of the amount of interference between paths. Logically, higher correlation corresponds to a higher probability of collision, which increases delay and decreases throughput.

Wu and Harms' definition of correlation implicitly assumes that the interference range is equal to the transmission range. To generalize their definition, we define *interference correlation* as the number of interference links between two node-disjoint paths, where an interference link exists between two nodes that are within interference range. An example of two paths with an interference correlation of five is shown in Fig. 6. Interference correlation is equivalent to correlation if the transmission and interference ranges are the same.

Intuitively, we wish to find the pair of paths between the source and destination that minimize the interference correlation. If there is a tie, we select the paths with the smallest sum of hop counts. While the hop count does not affect optimal scheduling, with more realistic models it has been shown that the throughput decreases as the path length increases [1].

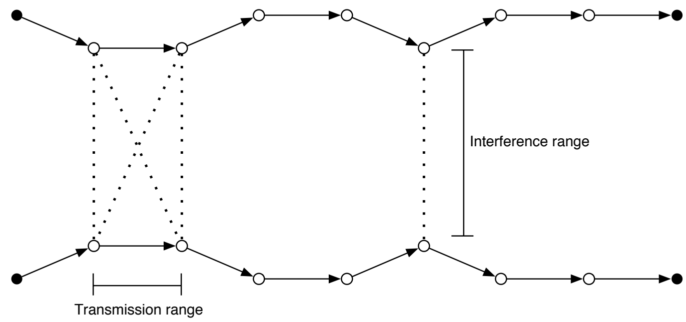

Fig. 6. Two paths with an interference correlation of five.

Additionally, we do not want to use multiple paths when there is no possible gain. From the analysis of the cross configuration, we have an upper bound on the possible gain, and we know the minimum hop count required before this gain can be realized. This is because the cross configuration represents an idealized version of multiple flows joined at one end. Joining the flows at both ends can only degrade the performance. Thus, we only search for multiple paths if the shortest path is longer than the minimum path length derived from the cross configuration.

The last hurdle is how do we know if an interference link exists between two nodes? Unlike connectivity, it is not obvious how to determine interference in a wireless network. Correlation can be computed because it relies solely on transmission range. In order to compute interference correlation, we need to estimate the distance between nodes. To do this, we assume that the shortest path between two nodes is composed entirely of maximum range hops. This is a reasonable assumption because minimizing the hop count tends to maximize the hop length. Thus, we estimate the distance between nodes to be the shortest path hop count multiplied by the transmission range. If this estimate is less than the interference range, an interference link exists between the two nodes. To compute this, each node requires the complete topology of the network, which is possible using a link-state routing protocol.

The pseudocode for the correlation path selection heuristic is shown in Fig. 7. It assumes that there is some algorithm that locates all possible paths between two nodes. The heuristic selects almost the same paths in grids as the manual routing described earlier, but it does not require any knowledge of the node positions.

IV. Simulation Experiments

We have shown that with a simple interference model and optimal scheduling, there is a benefit to using multipath load balancing with a single flow in two scenarios: a single node communicating with multiple gateways and networks with a structured topology. However, it is unclear how these results apply with a more realistic interference model and realistic transmission scheduling. In this section we answer this question by simulating the previous scenarios using ns-2. We use ns-2's interference model and the 802.11 MAC protocol for packet scheduling.

```
path: An ordered set of nodes that the data transmissions follow from the source to the destination.
path.length: The number of hops in the path, Equal to the number of nodes in the path minus one.
interferenceRangeHops: The number of maximum range hops equivalent to the interference range.
minimumMultipathHops: The minimum number of hops required to gain a benefit from load balancing.
route( source, destination ): Returns the set of all possible paths from source to destination.
allPairs( set ): Returns all possible combinations of two elements chosen from set.

function computeCorrelation( pathOne, pathTwo ):
    correlation ← 0
    for node in pathOne:
        for otherNode in pathTwo:
            if node = otherNode or shortestPath( node, otherNode ).length ≤ interferenceRangeHops:
                correlation ← correlation + 1
    return correlation

function findMultipaths( source, destination ):
    multipaths ← {}
    minimumCorrelation = ∞
    minimumHopSum = ∞

    if shortestPath( source, destination ) < minimumMultpathHops:
        return {}

    for pathOne, pathTwo in allPairs( route( source, destination ) ):
        hopSum ← pathOne.length + pathTwo.length
        correlation ← computeCorrelation( pathOne, pathTwo )
        if correlation < minimumCorrelation or (correlation = minimumCorrelation and hopSum < minimumHopSum):
            multipaths ← { pathOne, pathTwo }
            minimumCorrelation = correlation
            minimumHopSum = hopSum
    return multipaths
```

