# How to evaluate exotic wireless routing protocols?*

Dimitrios Koutsonikolas[1], Y. Charlie Hu[1], Konstantina Papagiannaki[2]
[1]Purdue University and [2]Intel Research, Pittsburgh

## ABSTRACT

The advent of static wireless mesh networks (WMNs) has largely shifted the design goals of wireless routing protocols from maintaining connectivity among routers to providing high throughput. The change of design goals has led to the creation of many "exotic" optimization techniques such as opportunistic routing and network coding, that promise a dramatic increase in overall network throughput. These "exotic" techniques have also moved many mechanisms such as reliability and rate control, that used to be below or above the routing layer in traditional protocols, to the routing layer.

In this paper, we first review the above evolution of routing protocol design and show that the consolidation of mechanisms from multiple layers into the routing layer poses new challenges to the methodology for evaluating and comparing this new generation of routing protocols. We then discuss the diverse set of current practices in evaluating recently proposed protocols and their strengths and weaknesses. Our discussion suggests that there is an urgent need to carefully rethink the implications of the new merged-layer routing protocol design and develop effective methodologies for meaningful and fair comparison of these protocols.

Finally, we make several concrete suggestions on the desired evaluation methodology. In particular, we show that the traffic sending rate plays a fundamental role and should be carefully controlled.

## 1. RENAISSANCE OF WIRELESS ROUTING PROTOCOL DESIGN

The recent evolution of wireless networking from the ad hoc networking era to the mesh networking era has ignited a new Renaissance of routing protocol design for multihop wireless networks.

In the ad hoc networking era, the primary challenge faced by routing protocols (e.g., DSR [11], AODV [17]) was to deal with frequent route breaks due to host mobility in a dynamic mobile environment. Accordingly, most research efforts were focused on designing efficient route discovery/repair schemes to discover or repair routes with minimum overhead. The routing process itself was simple; once a route from the source to a destination was known, each hop along the route simply transmitted the packet to the next hop via 802.11 unicast. These protocols relied on 802.11 unicast (with its built-in ACK-based local recovery scheme and exponential backoff) to deal with packet loss due to channel errors or collisions.

The design goals of the ad hoc routing protocols also drove their evaluation methodology. The comparison between different protocols was usually in terms of Packet Delivery Ratio (PDR) and control overhead (e.g. [2, 5]). The offered load, typically of some constant rate, was low so that the resulting data traffic and control overhead do not exceed the network capacity. The main parameter varied in the evaluations was the *pause time* of the random waypoint mobility model, which characterized how dynamic the environment was. The focus of such a methodology was to offer a direct comparison of various protocols' ability to transfer data to the destination under host mobility, while incurring low control overhead. Interestingly, often times the protocol comparisons boiled down to tradeoffs between PDR and control overhead [2, 5].

Transition to WMNs changed these rules. In a WMN, routers are static and hence route changes due to mobility are not a concern anymore. The main performance metric is now *throughput*, often times even at the cost of increased control overhead.

The first major effort towards the new design goal was on designing link-quality path metrics (e.g., ETX [4], ETT [6]) that replaced the commonly used shortest-path metric. The protocols using these link-quality metrics still followed the layering principle: the routing layer finds a good route, and 802.11 unicast is used to deliver packets hop by hop.

**Opportunistic Routing.** Seeking further throughput improvement, researchers looked into new, "exotic" techniques, which largely abandoned the layering principle. The first such technique was opportunistic routing as demonstrated in the ExOR protocol [1]. Instead of having a decoupled MAC and routing layer, ExOR explored an inherent property of the wireless medium, its broadcast nature. Instead of first determining the next hop and then sending the packet to it, it broadcasts the packet so that all neighbors have the chance to hear it; among those that received the packet, the node closest to the destination forwards the packet. This also implies that some coordination is required, so that the neighboring nodes can agree on who should rebroadcast the packet next. To reduce the coordination overhead, ExOR proposed sending packets in batches.

