Enhanced Loss Differentiation Algorithms for Use in TCP Sources over Heterogeneous Wireless Networks

Stefano Bregni, *Senior Member, IEEE*, Davide Caratti, Fabio Martignon

Politecnico di Milano, Dept. of Electronics and Information, Piazza L. Da Vinci 32, 20133 Milano, ITALY
Tel.: +39-02-2399.3503 – Fax: +39-02-2399.3413 – E-mail: {bregni, caratti, martignon}@elet.polimi.it

Abstract — **Loss Differentiation Algorithms (LDA) are used to provide TCP with an estimate of the cause of packet losses, to improve performance over heterogeneous networks including wired and wireless links. In this work, we compared by simulation the accuracy of several LDA schemes in various realistic scenarios. We experienced that LDA schemes originally proposed in literature exhibit poor performance in estimating the cause of packet losses. Thus, we propose enhancements to the Non Congestion Packet Loss Detection (NCPLD) and Vegas schemes, achieving higher accuracy in all network scenarios. We shown that our proposed enhanced schemes approach reasonably ideal accuracy of LDA having perfect knowledge of the cause of packet losses. These results entail the possibility to adopt such algorithms within TCP congestion control, to achieve higher performance over heterogeneous wireless networks.**

***Index Terms* — Internet, mobile computing, protocols, TCP, wireless networks.**

I. Introduction

The Transmission Control Protocol (TCP) is a popular Internet protocol for reliable data delivery, which adapts very well to disparate environments. However, it was designed optimising its performance to cope with packet loss due to network congestion: thus, it performs poorly in lossy environments.

The assumption that packet loss is an indicator of network congestion may not apply to heterogeneous networks, such as those including wireless links, in which packet loss may be induced also by noise or any other reason than congestion [1]. There, random loss due to bit corruption is misinterpreted: upon loss detection, the TCP sender reduces its transmission rate unnecessarily. To overcome this problem, TCP sources may estimate the cause of packet losses by guessing whether the current network state is congested or not congested [2].

To obtain this estimation, some Loss Differentiation Algorithms (LDA) have been recently proposed [3][4]. These algorithms estimate the cause of packet losses based on TCP state variables and information from acknowledgement packets (ACKs). Upon detection of a packet loss, the TCP sender bases on this estimate to decide the appropriate counteraction.

The key feature for LDA schemes is to be accurate in ascribing the cause of packet losses, as TCP error recovery reacts gently or aggressively depending on the LDA decision. To summarize, when packet error rate is low and most of the packet losses are due to congestion, LDA accuracy in ascribing losses is necessary to achieve fairness with concurrent TCP flows. On the other hand, when packet error rate is high such as in wireless links, LDA accuracy is necessary in order to achieve throughput gain.

In this work, we evaluated by simulation the accuracy of several LDA schemes proposed in literature (Flip Flop [3], Vegas [5], Non Congestion Packet Loss Detection, NCPLD [4]) in various realistic scenarios, comprising both wired and wireless links, following a similar approach as proposed in [5]. To assess the accuracy of these schemes, we compared their performance with an upper bound, obtained by ideally assuming perfect knowledge of the cause of packet losses, as well as with a lower bound, obtained by taking random or constant decisions about the nature of packet losses.

Simulations were carried out using the Network Simulator package (*ns* v.2 [6]), embedding various sender-side LDA schemes in a TCP NewReno source. First, we found that some of the LDA schemes originally proposed in literature are not very accurate neither effective when embedded in TCP sources. Moreover, we noticed that the accuracy of some LDA schemes could be improved by tuning their parameters. Then, we evaluated the performance of the so enhanced LDA algorithms, finding that they can be far more accurate than a random-decision LDA. These results entail the possibility to adopt such algorithms within TCP congestion control to achieve higher performance in heterogeneous networks.

This paper is organized as follows. Section II describes the LDA schemes examined. Section III presents the network scenarios used to assess their performance. Section IV defines the metric used to assess the accuracy of the LDA schemes. Section V studies the accuracy of the LDA schemes through numerical results. Finally, Section VI concludes the paper.