Fig. 7. The correlation path selection heuristic. The heuristic selects a pair of paths to use for multipath load balancing.

A. *Simulation Interference Model and Packet Scheduling*

We use the IEEE 802.11 MAC protocol because it is the most common wireless MAC. 802.11 is a family of wireless networking protocols based on carrier sense multiple access (CSMA), similar to Ethernet. Before transmitting, the sender performs carrier sensing. If it determines that the medium is busy, it waits a random period before retrying. Unlike wired Ethernet, collisions can only be detected at the receiver because of the nature of radio transmissions. Thus, all packets are acknowledged. If an acknowledgement is not received, the sender assumes there was a collision, and waits a random time before trying again. Like wired Ethernet, 802.11 uses exponential back off. This means that when multiple collisions occur, stations wait longer periods of time before retransmitting.

For a more realistic interference model we rely on ns-2's power-capture model. This model is still simple, but much more complex than the protocol model. To determine if a transmission is successful, the computed signal-to-noise ratio at the receiver must be above a predetermined threshold. All other transmissions are treated as noise. This model permits transmissions to be received at a distance of 250m, provided there are no other transmissions, and interference propagates up to 550m [16].

In all experiments, the data rate is set to 1 Mbps, the lowest rate supported by 802.11b. Results at higher rates are equivalent, just scaled to reflect the higher transmission rate. RTS/CTS is disabled because recent work in this area shows that it over reserves the wireless channel and leads to lower throughput [17]. All routes are assigned manually, eliminating any routing protocol overhead. When multiple paths are available, the load is spread evenly across all of them using per-packet round-robin load balancing.

The throughput measurements count all network layer bytes. This includes the IP headers but excludes the link layer headers. The tests are performed with unidirectional constant bit rate sources, using a wide range of data rates. The reported values are the ones that produced the maximum throughput. Delay is measured from end-to-end, and includes all MAC and queuing delays. A uniformly distributed jitter was added to packet transmission times to avoid self-synchronization, a phenomenon that can cause large numbers of packet drops at low rates if everything is timed perfectly.

B. *Cross Configuration Results*

The first experiment is to verify the results for the cross configuration shown in Fig. 3. Data is sent both to and from the middle node. The number of hops from the center node is varied from one to five, and the number of flows is varied from one to four. When using two flows, the paths opposite from each other are used in order to have the most physical separation possible.

As shown in Fig. 8, as the number of hops increases beyond three, the throughput for a single flow continues to decrease but the throughput for multiple paths stays relatively constant. The multipath throughput stays near 0.5 when sending in, and says near 0.333 when sending out. The primary throughput increase happens when adding the second path. When sending data towards the center over five hops, using two paths increases the throughput by 101%. There are small gains when