**Intra-flow network coding.** The second "exotic" technique applied network coding to multihop wireless networks. With network coding, each mesh router randomly mixes packets it has received before forwarding them. The random mixing ensures with high probability that nodes will not forward the same packet, and hence coordination overhead is min-

*We thank the reviewers for their insightful comments. This work was supported in part by NSF grant CNS-0626703.

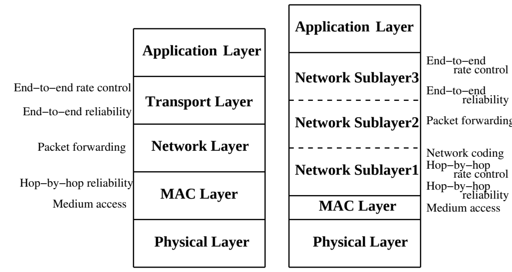
imized. Network coding has one more positive effect. It resembles traditional Forward Error Correction (FEC) techniques, which offer reliability through redundancy, with the extra advantage that it is applied at every hop, and not end-to-end [16, 8]. Together, network coding eliminates the need for reliability on a per-hop or per-packet basis. Since each coded packet contains information about many packets, the destination can reconstruct the original data if it receives sufficiently many packets. MORE [3] was the first protocol to combine opportunistic routing with network coding.

Both techniques use unreliable 802.11 broadcast as the hop-by-hop forwarding technique, which is a significant departure from traditional routing protocols. The use of broadcast is a necessity for opportunistic routing as well as effective network coding. Since the MAC now does not have to deal with retransmissions and exponential backoffs, it can send at much higher packet rates than in the unicast mode; it is essentially limited only by carrier sensing. Sending at higher rates potentially implies higher goodput. Since the design goal is focused on high throughput, this observation has an immediate implication for the evaluation methodology of these new protocols: instead of using a constant rate (CBR) of $X$ packets per second, the source node should send *as fast as the MAC allows*.

However, making the sources send as fast as the MAC allows has a serious side effect. It can cause congestion in the network if the aggregate transmission rate of the nodes exceeds the network capacity. As [14] showed, in contrast to the wired Internet, where congestion is the result of a complex interaction among many flows, in a wireless network, congestion can happen even with a single flow, in a simple topology and even with 802.11 unicast. The use of broadcast in this new generation of routing schemes simply worsens the situation, since the lack of exponential backoff in the 802.11 broadcast mode means nodes never really slow down.

**Rate control.** With congestion, the queues of the nodes become full, causing significant packet loss. We thus need to reintroduce the mechanism for preventing the network from reaching this state: rate control. SOAR [18] is a new opportunistic routing protocol that has a built-in rate control mechanism, both at the source (using a sliding window) and at intermediate routers (using small queues to avoid unpredictable queuing delays). Other protocols (e.g., [19, 9]) propose hop-by-hop, backpressure-based mechanisms to limit the amount of traffic injected in the network. Hence, *rate control, which used to be the responsibility of higher layer protocols (transport or application), is now brought down to the routing layer.*

**Inter-flow network coding.** The final frontier is that of *increasing the network capacity* itself! The basic idea is again simple: a router can XOR packets from different flows (hence inter-flow network coding as opposed to intra-flow network coding discussed previously) and broadcast them. If the next hop of each flow has already overheard all the mixed packets except for the one destined for it, it can XOR them again with the XORed packet to obtain its own packet.




(a) Traditional Network Stack (b) New Network Stack

**Figure 1**: The evolution of the protocol stack.

COPE [12] was the first protocol that brought this idea from theory into practice. By mixing packets belonging to different flows and transmitting them as one, one reduces the total number of transmissions required, and hence increases the "effective" capacity of the network.

Since the technique stretches the capacity of the network, the most natural way to show its improvement, i.e., the implied evaluation methodology, is to subject the network to a traffic load (not too much) above the physical capacity, i.e., the network should already be congested before network coding is turned on, which will then increase the effective capacity just enough to eliminate the congestion.

