Second-Order Rate-Control Based Transport Protocols Over Mobile Wireless Networks

Xi Zhang
Networking and Information Systems Labs.
ECE Dept., Texas A&M University
College Station, TX, 77843, USA
Email: *xizhang@ece.tamu.edu*

Hsiao-Hwa Chen
Institute of Communications Engin.
National Sun Yat-Sen University
70 Lien-Hai Road, Kaohsiung, Taiwan
E-mail: *hshwchen@ieee.org*

Mohsen Guizani
Department of Computer Science
Western Michigan University
Kalamazoo, MI 49008, USA
Email: *mguizani@cs.wmich.edu*

Abstract— While TCP (Transmission Control Protocol) is an efficient transport protocol in the wired Internet, it performs poorly when used in wireless environments. This is because TCP *couples* the error and flow control by using packet loss to *infer* the network congestion and thus the random loss in wireless Internet can inevitably mislead TCP dropping its flow-control window unnecessarily, even if the network is not congested at all. To overcome this problem, we propose the second-order rate-based flow control and the *decoupled* window-based error-control schemes for high-throughput transport protocols over the wireless networks. The second-order rate control minimizes congestive losses by using the Explicit Congestion Notification (ECN)-bit feedback to adapt the rate-gain parameter to the variations of the round-trip time (RTT) and cross-traffic flows. The error-control scheme detects and selectively retransmits the lost packets caused by either congestion or random-noise/handoffs on wireless links, which is decoupled from the flow control such that the rate control is independent of the random loss of wireless links. Using the fluid analysis, we establish the rate-control model, and derive expressions for throughput, losses, and link-transmission efficiency. Through extensive simulations, the proposed transport protocol is shown to possess the TCP-compatibility in bandwidth while coexisting with TCP-Reno traffics in the wired Internet. Our simulations also verify the analysis, and demonstrate the significant superiority of our scheme to TCP in terms of increasing the average throughput over wireless links and the robustness to the variation of wireless random-loss probability while minimizing the losses and retransmissions.

Index Terms— Wireless Internet, random loss, transport protocol, decoupled flow and error control, second-order rate control.

I. INTRODUCTION

THE demand for wireless Internet access in mobiles, offices, and homes continues to grow. Consequently, a high-throughput and reliable transport protocol is needed for lossless data transmission through the wireless Internet [1]. While TCP is an efficient high-throughout transport protocol in the networks composed of wired links and stationary hosts, it performs poorly when used in wireless environments. This is because TCP couples the error control with flow control by using packet loss to *infer* the network congestion, and thus the random loss over the wireless links can inevitably mislead TCP dropping its flow-control window unnecessarily, even if the network is not congested at all. A number of studies have been reported in the literature [2], [3], [4] in dealing with random-loss impact on TCP caused by wireless links. However, all the previous works share the same philosophy of complying with the TCP's basic flow and error control framework while adapting it to wireless networks with the inevitable modifications. Unfortunately, the coupling of flow and error control in TCP doesn't in principle fit the wireless networks, because by implicit feedback TCP cannot distinguish between the congestive loss and the random loss of wireless links, which significantly degrades the throughput of any TCP-modification based schemes. Thus, the TCP-modification based schemes can only alleviate, rather than completely avoid, the unnecessary throughput degradation caused by TCP's flow and error control schemes in mobile computing environments.

To fundamentally solve the throughput-degradation problem with the transport protocols as used in the wireless networks while upper-bounding maximum queue size, we propose an efficient flow and error control scheme for high-throughput transport protocols over the wireless networks by using the second-order rate control, called the α-*control*, and the *decoupled* sliding-window error control. The proposed protocol separates flow control from error control, such that the rate control is independent of the random loss caused by random-noise/handoffs on wireless links. Functionally, we apply the α-control to reduce the congestive loss while using the decoupled error-control window to detect and selectively retransmit lost packets, caused by either congestion or random-noise/handoffs of wireless links.