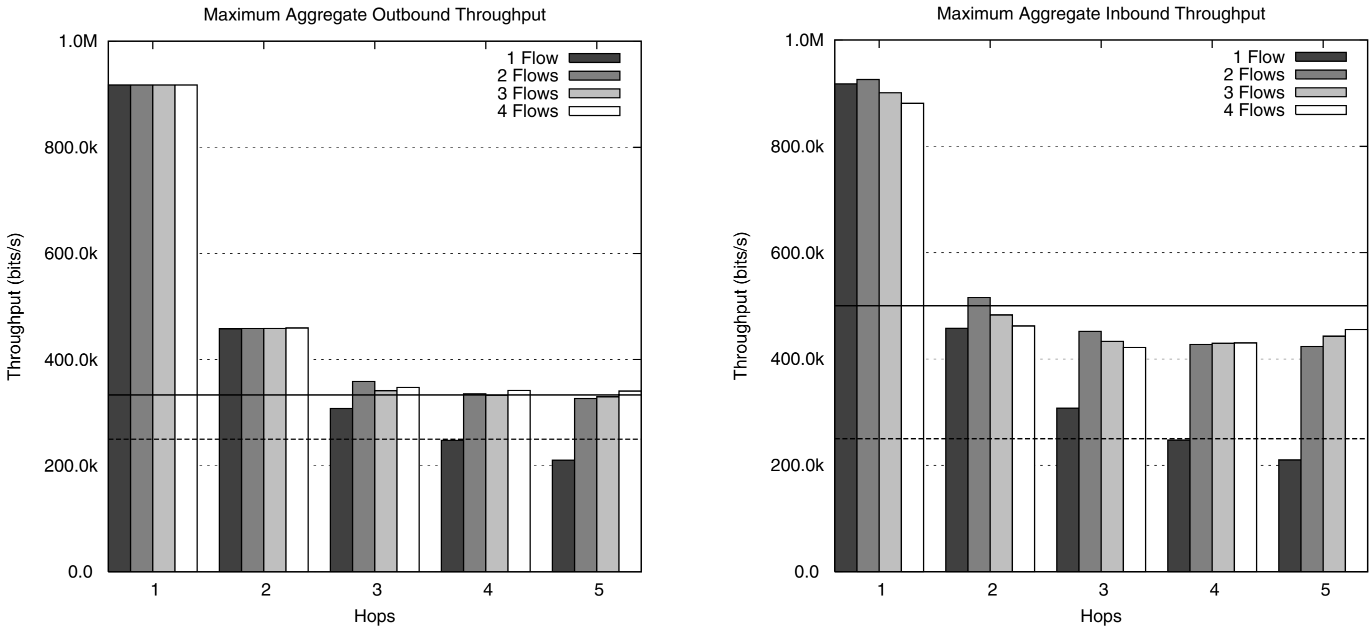

Fig. 8. Cross configuration maximum aggregate throughput. The solid lines mark the analytical multipath throughput for an infinite length chain with rate limiting, and the dashed lines indicate the analytical throughput for a single infinite length flow.

TABLE II
4×4 GRID MULTIPATH PERFORMANCE

Metric	Single Path	Edge Path	Multipath
Path Length (hops)	4	6	6
Throughput (bits/s)	252 720	196 440	267 840
Avg .Delay at 120 kbps	54.4 ms	80.8 ms	78.9 ms

using more than two flows with long paths.

The results from the analytical model for a rate-limited sender, shown in Table 1, predict that the maximum rate when sending data towards the center was 0.5, and for sending out from the center was 0.333. These rates are marked with solid lines in Fig. 8. The simulation results closely approximate the analytical predictions. In some case, the actual transmission rates exceed the prediction, presumably because random jitter can occasionally produce a globally optimal schedule.

The average delay for sending data towards the center with five hops is shown in Fig. 9. The cases with fewer hops follow the same trends, but the differences are more obvious with more hops. At low data rates, the delay increases slightly with more flows because more nodes are contending for the medium at the center node, and hence there is a higher probability of collision. Before saturation, the delay increases roughly linearly with the throughput, at a similar rate for all numbers of flows. Once the network is saturated, the delay increases dramatically because the queues at each node build up. At this point, nodes begin dropping a large percentage of packets and the network is basically unusable.