**Reliability.** Since 802.11 broadcast is unreliable, with the exception of intra-flow network coding, which embraces FEC, all other techniques, which rely on MAC-layer broadcast, require some ARQ-based recovery mechanism. ExOR uses end-to-end retransmissions by going through the same batch of packets until 90% of them are received by the destination; SOAR and COPE use asynchronous cumulative hop-by-hop acknowledgments; COPE also relies partly on 802.11 unicast (known as pseudobroadcast [12]). Hence, in addition to rate control, one more mechanism, *reliability, which used to be the responsibility of either upper (end-to-end) or lower (hop-by-hop) layers, is now brought to the routing layer.*

In summary, the "exotic" techniques used in new routing protocols for WMNs have largely abandoned the layering principle and adopted a merged-layer approach, as shown in Figure 1. Mechanisms that used to be at lower or higher layers are now blended into the routing layer. *This consolidation of mechanisms and techniques into the routing layer has made the evaluation of routing protocol performance a much subtler task than before*. For example, some mechanisms and techniques may be conflicting: inter-flow network coding desires traffic load to be above the network capacity while rate control targets the exact opposite.

In the next section, we discuss the resulting diverse set of current practices in evaluating this new generation of routing protocols. We show that, in contrast to traditional routing protocols, there have been no clear guidelines that drive the evaluation of these protocols; often times each new protocol is evaluated with a different methodology.

## 2. STATE OF AFFAIRS

There have been many high-throughput routing protocols for WMNs proposed over the last few years. Due to the page

**Table 1**: Methodologies used in evaluating recent high-throughput WMN routing protocols.

| | Evaluation Methodology | Example |
|---|---|---|
| **Unreliable protocols** | Make both protocols reliable but in different ways | ExOR [1] |
| | Evaluate for a wide range of sending rates, with deteriorating PDR | COPE [12] |
| | Compare a protocol with rate control against a protocol without rate control | SOAR [18] |
| | Old ad hoc methodology: keep the sending rate fixed below capacity, measure PDR | ROMER [20] |
| **Reliable protocols** | Compare a reliable protocol against an unreliable protocol | MORE [3] |
| | Compare a reliable protocol against an unreliable protocol under TCP | noCoCo [19] |
| | Modify an unreliable protocol to incorporate the same reliability mechanism of a new protocol | noCoCo [19] |

limit, we review here the evaluation methodologies used in a subset of them, as summarized in Table 1.

## 2.1 Evaluation of Unreliable Protocols

In the case of unreliable protocols (e.g., for multimedia applications that do not require 100% PDR), the main objective is high throughput perceived by the destinations, i.e., high goodput. The new trend in the evaluation methodology is to saturate the network, letting the sources send as fast as possible so that the traffic load in the network exceeds the available capacity; then measure the maximum amount of traffic the protocol can deliver to the destination.

However, such a methodology is flawed in that it completely deemphasizes the PDR metric. The fact that certain applications do not require 100% PDR does not mean that reliability is a factor that can be completely neglected. Many applications have certain lower bounds for reliability; for example the quality of a video deteriorates with packet loss, and hence if the PDR drops below a threshold, the video quality becomes unacceptable.

**Practice 1: Making both protocols reliable.** ExOR guarantees reliable end-to-end delivery of 90% of each batch; every node keeps retransmitting packets belonging to a given batch until they are acknowledged by a node closer to the destination. The last 10% of the packets could incur a lot of overhead if they were sent through ExOR, and hence they are sent through traditional routing, which does not offer any guarantee for end-to-end reliability.