II. Loss Differentiation Algorithms

The loss differentiation decision can be obtained based on TCP state variables, namely congestion window (*cwnd*), slow start threshold (*ssthresh*) and Round Trip Time (*RTT*). Moreover, LDA can use additional information, such as rate estimations: recent proposals of TCP enhancements are based on techniques to estimate the actual TCP transmission rate [7]. In conclusion, a generic LDA is a Boolean function of TCP state variables and sender-side network measurements, returning the network state on packet loss detection: either congested or not congested.

A first class of LDAs estimates the cause of packet losses based on rate estimates. In particular, the *Vegas predictor* proposed in [5] bases its decision on the function

$$diff_V = \left(\frac{cwnd}{RTT_{\min}} - \frac{cwnd}{RTT} \right) \cdot RTT_{\min} \quad (1)$$

where *cwnd* is the congestion window, $RTT_{\min}$ is the minimum *RTT* sample measured during the TCP session, *cwnd*/*RTT* is an estimate of the actual transmission rate of the TCP source, $cwnd/RTT_{\min}$ is an estimate of the expected transmission rate. Given the two parameters α and β [segments], when $diff_V \geq \beta$, TCP Vegas assumes that the network is congested; when $diff_V \leq \alpha$, possible losses are ascribed to transmission random errors. Finally, when $\alpha < diff_V < \beta$, the predictor assumes that the network state is the same as in the previous estimation.

The Vegas predictor with α=1, β=1, studied in [5], proved not accurate. On the other hand, by simulations on a wide set of values (α, β), we found that accuracy can be substantially improved by tuning the parameters.

A second class of LDAs uses delay measures to estimate the congestion status. Higher *RTT* values are supposed to be the effect of increased queuing delay over the network.

The *Non Congestion Packet Loss Detection* (NCPLD) scheme [4] estimates the cause of packet losses by detecting the knee-point in the load-throughput curve of the network [8]. The TCP sender estimates the total number of segments in flight over the path to the receiver (*TotalPipeSize*) as

$$TotalPipeSize = \frac{1}{2} \cdot RTT_k \cdot \frac{W_k - W_{k-1}}{RTT_k - RTT_{k-1}} \quad (2)$$

where W_k and RTT_k are respectively the flight-size and the round trip time measured on reception of the k^{th} ACK. The NCPLD scheme needs also an estimate of the bandwidth-delay product [4]. To this aim, we used an exponentially-weighted moving average (EWMA) filtering the ACK reception rate to get an estimate of the transmission rate B_{TX}. Both estimates yield the current value of RTT at the knee-point, as

$$RTT_{kp} = RTT_{\min} + \frac{1}{2} RTT \cdot \frac{B_{TX} \cdot RTT_{\min}}{TotalPipeSize} \quad (3).$$

The NCPLD scheme ascribes packet loss to network congestion if the current *RTT* sample is greater than RTT_{kp}, to a transmission error otherwise.

However, early simulations showed that the original NCPLD scheme [4] does not exhibit high accuracy, mainly due to the coarseness with which the TCP source measures the round trip time. To solve this problem, we propose a modified version of the NCPLD scheme that approximates the difference between two consecutive *RTT* samples to the value of TCP clock granularity [9], when they assume the same value. This allows the NCPLD scheme to provide correct estimations of *TotalPipeSize* even when consecutive samples of round trip time have the same value.

The *Flip Flop* scheme, proposed in [3], uses a Flip Flop Filter on the last measured *RTT* samples to detect congestion events: a vector with length *L* stores the type of the last *L* used filters (stable or agile) [7]. If the number of stable filters exceeds a fixed threshold η, packet losses are ascribed to network congestion, to transmission error otherwise.

Finally, to assess the accuracy of these three schemes by establishing upper and lower bounds, we also implemented the following schemes:

- *Ideal-LDA*. It knows the exact cause of the last packet loss, yielding an upper bound to the accuracy.
- *Constant-LDA*. It produces a constant estimated state, no matter what the input conditions are. Therefore, two constant LDA schemes can be devised: the *Always Congested* and the *Always Wireless* scheme, the former assuming all losses are due to buffer overflow, the latter ascribing all losses to transmission errors over the wireless path.
- *Random-LDA*. It produces an estimate of the network as a random result. As there is no *a priori* information about the distribution of congestion and wireless events, we set equiprobable decisions on output. This scheme yields a lower bound to the accuracy.