The simulation results from the test closely match the analytical results from the simpler model. They indicate that limiting the sending rate is an effective approach, even with the somewhat random scheduling of 802.11.We conclude that a path must be four hops or longer before using multiple paths can be beneficial. With rate limiting, there is only a marginal benefit for using more than two paths, so it is not worth the additional effort. There is a trade-off for the increase in throughput. Even if path lengths are unchanged, sending data with multiple flows has slightly higher end-to-end delay because of an increased probability of collision.

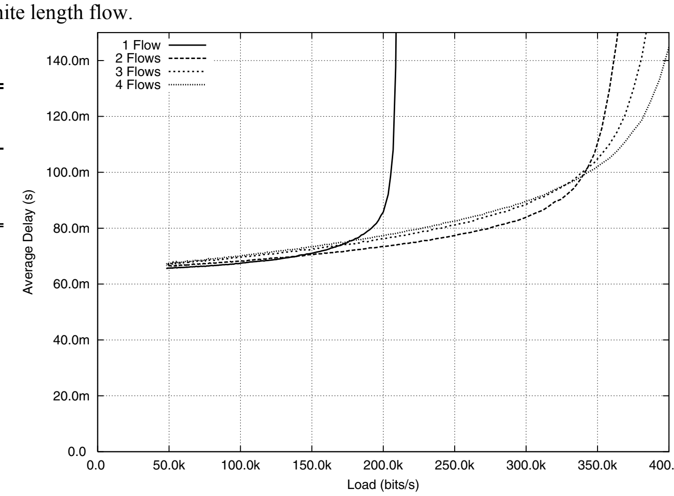

Fig. 9. Cross configuration five hop average end-to-end delay. When the throughput is low, more flows produces a larger delay due to the increase probability of collisions.

C. Simple 4×4 Grid

Next, we examine the performance of the grid shown in Fig. 4. In the previous analysis, the simple model predicted that the shortest path would get a throughput of 0.25, and using the two edge paths would get a throughput of 0.333 with a rate-limited sender, for an improvement of 33%. Unfortunately, the throughput improvement over the shortest path, shown in Table 2, is only 6%, and the delay increases by 45%. The cause for this discrepancy is that a six-hop chain has

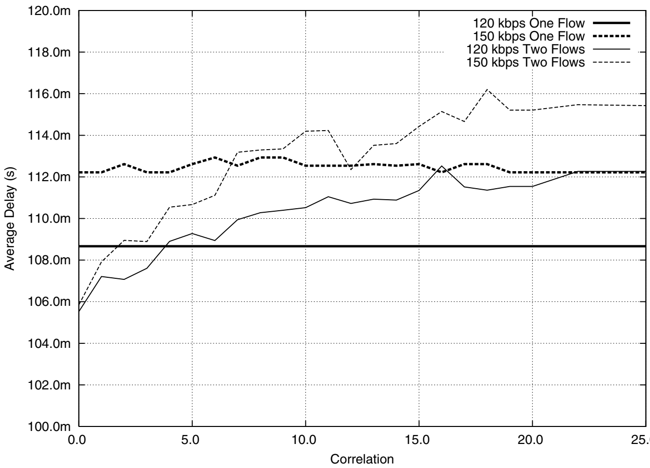

Fig. 10. Average end-to-end delay versus interference correlation. The delay increases with the correlation.

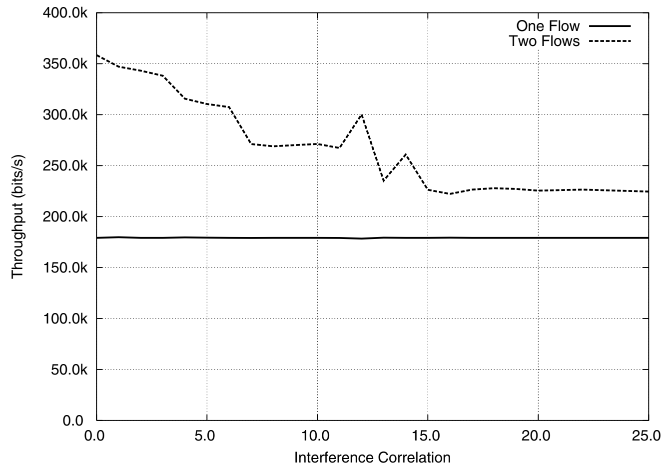

Fig. 11. Maximum aggregate throughput versus interference correlation. The throughput decreases with the correlation.