The authors argued that a direct comparison of ExOR with traditional routing would be unfair and they conducted the experiments in a way that guaranteed 100% PDR with both of them. In each case, the size of the file to be downloaded was 1MB. Instead of using traditional routing to carry the last 10% of the file, the evaluation of ExOR was based on the transmission of a 1.1 MB file, so as to compensate for loss. In contrast, the traditional routing protocol was only used to determine the route offline. The 1MB file was then transfered sequentially hop-by-hop, thus eliminating collisions, and also packet drops due to queue overflows.[1]

While this methodology was largely fair, it eliminated one important feature of traditional routing that does not exist in ExOR: spatial reuse. To avoid duplicate transmissions, nodes in ExOR are assigned priorities, and only one node transmits at a time – hence, coordination is achieved at the cost of reduced spatial reuse. In contrast, with traditional routing simultaneous transmissions can take place across the network as long as they do not interfere with each other. This advantage can turn into a drawback in the presence of a large number of hidden terminals. In other words, by trying to make the comparison fair by adding reliability to traditional routing, the authors also removed one feature of traditional routing. Whether this feature harmed traditional routing depends on the particular environment used for the evaluation.

**Practice 2: No rate control - varying the sending rate.** COPE in [12] was compared against a traditional routing protocol (Srcr), under UDP traffic.[2] In an 802.11a network with a nominal bitrate of 6Mbps, the experiment was repeated for gradually increased total offered load. The aggregate throughput over the total offered load for the two protocols was then presented, as shown in Figure 2 (Figure 12 in [12]).

We make several observations on Figure 2. First, the advantage of COPE is best shown when the traffic load in the network is pushed beyond the capacity. Since it is not clear what the traffic load is, the best thing is to measure throughput for varying offered load, as done by the authors. As expected, at low loads, COPE performs similarly to traditional routing. As the load increases, COPE offers on *average* 3-4$x$ throughput improvement over traditional routing. Second, like traditional routing, the goodput of COPE also peaks when the offered load is around the effective capacity of the network (now higher because of inter-flow network coding), and decreases quickly as the load further increases, and the PDR value, which can be easily calculated by dividing the y value by the x value, deteriorates sharply, possibly below the acceptable level of many applications. Third, if the protocols have rate control mechanisms, ideally the goodput should remain constant when the offered load is increased to beyond the network capacity. Since neither protocol has rate control, we witness the decline of the goodput.

**Practice 3: Comparing a protocol with rate control against a protocol without.** SOAR applies sliding window-based rate control at the sources, trying to discover the optimal sending rate online. In contrast, traditional routing has no rate control. This immediately creates a challenge for a fair comparison of the two protocols. Faced with this challenge, the authors decided to perform the evaluation in a saturated network, where each source transmits at 6Mbps, same as the nominal bitrate of the network.

[1]The packet losses due to channel errors were masked in the testbed through 802.11 retransmissions.

[2] [12] also evaluated COPE and Srcr under TCP. In that case, although the two protocols are unreliable, reliability is provided by the transport layer.

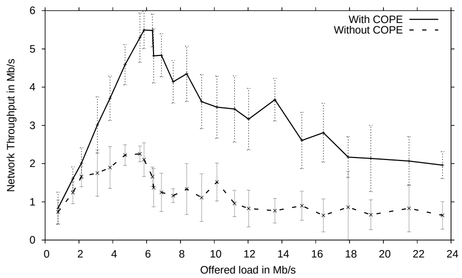



**Figure 2**: Evaluation of COPE and traditional routing in an ad hoc network for UDP flows. Reproduced Figure 12 from [12].

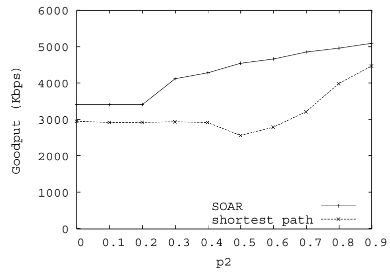



**Figure 3**: Evaluation of SOAR and traditional routing: 4 parallel flows in a grid topology. Reproduced Figure 14(b) from [18].