III. Simulation Models

Simulations were carried out using the Network Simulator package (*ns* v.2 [6]). The accuracy of the LDA schemes outlined in the previous section has been evaluated in several network scenarios.

Throughout this paper, the Maximum Segment Size of TCP sources is equal to 1500 bytes and all queues can store a number of packets equal to the bandwidth-delay product. TCP receivers always implement the Delayed ACKs algorithm.

The simplest network topology, named *Single Link*, is depicted in Fig. 1: a single TCP-NewReno source S, provided with LDA estimator, performs a bulk FTP transfer. The wired link, N1-N2, has capacity *C*=10 Mbit/s and propagation delay τ=50 ms. The parameters of the wireless link N2-D are *C*=10 Mbit/s and τ=0.01 ms.

In our simulations, the wireless link may drop packets according to two different error models. In one case, packets are dropped independently without time correlation, with Packet Error Rate (PER) ranging from 10^{-5} to 10^{-1}. In the other case, the link N2-D was modelled using a two-state Markov chain [10], where the channel is either in *Good* or *Bad* state. In the *Good* state, the link does not drop any packet. In the *Bad* state, PER is varied in the 0 to 80% range, to simulate different fading conditions. The time spent in each state is exponentially distributed, with average equal to 1 s for the *Good* state and 50 ms for the *Bad* state, again according to [10].

In the topology named *Congested Dumbbell,* shown in Fig. 2, we simulate a simple hybrid environment: a single TCP source provided with LDA shares the bottleneck link N1-N2, with capacity *C*=10 Mbit/s and delay τ=50 ms, with 30 UDP sources having the same priority as the TCP source. Each UDP source switches between ON and OFF states, with Pareto-distributed periods having shape parameter equal to 1.5 and mean set to 100 ms and 200 ms, respectively. In the ON state, each UDP source send packets with size 1500 byte at constant rate 0.5 Mbit/s. In the OFF state, UDP sources do not send any packet.

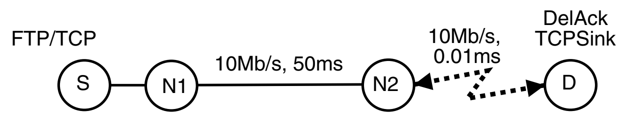

Fig. 1: S*ingle Link* network topology. The dashed line represents the wireless link affected by random transmission errors.

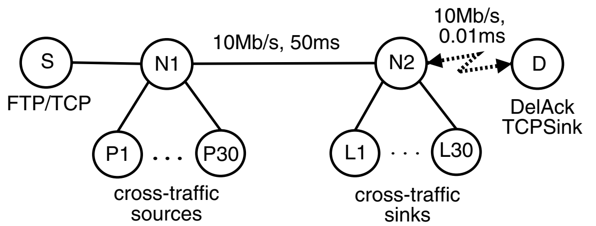

Fig. 2: C*ongested Dumbbell* network topology. The dashed line represents the wireless link affected by random transmission errors.

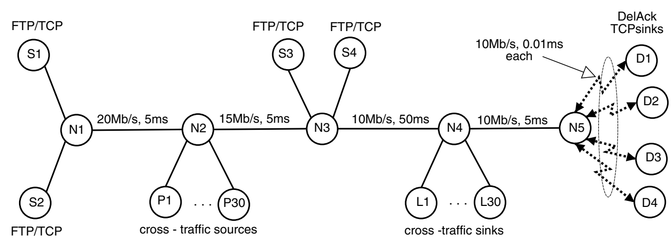

Fig. 3: *Multihop* network topology. The dashed lines represent wireless links affected by random transmission errors.

Such cross-traffic configuration leaves to the TCP source an available bandwidth that varies randomly during the simulation, with average equal to half the bottleneck capacity. The wireless link N2-D is affected by random losses according to the models described in the Single Link topology.

Finally, in the *Multihop* topology shown in Fig. 3, two couples of TCP sources are connected on different entry nodes. They share the bottleneck N3-N4 with 30 UDP sources, as in the previous case, but with ON rate 0.2 Mbit/s. The last hop of each TCP connection is affected by random transmission errors as in the Single Link case.