lower throughput than a four-hop chain, due to inefficiencies in the 802.11 protocol [1]. The analytical model does not take this into account. If we compare the multipath performance to the path along the edge, the throughput improves by 36%, which closely matches the analytical model.

D. *Impact of Interference Correlation on Throughput*

The previous test showed that results from the simpler model presented earlier can still be relevant in the more complex model used by ns-2. However, in order for our path-selection heuristic to be useful, we need to investigate the effects of interference correlation on throughput and delay.

To do this, we test the performance of two parallel paths with varying interference correlation values. The paths have eight hops with nodes spaced 220m apart. The paths are initially 600m apart, just out of interference range. To increase the correlation, certain pairs of nodes are moved within 480m or 520m of each other. For example, the path shown in Fig. 6 has an interference correlation of five. The minimum interference correlation along these paths is zero, and the maximum is 25, when the paths are 480m apart.

The average delay for sending data along these paths at two data rates is shown in Fig. 10. Initially, the end-to-end delay is lower with two paths. This is because the load is divided in half across the two paths, and delay in a chain increases linearly with throughput, as shown in Fig. 9. However, the delay quickly exceeds the single path case as the interference correlation increases. This verifies that Wu and Harms' result still holds for interference correlation, under ns-2's model of interference. The throughput decreases as the interference correlation increases, as shown in Fig. 11. At the highest level of interference correlation in this test, the two paths remain outside of transmission range. In this configuration, the interference is not strong enough to prevent all gains, but it does decrease the throughput by 75% compared to the initial configuration of two paths with no interference. This shows that some interference correlation between the paths can be tolerated for both delay and throughput, but it can easily limit the improvement. Most importantly, it shows that the effects of the interference range cannot be ignored.

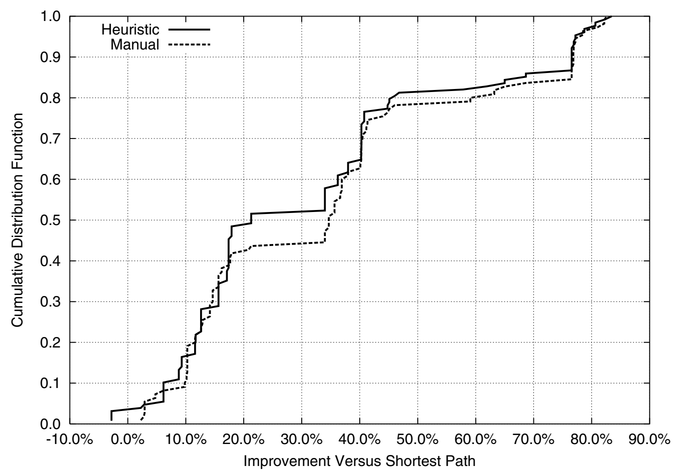

Fig. 12. Cumulative distribution function of the throughput improvement. The performance of the manual and the heuristic schemes is very similar.

E. *Effective-Load Balancing Paths in Structured Networks*

Since we have verified that the model we are using is sound, and that interference correlation has an impact on throughput, we now test our heuristic for effective load-balancing paths in structured networks. We examine a 5×5 grid network where nodes are spaced one transmission range apart. We test the performance of flows between all pairs of nodes in the network. We use the shortest path, the pair of paths found by the correlation path selection heuristic, and the pair of paths found by manual routing with the complete node-location information. In this network, there are 25×24=600 (source, destination) pairs.