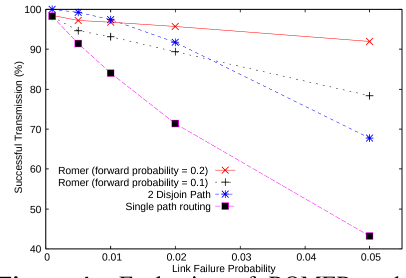



**Figure 4**: Evaluation of ROMER and traditional routing: PDR over link failure probability. Reproduced Figure 5 from [20].

Saturating the network creates an adverse situation for traditional routing, which is expected to perform poorly under these conditions and suffer significant packet loss due to queue overflows. In contrast, SOAR adapts the sending rate online, based on the network conditions. SOAR was shown to offer a large throughput improvement over traditional routing (one example is shown in Figure 3, Figure 14(b) in [18]). However, it is not clear what part of the improvement is because of the opportunistic forwarding and what part is because of the rate control.

**Practice 4: Old methodology (for evaluating ad hoc protocols).** ROMER [20] is another opportunistic routing protocol which exploits link bitrate diversity in order to maximize throughput. It uses the 802.11 unicast autorate adaptation mechanism, and tries to send traffic over high rate links, in contrast to ExOR and SOAR, which always use a fixed link bitrate. ROMER was evaluated under yet another methodology different from ExOR and SOAR. The authors compared the PDR (and throughput gain) achieved by ROMER over traditional routing, following the old methodology for ad hoc protocol evaluation. The parameter varied is the link failure probability, while the source sending rate is kept constant (and the value is unclear).

Due to the autorate adaptation, it is difficult to estimate the capacity of the network used for the evaluation. The high delivery rates achieved (at least) by ROMER (in Figure 4, Figure 5 in [20]) make us conjecture that the sending rate was not high enough to congest the network, in contrast to in [18] and [12]. However, a single sending rate does not reveal the maximum gain achieved by ROMER, in particular if this rate is far below the capacity of the network.

## 2.2 Evaluation of Reliable Protocols

Traditional routing protocols left to the transport layer the responsibility for end-to-end reliability. However, TCP, the de facto reliable transport layer protocol for the wired Internet, has been reported to perform poorly in multihop wireless networks [7, 13, 10], especially in environments with many hidden terminals and highly lossy links. The reason is that TCP performs congestion control in addition to reliability and correlates these two mechanisms. High packet loss causes TCP flows to suffer timeouts and excessive backoff, and it prevents them from increasing their window size and utilizing the wireless medium efficiently. This is the reason many new protocols ignore TCP, and incorporate mechanisms for end-to-end reliability at the network layer instead.

**Practice 5: Comparing a reliable with an unreliable protocol.** In [3], MORE is compared against traditional routing showing a median throughput gain of 95%. The authors used UDP traffic for both protocols sent at the maximum possible data rate, i.e., the source transmitted as fast as the MAC allowed. As we have already explained, in a highly congested environment, 802.11 unicast cannot help traditional routing to recover from packet drops due to queue overflows. In contrast, with MORE there is no queuing. With a batch size of $k$ packets, every MORE router only needs to keep $k$ linearly independent packets in a buffer; linearly dependent packets do not include any new information and can be safely dropped. Hence, a MORE router does not experience losses due to queue overflows, no matter how fast it receives packets from its upstream nodes. In addition, the FEC element contained in network coding masks packet losses due to collisions and channel errors through redundancy. Thus, a reliable protocol was compared against an unreliable one.

This does not necessarily mean that the comparison favored MORE over traditional routing. In the evaluation of the two protocols, a fixed size file was sent from the source to the destination with each protocol, however with traditional routing only a fraction of this file is finally delivered to the destination. Depending on the fraction of the file that is lost and the time taken for the transfer, this evaluation could favor any of the two protocols. In other words, adding an end-to-end reliability mechanism to traditional routing would increase the numerator of the throughput formula (the amount of data delivered) but it would also increase the denominator (the time taken for the total transfer); this could lead to either an increase or a decrease to the throughput achieved with traditional routing.