IV. Performance Metric

Let *LE* be the number of loss events detected by the TCP source based on the reception of triple duplicate acknowledgements or retransmission timeout expirations. We define *wireless loss (WL)* a packet loss caused by the wireless noisy channel. Instead, a *congestion loss (CL)* is defined as a packet loss caused by network congestion.

Let CL_D and WL_D be, respectively, the number of congestion and wireless losses correctly classified by the LDA scheme under investigation. The accuracy of loss classification A_L can be defined as the ratio between the number of correct classifications and the total number of loss events, i.e.

$$A_L = \frac{CL_D + WL_D}{LE} \quad (4).$$

V. Simulation Results

By extensive simulations, we measured the accuracy of the LDA schemes described previously in a variety of realistic scenarios. Multiple long-lived file transfers were simulated, achieving very narrow 97.5% confidence intervals [11].

First, the accuracy of the *Always-Congested* and the *Always-Wireless* schemes was measured in all network scenarios. Fig. 4 shows the accuracy A_L for the Always-Congested scheme as the PER varies in the 0.001% to 1% range in all the three topologies described in Section III. The accuracy of the Always-Wireless scheme is always equal to 1-A_L, where A_L is the accuracy of the Always-Congested scheme in Fig. 4.

The Always-Congested scheme provides a measure of the frequency of congestion events in the three topologies as a function of PER. As shown in Fig. 4, loss events are more frequent in the Congested-Dumbbell and less frequent in the Single-Link topology, while the Multihop yields intermediate results. Fig. 4 also shows the accuracy achieved by the Ideal LDA, always equal to 1, and by the Random LDA, equal to 0.5 in the whole range of PER and in all simulated scenarios.

We then examined the accuracy of real LDA schemes proposed in literature. We considered the *Flip Flop LDA scheme*, configuring it with history length L=8 (as suggested in [3]) and with threshold η varying in the 2 to 6 range. Fig. 5 shows the accuracy achieved by the Flip Flop LDA in the Multihop topology. The accuracy achieved by the Always Congested and the Always Wireless schemes is also reported to provide a comparison. Lower values of the threshold η cause Flip Flop to be less accurate in classifying wireless losses, in accordance with the observation that the accuracy on wireless losses is higher as the threshold η approaches the history length L [3].

The accuracy of the Flip Flop LDA was measured also in the Congested-Dumbbell and in the Single-Link topology, in order to evaluate the performance of the Flip Flop scheme with different traffic conditions. We found that the best tradeoff for the accuracy on packet loss classification is achieved when the threshold η is set to 6, according with the measurements reported in [3].

We then tested the accuracy of the *Vegas Predictor,* by varying the parameters α and β. Fig. 6 shows the accuracy achieved in the Multihop topology for $\{\alpha, \beta\}$ equal to {1.5, 2.5}, {2, 2}, {1, 3} and {1,1} (original setting studied in [5]). We found that accuracy is substantially improved by choosing the interval $[\alpha, \beta]$ centered on 2 segments. With this choice, Vegas exhibits low sensitivity to parameter setting; however, it achieves the highest accuracy for α=1 and β=3.

Finally, we compared the best configurations of the evaluated LDA schemes to determine which algorithm provides the best accuracy.

First, Fig. 7 compares the accuracy of loss classifications as a function of PER for all studied LDAs in the *Single Link topology*. The Vegas and NCPLD schemes are more accurate than Flip Flop for all values of PER. The NCPLD scheme is more accurate than Vegas in predicting congestion losses, i.e.

for lower packet error rates. On the other hand, Vegas is more accurate than NCPLD in detecting wireless losses.

The performance of the LDA schemes improves when congestion is the prevalent cause of packet losses. Fig. 8 shows that, in the *Congested Dumbbell* topology, all the LDA schemes provide classification accuracy higher than 70%.