The cumulative distribution function of the improvement over the shortest-path routing is shown in Fig. 12. This figure shows that both routing schemes can find some paths that significantly improve the throughput, up to approximately 80%. Half the paths found have greater than 35% improvement. The diagram also shows that manual routing slightly outperforms the heuristic. The heuristic even finds four paths that degrade the performance by an average of 2.8%. However, the performance of the two schemes is very similar. The average improvement with manual routing is 35% compared with 32% for the heuristic. The heuristic was able to use load balancing for nearly half the paths in the entire network (44%). A summary of the results is shown in Table 3.

F. Multiple Simultaneous Flows

We have shown that load balancing can improve the performance of single flows. However, these paths can require more hops than the shortest path in order to route around interference. There are potentially more transmissions for each packet, and intuitively that should degrade the performance of the overall network. Contrary to this logic, previous work has shown that a perfect load balancing scheme can improve the total capacity of a network, because it spreads the load evenly throughout the entire topology [4]. Thus, it is unclear what impact the multipath routing will have on larger networks.

To investigate this issue, we simulate a set of 10×10 grids with different traffic loads. Sources and destinations are selected randomly. However, if a pair is too close together to use multiple paths, they are discarded and a new pair selected. Seven networks of twenty sources and destinations are generated. These networks are simulated with five, ten, fifteen and twenty simultaneous flows. The aggregate throughput results are shown in Fig. 13.

In the majority of the cases, multipath routing performs better. When it is worse than the single path routing, it is only slightly worse. This seems to indicate that multipath routing can improve the performance of large structured networks.

These results are limited in two important ways. First, all sources are sending at the same rate. This means the throughput is to some extent limited by the slowest flow. Other flows might be capable of sending faster, meaning that the actual maximum throughput could be higher. The multipath case has a lower per flow throughput, so if a single flow is a bottleneck, there are twice as many flows, potentially producing a higher aggregate throughput. The second limitation is that in grid networks, most of the multipath routes have the same length as the shortest path. It uses longer paths only when the source and destination are near the same horizontal or vertical line. Hence, the problems caused by additional hops will not be severe. Therefore, it would be unrealistic to generalize these results to arbitrary topologies.

TABLE III
LOAD BALANCING ROUTING SCHEME PERFORMANCE COMPARISON

Criteria	Manual Routing	Heuristic
Paths Found	110/300 (37.6%)	128/300 (43.8%)
Paths Improved	110	124
Paths Degraded	0	4
Min. Improvement	2.24%	-2.81%
Avg. Improvement	32.0%	34.6%
Max. Improvement	83.5%	82.6%

V. CONCLUSION

This paper presents a study on the performance of multipath load balancing in multi-hop wireless networks. Our results show that it is possible to use multiple paths to increase the throughput of single flows, at least in two specific scenarios. The first scenario is a cross topology. In this scenario, our results show that by using a second path, the throughput can be improved by up to 100%. If a global packet schedule could be applied, it would be possible to further improve the throughput. Using the simpler option of rate-limiting senders, adding more paths beyond two only has a small incremental improvement. These results indicate that multipath routing could be useful for networks where all communication goes through a set of gateways, such as Internet access networks. If a node uses multiple gateways simultaneously, the communication in these networks looks similar to the cross topology.

The second scenario where multipath load balancing is useful is in wireless networks with a structured topology. In the grid networks studied here, we show that a performance increase of up to 80% is possible by using multiple paths. We also show that some performance increase is possible when there are multiple data flows.

This work is an initial investigation of the fundamental requirements for using multipath load balancing in multi-hop wireless networks. In order to make these results more generally applicable, they must be extended to arbitrary topologies, and the interactions with TCP must be studied, since is the most common transport layer protocol. We are actively investigating the multiple gateway scenario, since our results indicate that significant performance gains may be possible.