**Practice 6: Running an unreliable protocol under TCP.** An easy way to provide end-to-end reliability with an unreliable routing protocol is to run it under TCP; no change is required to the protocol itself. This is one of the approaches followed by [19] in the evaluation of noCoCo. noCoCo improves COPE by scheduling the transmissions at the nodes in order to maximize the gain from inter-flow network coding. Coupled with scheduling in noCoCo is a backpressure, hop-by-hop congestion control mechanism. This mechanism eliminates queue overflows and packet dropping and guarantees end-to-end reliable packet delivery. Hence, in noCoCo, sources do not transmit as fast as the MAC allows; their sending rates are limited by the congestion control mechanism.

In the evaluation, noCoCo was compared against COPE [12] and traditional routing. The main goal was to quantify the gains of coordinated network coding used in noCoCo against opportunistic network coding, used in COPE. TCP was used with COPE and traditional routing to provide reliability (and congestion control) at the transport layer. However, TCP is known to perform poorly in multihop wireless networks; in addition, it was shown to interact poorly with COPE and limit the coding opportunities and consequently the throughput gain [12]. Hence, this methodology again blurred the true gain from coordinated coding, since different congestion control and reliability mechanisms are used. The authors acknowledged this point and noted that it should be taken into account when trying to interpret the results.

**Practice 7: Modifying an unreliable protocol.** To finally isolate the gain from coordinated coding, the authors of noCoCo also modified traditional routing and COPE to use the same backpressure-based algorithm for congestion control and reliability, thus removing the negative side-effects of TCP.

### 2.3 Use (or No Use) of Autorate Adaptation

802.11 unicast allows a sender to change the bit rate automatically, based on the quality of the link to the receiver. On the other hand, the majority of the "exotic" optimization techniques are based on 802.11 broadcast, and hence most of the new routing protocols based on these techniques (with the exception of ROMER) do not use autorate adaptation. For "fair" comparison, the evaluation of these protocols often disables autorate adaptation for the traditional, unicast routing, e.g., in [1, 12, 18, 19] (one notable exception is [3]). We argue the contrary; the methodology is unfair to traditional routing if it can benefit from autorate adaptation.

## 3. RECOMMENDATIONS

We have witnessed the inconsistencies in the current evaluation methodologies of the new generation of routing protocols. In the following, we make recommendations for more consistent and meaningful evaluation methodologies.

**The importance of rate control.** Rate control is fundamental for the optimal operation of any (unreliable or reliable) protocol, as it ensures that the traffic load does not exceed the network capacity limit.

Figure 5 shows our envisioned throughput performance for well designed unreliable protocols. Traditional routing under UDP has no rate control mechanism incorporated. When the offered load exceeds the network capacity, packets start getting dropped due to congestion, possibly reducing the throughput much below its possible maximum value. New protocols with "exotic" techniques are expected to offer a dramatic increase to the throughput; they can even increase the capacity bound (e.g., from inter-flow network coding). However, without rate control, congestion can build up and throughput will also start decreasing when the (new) capacity point is exceeded. By adding appropriate rate control, the goodput is expected to remain constant when the offered load is beyond the capacity. One implication of this design guideline is that there may be no need to vary the offered load beyond the capacity point any more.

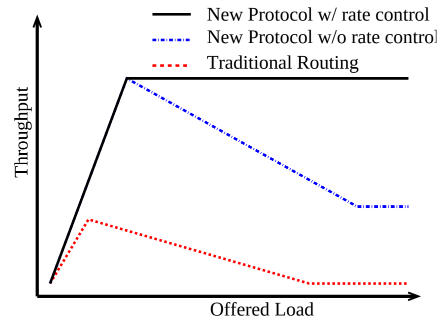


**Figure 5**: Envisioned throughput performance for well designed unreliable protocols (with built-in rate control), in contrast to traditional routing and high-throughput protocols without rate control.