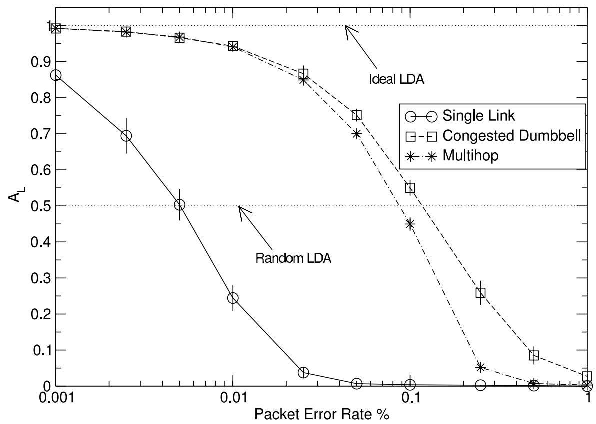

Fig. 4: Accuracy of the *Always Congested* scheme as a function of PER.

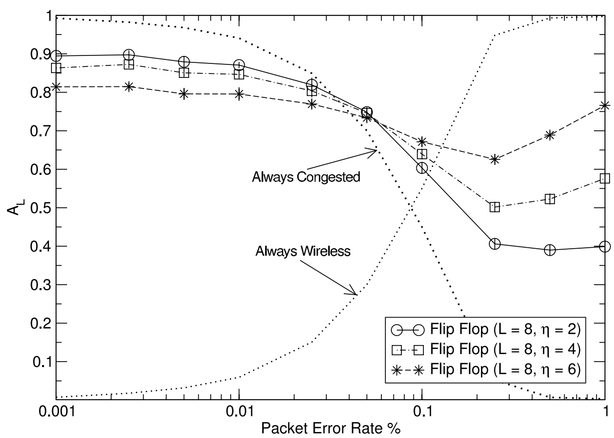

Fig. 5: Accuracy of the *Flip Flop* scheme as a function of PER for different values of η in the *Multihop* topology.

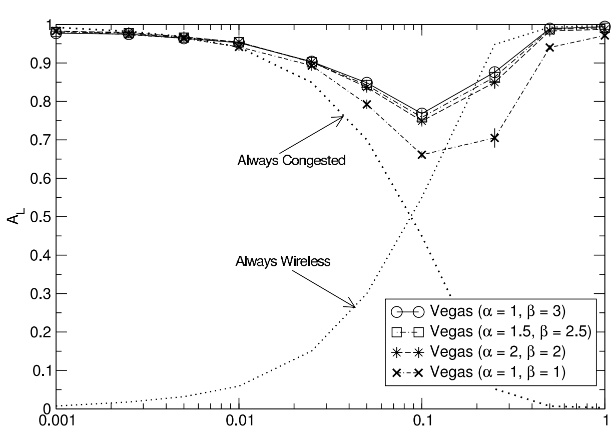

Fig. 6: Accuracy of the Vegas Scheme as a function of PER for different values of $\{\alpha, \beta\}$ in the *Multihop* Topology.

The best performance was obtained by NCPLD and Vegas {1, 3}, as found for the Single Link topology.

Finally, Fig. 9 and Fig. 10 compare the accuracy of LDA schemes in the *Multihop* topology.

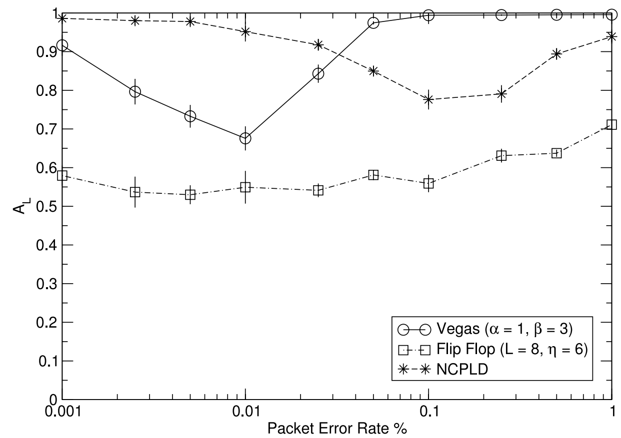

Fig. 7: Accuracy of packet loss classification as a function of PER for all LDA schemes in the S*ingle link* Topology.