VI. ACKNOWLEDGEMENTS

We would like to thank Sonia Waharte for finding an error in a draft of this paper.

REFERENCES

[1] J. Li, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris, "Capacity of ad hoc wireless networks", in *Proc. 7th ACM Int. Conf. on Mobile Computing and Networking (MobiCom)*, Rome, Italy, July 2001.

[2] C. E. Perkins, and P. Bhagwat, "Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers", In *Proc. SIGCOMM '94*, August 1994.

[3] C. E. Perkins, and E. M. Royer, "Ad hoc on-demand distance vector routing", In *Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, USA, February 1999.

[4] P. P. Pham, and S. Perreau, "Performance analysis of reactive shortest path and multi-path routing mechanism with load balance," in *Proc. of IEEE INFOCOM*, 2003.

[5] M. K. Marina and S. R. Das, "On-demand multipath distance vector routing in ad hoc networks," *in Proc. Int. Conf. for Network Procotols (ICNP)*, pp. 14–23, Nov. 2001.

[6] A. Nasipuri, R. Castaneda, and S.R. Das, "Performance of multipath routing for on-demand protocols in mobile ad hoc networks," *Mobile Networks and Applications (MONET)*, vol. 6, no. 4, pp. 339–349, 2001.

[7] Z. Ye, S. Krishnamurthy, and S. Tripathi, "A framework for reliable routing in mobile ad hoc networks," in *Proc. of IEEE INFOCOM*, 2003.

[8] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proc. 9th ACM Int. Conf. on Mobile Computing and Networking (MobiCom)*, San Diego, CA, USA, Sept. 2003.

