MobiCom Poster Abstract: Interference-aware Fast Path Adaptation in Wireless Mesh Networks

Vishnu Navda **Samir Das**
{*vnavda, samir*}*@cs.sunysb.edu.edu*
Stony Brook University, NY

Samrat Ganguly
samrat@nec-labs.com
NEC Labs America, NJ

I. Introduction

Existing multihop wireless routing protocols are incapable of reacting quickly to transient degradation of link quality rendering them unsuitable for supporting real-time applications such as streaming/interactive voice and video. This limitation in the sensitivity of the routing protocols to link quality is difficult to overcome due to the following reasons. All link quality metrics used in routing, such as ETX [1, 2] require active probing. Too frequent probing results in high overhead and can interfere with the actual traffic. Further, a time window of 5 to 10 seconds is required to gather enough probe samples to allow reasonable confidence in the statistics and avoid load sensitivity. Additional latency is also incurred in propagating routing updates.

We designed Deflect, a lightweight and opportunistic mechanism that works underneath the routing layer. Deflect enables fast local adaptation of end-to-end routes in response to sudden drops in link delivery ratio or node failures. The design of Deflect achieves the following goals: a) zero message overhead mechanism for monitoring link quality at very short time scale; b) fast switching of traffic through a node in neighborhood of the poor link; c) transparency to end-to-end routing protocol. We show that Deflect can react to link quality degradation within a few hundred milliseconds, compared to tens of seconds for an end-to-end routing protocol.

II. Design of Deflect

The basic idea in Deflect is that a node can volunteer to replace a neighboring relay node on a routing path if it determines – via passive measurements based on snooping – that overall delivery rate will improve by this action. Consider Figure 1 for an illustration. Assume that $A-B-C-D-E$ is a route computed by the routing protocol based on long term observation of the ETX metric [1] of each link of the network. Note that alternative routes, such as $A-X-C-Y-E$, are possible, let us assume that they have a higher cost as determined by the metric. Deflect provides a low overhead mechanism for fast switching to the alternate route depending upon the condition of links AB and CD. Reconstruction of an overall better end-to-end path is still the responsibility of the routing protocol.

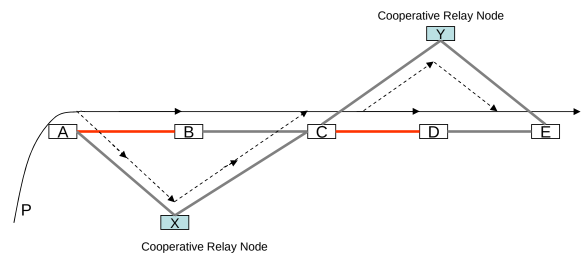

Figure 1: Path adaptation overview.

In Deflect, a node such as X is called a *cooperative relay* for a node such as B that is the original relay node. X monitors the performance of the AB and BC links (in terms of their delivery ratios) by overhearing the packets transmitted from A and from B. Now, if X determines, based on this estimation, that the product of the delivery ratios of AB and BC is worse than that of AX and XC, it signals node A so that A temporarily flips the next hop route through X. B then starts playing the role of X. Similar relay switching can happen at any hop in an end-to-end path, for example, again at Y. Clearly, the scheme depends on existence of such cooperative relay nodes X and Y. In a dense, large-scale deployment network such cooperative relays are expected to be quite common. Even in our small 20 node mesh testbed, we found several nodes that can act as cooperative relays between end-to-end paths. The scheme also depends on the accuracy of estimation of the link qualities for AB, BC, AX and XC.

Deflect keeps track of 802.11 MAC layer retransmissions in order to estimate link qualities.

Estimating link AB: The delivery ratio of link AB can be estimated by simply observing on node X all frames on link AB and computing the fraction of the frames that have unique sequence numbers.