For reliable protocols, PDR remains 100% but the argument for rate control is still valid. When reliability is provided through the traditional way (ARQ), some rate control is implicitly imposed, since retransmissions are given priority over new packets. However, when reliability is part of the "exotic" technique (e.g., intra-flow network coding embraces FEC), the source may never slow down, unless explicitly forced by rate control. In any case, exceeding the capacity of the network will lead to unpredictable behavior which will appear either in the form of increased delays, severe unfairness among flows, or reduced throughput. As an example, the gain of MORE over traditional routing in [3] is reduced in the presence of multiple flows. A related recommendation is that a protocol should also be evaluated with multiple flows, e.g., as in [3, 18], as the rate control for each flow becomes more challenging.

Note that the best method for applying rate control in wireless networks is still an open problem and is out of the scope of this paper. In general, online mechanisms (both end-to-end, e.g., sliding-window based [18], and hop-by-hop, e.g., backpressure based [19, 9]) or even offline computations [14] can be applied.[3]

**Isolating the benefit from new optimization techniques.** The evaluation of a new protocol that exploits a new optimization technique should try to isolate the gain from this "exotic" technique, alone. The tricky part here is that in adding a new optimization technique, a new protocol often incorporates other old techniques brought down to the rout-

[3]Interestingly, the importance of rate control has attracted significant interest in recent years in the theory community in the form of cross-layer optimizations (e.g. [15]).

ing layer from the upper layers, such as end-to-end reliability and rate control. To isolate the benefit of the new optimization, such techniques should be also incorporated in the traditional routing protocols. Similarly, comparing a reliable protocol against an unreliable one should be avoided; if the new protocol includes a mechanism for end-to-end reliability, a similar mechanism should be added to the old protocol.

**Separating rate control from end-to-end reliability.** When comparing a new reliable protocol to an unreliable one, the simplest method to add end-to-end reliability to the unreliable (traditional or not) routing protocol is to run it under TCP [19]. While this approach is simple, as no modification to the protocol itself is required, it may obscure the performance gain.

If the new protocol includes only reliability but no online congestion control (e.g., as is the case with FEC-style reliability), it is overkill to run the old protocol under TCP which includes both mechanisms which interact with each other. In this case, the throughput gap between the new and the old protocols may appear larger as a result of poor performance of TCP congestion control.

If the new protocol includes both reliability and online rate control (e.g., as is the case with ARQ-style reliability), it can be compared against the old protocol under TCP as a base-case comparison. Even so, since it is known that TCP performs poorly in wireless environments, it may still be unclear what the real gain from the new "exotic" technique is.

We advocate that in both cases, one should attempt to incorporate the reliability/rate control features of the new protocol to the old protocol, following the methodology of [19]. In this case, the comparison will be able to isolate the gain from the "exotic" technique exploited in the new protocol. We acknowledge this is not always easy to do. In some cases the reliability and congestion control mechanisms are disjoint components of the new protocol, not related to the new "exotic" technique used (e.g., in noCoCo). In this case reliability is typically provided in the traditional way (through retransmissions). This disjoint mechanism should be also incorporated to the old protocol used for comparison. In other cases, the reliability component of the new protocol may be part of the "exotic" technique itself (e.g., in MORE), and not a disjoint ARQ component. In such cases, the reliability component should be carefully added to the old protocol, for example, by adding FEC, and not by running it under TCP, so that the comparison is not affected by the negative effects of TCP's rate control mechanism.

**How to incorporate rate control to traditional routing?** Similar arguments against TCP apply here. If two unreliable protocols are compared, one with a rate control component and one without, running the second protocol under TCP is not a good solution, because the reliability mechanism is not required. What should be done is again incorporating the rate control mechanism of the new protocol to the old protocol. For example, in the evaluation of SOAR, the window-based rate control mechanism used in SOAR could be easily incorporated to traditional routing; in that case the comparison would isolate the gain of opportunistic forwarding.