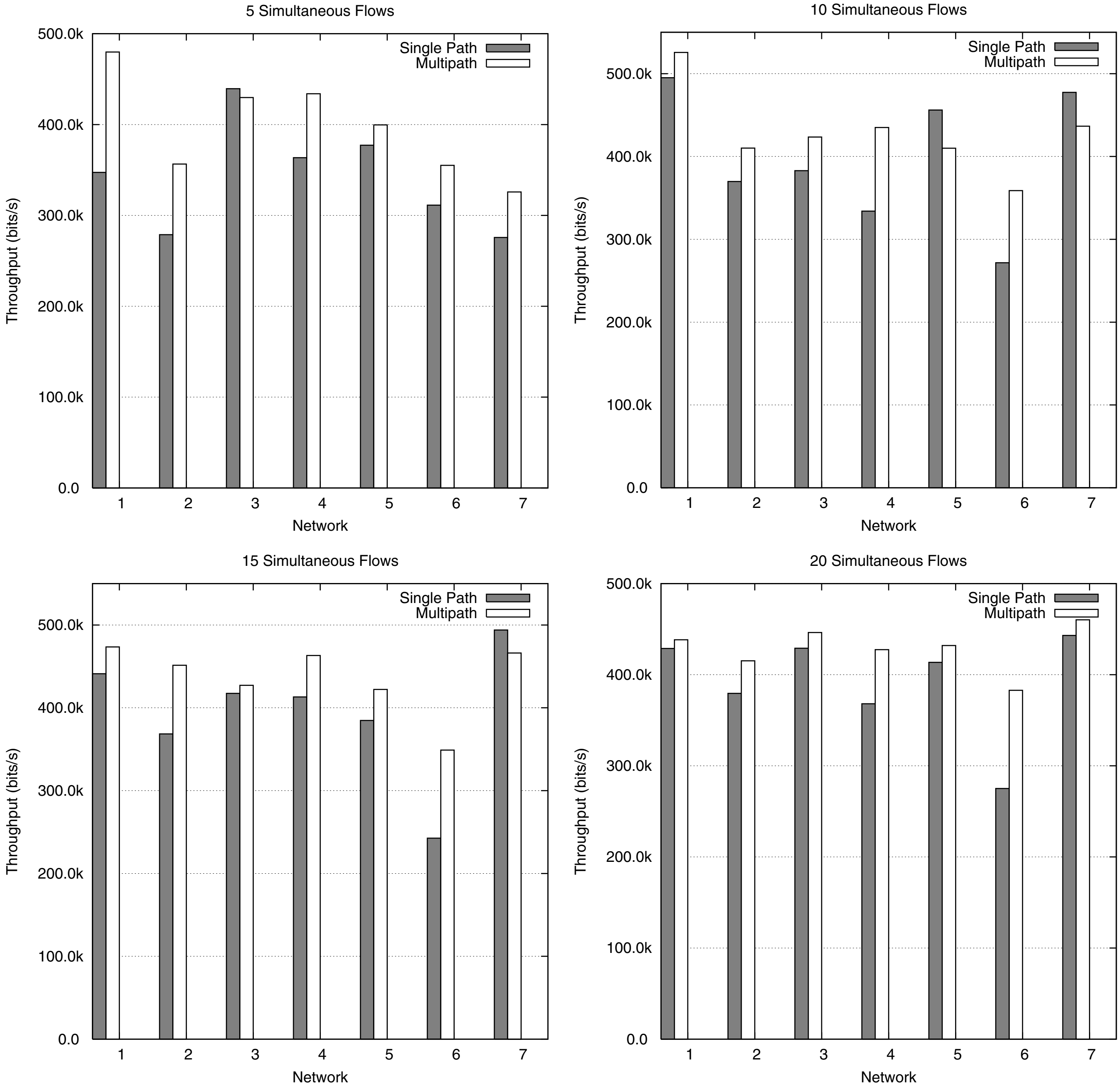

Fig. 13. Grid network aggregate throughput. This shows that using multipath routing can improve the aggregate throughput of grid networks in the majority of cases. It appears that multipath performance depends strongly on the traffic pattern, as all of these scenarios have the same topology, but use different paths.

[9] L. Wang, L.F. Zhang, Y.T. Shu, M. Dong, and O.W.W. Yang, "Multipath source routing in wireless ad hoc networks," in *Proc. IEEE CCECE*, p. 479, 2000.

[10] A. Valera, W. Seah and S.V. Rao, "Cooperative packet caching and shortest multipath routing in mobile ad hoc networks," in *Proc. IEEE INFOCOM*, 2003.

[11] S. Roy, S. Bandyopadhyay, T. Ueda, and K. Hasuike, "Multipath routing in ad hoc wireless networks with omni directional and directional antenna: a comparative study," in *Proc. 4th Int. Workshop on Distributed Computing, Mobile and Wireless Computing (IWDC)*, pp. 184-191, 2002.

[12] M.R. Pearlman, Z.J. Haas, P. Sholander, S.S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc network," in *Proc. 5th ACM Intl. Symposium on Mobile Ad Hoc Networking and Computing* (*MobiHOC)*, p. 150, 2000.

[13] K. Wu and J. Harms, "Performance study of a multipath routing method for wireless mobile ad hoc networks," in *Proc. 9th Int. Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS '01)*, Aug. 2001.

[14] Y. Ganjali and A. Keshavarzian. "Load balancing in ad hoc networks: single path routing vs. multi-path routing", in *Proc. of IEEE INFOCOM*, 2004.

[15] N. T. Nguyen, A. A. Wang, P. Reiher, and G. Kuenning, "Electric-field-based routing: a reliable framework for routing in MANETs," *SIGMOBILE Mobile Computing Communications*, vol. 8, no. 2, pp. 35-49, 2004.

[16] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proc. 4th ACM Int. Conf. on Mobile Computing and Networking (MobiCom)*, Dallas, Texas, USA, pp. 85–97, 1998.

[17] S. Ray, J. B. Carruthers, and D. Starobinski, "RTS/CTS induced congestion in ad hoc wireless LANs", In *Proc. Wireless Communications and Networking Conference (WCNC)*, March 2003.