Estimating link AX: The total number of unique frames transmitted by A can be estimated by computing the sum of number of unique frames heard at X

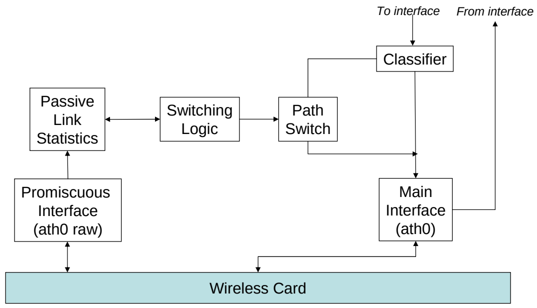

Figure 2: Block diagram describing operation of Deflect beneath the routing/packet forwarding layer.

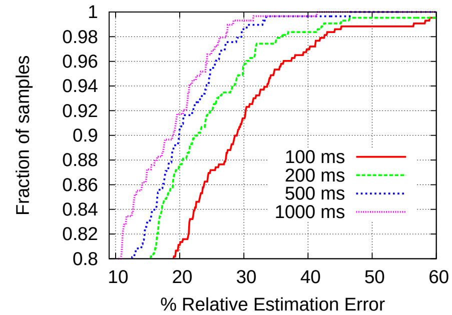

Figure 3: Relative estimation error (%) with time-varying interference pattern.

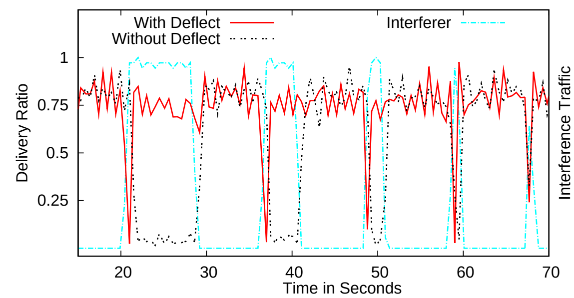

Figure 4: Benefit of Deflect in presence of time varying interference.

and the number of "holes" in the sequence numbers among these packets. The delivery ratio of link AX is estimated as the ratio of unique frames heard to the total number of unique frames transmitted.

Estimating link BC: This procedure is similar to link AB. Note that here we exploit the ability of X to overhear B via link BX.

Estimating link XC: The delivery ratio from C to X in one direction is used as an estimator for link XC. The technique to estimate CX changes depending on whether C transmits any data packets. If it does, then the procedure is similar to link AX. Otherwise, the procedure involves estimating what fraction of ACK frame transmissions from C to B are heard by X. If X hears the data frame on link BC, but misses the corresponding ACK frame and does not hear a following data frame retransmission, X concludes that it has lost the ACK frame.

All potential cooperative relays passively monitor the link qualities continuously on the path(s) it can help bypass. If the estimated delivery ratio on the alternate path exceeds that in the primary path by threshold T in a time window, the cooperative relay transmits a *signaling message* to the source of the two-hop path (i.e., node A) notifying the availability of a better path. The source node at this point switches the next hop to the cooperative relay (X). After a path is switched, the role of the original relay node (e.g., B) and the cooperative relay (e.g., X) is reversed. This happens automatically, and not through explicit messaging. Thus, the path can be switched back to the original later when the link qualities change. The entire Deflect mechanism is independent of routing, and operates in a *sub-layer* below IP packet forwarding, but above the link layer.

III. Evaluation

Deflect is implemented using *Click* modular router on Linux as shown in Figure 2. We evaluated Deflect on a 20 node indoor mesh network testbed. In the setup, particular nodes create interference on target links by generating traffic *bursts* – with alternating on and off periods. The burst interval is varied to study the sensitivity of Deflect with different interference durations.

Sensitivity to time varying interference: Figure 3 shows the dependency of relative estimation error for different on (burst) periods. We note that the mean error ranges from 6% to 12%. From figure 3, we observe that around 80% of the samples have less than 15% error, when the on period is 200ms or more. Figure 4 shows a trace of Deflect in action when the interferer turns on or off. The interference bursts are of a shorter duration than the ETX evaluation period (10sec). In absence of Deflect, the packet delivery ratio on the 2-hop path suffers greatly when the interferer is on. With Deflect, the delivery ratio is quite stable due to path switching.

References

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

[2] Richard Draves, Jitendra Padhye, and Brian Zill. Comparison of routing metrics for static multi-hop wireless networks. *SIGCOMM Comput. Commun. Rev.*, 34(4):133–144, 2004.