**MAC autorate adaptation.** We argue that a good practice is for new "exotic" protocols to make an attempt to incorporate autorate adaptation. We acknowledge this is not an easy task and perhaps it is not always feasible. Even in those cases, we argue autorate adaptation should always be enabled for the case of traditional routing; an "exotic" protocol should be shown to outperform traditional routing both with and without autorate adaptation.

## 4. SUMMARY

In summary, we postulate that a fundamental reason for the complexity of evaluating high-throughput WMN routing protocols is that the research community still does not have a unified framework for understanding the interactions of MAC layer, congestion, interference, network coding, and reliability. WMN routing schemes are still being proposed as point solutions in a space of options; the real problem goes beyond how to evaluate them, but rather lies in how to understand the fundamental roles of their constituent parts.

## 5. REFERENCES


[1] S. Biswas and R. Morris. ExOR: Opportunistic multi-hop routing for wireless networks. In *Proc of ACM SIGCOMM*, 2005.

[2] 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. of ACM MobiCom*, 1998.

[3] S. Chachulski, M. Jennings, S. Katti, and D. Katabi. Trading structure for randomness in wireless opportunistic routing. In *ACM SIGCOMM*, 2007.

[4] D. S. J. D. Couto, D. Aguayo, J. C. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *Proc. of ACM MobiCom*, 2003.

[5] S. R. Das, C. E. Perkins, and E. M. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *Proc. of IEEE INFOCOM*, 2000.

[6] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *Proc. of ACM MobiCom*, 2004.

[7] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla. The impact of multihop wireless channel on TCP throughput and loss. In *IEEE Infocom*, 2003.

[8] M. Ghaderi, D. Towsley, and J. Kurose. Reliability Gain of Network Coding in Lossy Wireless Networks . In *Proc. of IEEE INFOCOM*, 2008.

[9] C. Gkantsidis, W. Hu, P. Key, B. Radunovic, S. Gheorghiu, and P. Rodriguez. Multipath code casting for wireless mesh networks. In *Proc. of CoNEXT*, 2007.

[10] A. Gupta, Wormsbecker, and C. Williamson. Experimental evaluation of TCP performance in multi-hop wireless ad hoc networks. In *Proc. of MASCOTS*, 2004.

[11] D. B. Johnson and D. A. Maltz. *Dynamic Source Routing in Ad Hoc Wireless Networks*. Kluwer Academic, 1996.

[12] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft. XORs in the air: Practical wireless network coding. In *Proc. of ACM SIGCOMM*, 2006.

[13] V. Kawadia and P. Kumar. Experimental investigations into TCP performance over wireless multihop networks. In *SIGCOMM E-WIND Workshop*, 2005.

[14] Y. Li, L. Qiu, Y. Zhang, R. Mahajan, Z. Zhong, G. Deshpande, and E. Rozner. Effects of interference on throughput of wireless mesh networks: Pathologies and a preliminary solution. In *Proc. of HotNets-VI*, 2007.

[15] X. Lin and N. Shroff. The impact of imperfect scheduling on cross-layer rate control in multihop wireless networks. In *Proc. of IEEE INFOCOM*, 2005.

[16] D. Lun, M. Medard, and R. Koetter. Efficient operation of wireless packet networks using network coding. In *Proc. of IWCT*, 2005.

[17] C. E. Perkins and E. M. Royer. Ad hoc on-demand distance vector routing. In *Proc. of IEEE WMCSA*, 1999.

[18] E. Rozner, J. Seshadri, Y. Mehta, and L. Qiu. Simple opportunistic routing protocol for wireless mesh networks. In *Proc. of IEEE WiMesh*, 2006.

[19] B. Scheuermann, W. Hu, and J. Crowcroft. Near-optimal coordinated coding in wireless multihop networks. In *Proc. of CoNEXT*, 2007.

[20] Y. Yuan, H. Yang, S. H. Wong, S. Lu, and W. Arbaugh. ROMER: Resilient opportunistic mesh routing for wireless mesh networks. In *Proc. of IEEE WiMesh*, 2005.