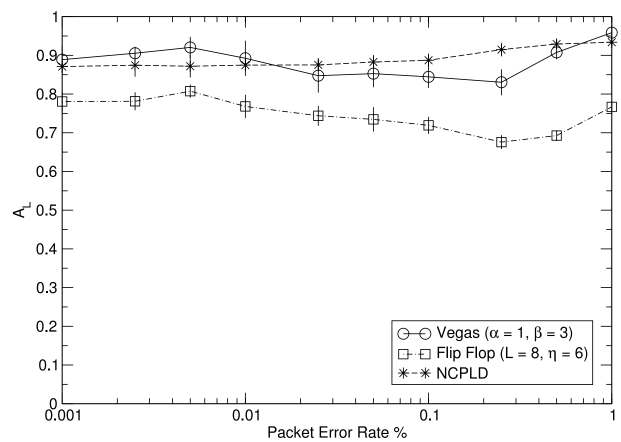

Fig. 8: Accuracy of packet loss classification as a function of PER for all LDA schemes in the *Congested Dumbbell* topology.

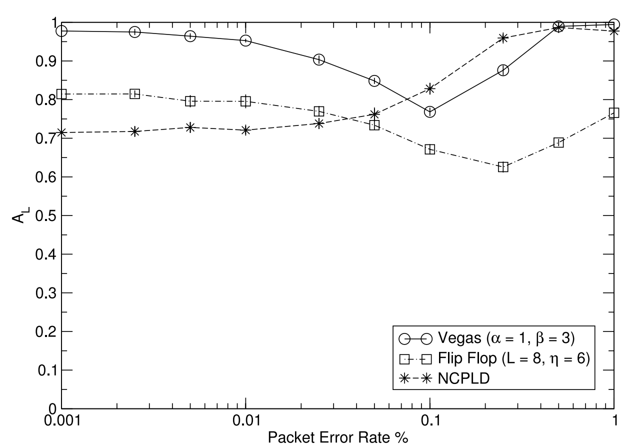

Fig. 9: Accuracy of packet loss classification as a function of PER for all LDA schemes in the *Multihop* topology.

In this more complex scenario, the NCPLD scheme is less accurate in detecting congestion losses, while the Flip Flop and Vegas schemes achieve approximately the same level of accuracy as in the Congested Dumbbell scenario. Moreover, the accuracy of packet loss classification for the NCPLD scheme decreases for PER>1%.

The main reason for this behaviour is the inability of the EWMA bandwidth estimator to measure correctly the bottleneck capacity when either the occurrence of wireless losses is high or the network topology becomes complex. We think that a future improvement of the NCPLD scheme could provide a more accurate bandwidth estimator in order to overcome this problem. On the contrary, the accuracy of the Flip Flop and Vegas schemes, also shown in Fig. 10, does not degrade as the PER increases. The accuracy of the Vegas predictor approaches 100% at high wireless loss rate in all our measurements.

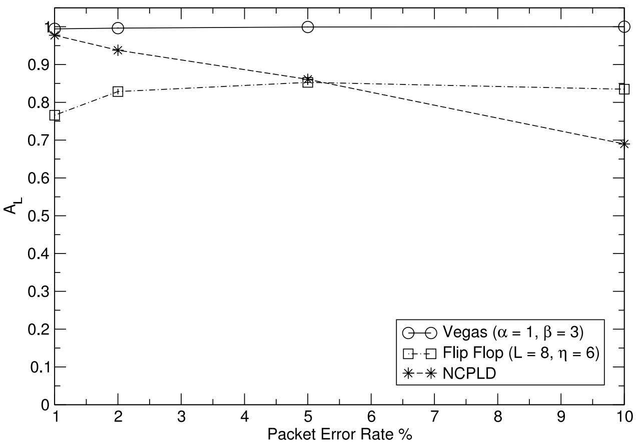

Fig. 10: Accuracy of packet loss classification for all LDA schemes in the *Multihop* topology with high PER values.

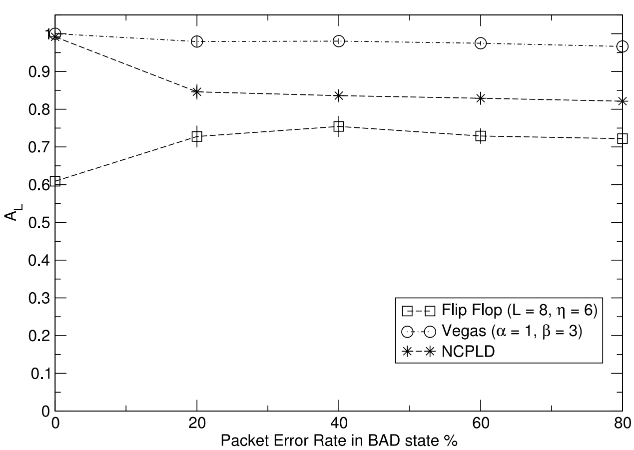

Fig. 11: Accuracy of packet loss classification for all LDA schemes as a function of PER in the *Bad* state in the *Single Link* topology.

Finally, Fig. 11 shows the accuracy of the LDA schemes in the Single Link topology when transmission errors are correlated and modelled according to the Markov chain described in Section III. The PER in the *Bad* state varies from 0% to 80%. The Vegas predictor provides higher accuracy than the other LDA schemes and approaches the ideal estimator for the whole range of packet error rates. Also the two other predictors achieve a good level of accuracy of classifications. Similar results, not reported for the sake of brevity, were observed in the other topologies. Note that the modified Vegas predictor proposed in this paper always achieves an accuracy A_L>70%, thus providing much higher accuracy than a random coin tossing scheme.

VI. CONCLUSIONS

In this paper, we compared by simulation the accuracy of several LDA schemes in various realistic scenarios.

In early simulations, we experienced that some LDA schemes originally proposed in literature exhibit poor performance in estimating the cause of packet losses. Then, we proposed enhancements to the NCPLD and Vegas schemes, which achieved higher accuracy in all network scenarios. In our experiments, the accuracy achieved by the enhanced Vegas LDA was always much higher than that obtained by taking random or constant decisions. It also approached the upper bound of an ideal LDA that assumes perfect knowledge of the cause of packet losses. Therefore, our results entail the possibility to adopt such algorithms within TCP congestion control to achieve higher performance over heterogeneous wireless networks.

REFERENCES

[1] V. Tsaoussidis, I. Matta, "Open Issues on TCP for Mobile Computing", *Journal of Wireless Communications and Mobile Computing*, Wiley Academic Publishers, Issue 2, Vol.2, February 2002.
[2] T.E. Kim, S. Lu, and V. Bharghavan, "Improving Congestion Control Performance Through Loss Differentiation", *Proc. of 8th Int. Conf. on Computer Communications and Networks*, Boston, MA, USA, 1999.
[3] D. Barman and I. Matta, "Effectiveness of Loss Labeling in Improving TCP Performance in Wired/Wireless Networks", Boston University Technical Report, 2002.
[4] N.K.G. Samaraweera, "Non-Congestion Packet Loss Detection for TCP Error Recovery using Wireless Links", *IEE Proceedings Communications*, volume 146 (4), pages 222-230, August 1999.
[5] S. Biaz and N. H. Vaidya, "Distinguishing Congestion Losses from Wireless Transmission Losses: a Negative Result", *Proc. of IEEE 7th Int. Conf. on Computer Communications and Networks*, New Orleans, LA, USA, October 1998.
[6] VINT project U.C. Berkeley/LBNL, *"ns-2 network simulator (ver. 2)"*. Available at URL: http://www.isi.edu/nsnam/ns.
[7] M. Kim, B. Noble, "Mobile Network Estimation", *Proc. of the 7th ACM SIGMOBILE*, Roma, Italy, July 2001.
[8] D.Bertsekas, R.Gallager, *"Data Networks"*, Prentice Hall, 1987.
[9] V. Paxson and M. Allman, "Computing TCP's Retransmission Timer", RFC2988, November 2000.
[10] P. Bhagwat, P. Bhattacharya, A. Khrishna, S. Tripathi, "Using Channel State Dependent Packet Scheduling To Improve TCP Throughput over Wireless LANs", *Wireless Networks*, vol. 3(1): 91-102, March 1997.
[11] K. Pawlikowski, H.-D. Joshua Jeong, J.-S. Ruth Lee, "On Credibility of Simulation Studies of Telecommunications Networks", *IEEE Communications Magazine*, pp. 132-139, January 2002.