Unlike the TCP that uses an implicit feedback congestion signal, the α-control employs a feedback mechanism, similar to Explicit Congestion Notification (ECN) [5], [6], [7], [8] scheme in TCP/IP as recently proposed by IETF, to detect an incipient congestion. The ECN-like mechanism used in the α-control can inform sources of congestion quickly and unambiguously, instead of making the source wait for either a retransmission timeout (TCP-Tahoe [9]), or three duplicate

ACKs (TCP-Reno [10]), to infer the network congestion. Consequently, the ECN-based α-control can not only minimize packet losses and retransmissions caused by the TCP flow-/error-control scheme itself [11], [12], but also eliminate the unnecessary throughput degradation caused by TCP flow and error control schemes when used in mobile networkings, which significantly improves the average throughput of the transport protocol over wireless networks.

The paper is organized as follows. Section II proposes the decoupled flow- and error-control transport protocol. Using the fluid analysis, Section III establishes the rate-control and random loss models over both wired and wireless links. Section IV evaluates our proposed scheme over wireless networks. The paper concludes with Section V.

II. DECOUPLED FLOW AND ERROR CONTROL SCHEME

To overcome the aforementioned throughput degradation caused by wireless links, we propose a *decoupled* flow and error control scheme for the transport protocols over mobile wireless networks to achieve high throughput while ensuring lossless transmissions. Fig. 1 shows the transport protocol connection under the proposed scheme, from a stationary host to a mobile/wireless host receiver via the Internet, network gateway, and wireless base station.[1] Control packets are used to *periodically* convey both flow and error control information through the connection between the source host and the mobile receiver. The source sends a forward control packet periodically for every Δ time unit, and the mobile receiver replies with a feedback control packet. The inter-control packet interval is typically a fraction of RTT. Control packet's flow-control information (ECN) is set by the mobile receiver or IP routers when the control packet passes through in either direction, and error-control information (ACK/NACK) is updated by the mobile receiver before returning a feedback control packet to the source. Upon arrival of a feedback control packet at the source, the control information is split into two parts: 1) the flow-control information contained in ECN-bit for the rate controller, and 2) the error-control information contained in ACK(N) (i.e., no loss) or NACK($N, M, Recv\_BIT\_MAP$) (i.e., there are losses) for the error controller (see Fig. 1), corresponding to the separated flow-control and error-control schemes.

A. The Second-Order Rate Control Based Flow-Control

The flow control traditionally employs the AIMD (Additively Increase and Multiplicatively Decrease) algorithm, which only controls the source rate $R(t)$ (first-order rate control), but does not upper-bound the maximum queue length

Q_{max} [14]. This is because AIMD can only make $R(t)$ fluctuate around the target bandwidth, but cannot adjust the rate-fluctuation amplitude that determines Q_{max}. So, the AIMD or the first-order rate control only applies the control over bandwidth while leaving bottleneck buffers un-controlled. In [14] we analytically showed that Q_{max} increases with both the rate-gain parameter $\alpha = dR(t)/dt$, and the connection's RTT (τ), and thus we proposed the *second-order rate control* or α-*control* [14], which can effectively deal with RTT variations.

In this paper, we propose to use the α-control [14] to handle the variations in the superposition of rate-gain parameters (α's) of the traffic flows sharing the same bottleneck and their RTTs. The α-control is a queue control mechanism at the bottleneck buffer, converging Q_{max} to the target buffer occupancy Q_{goal} (setpoint) as the number of the cross-traffic flows and the RTTs vary. If the number of flows sharing the bottleneck or RTT increases, Q_{max} increases. When $Q_{max} > Q_{goal}$, the buffer intends to overflow, implying that the current value of the superposed rate-gain α is too large, and thus all the connections sharing the bottleneck must reduce their α's. On the other hand, if $Q_{max} < Q_{goal}$, i.e., only a small portion of buffer is utilized, it indicates that the current α is too small for the reduced number of sharing flows or their RTTs, and thus each source should increase its α for improving buffer utilization and the responsiveness to the emerging bandwidth. Corresponding to network bandwidth and buffer resources, the α-control based scheme distinguishes the following two types of congestion:

Bandwidth Congestion: If the IP router's queue length $Q(t) > Q_t$, where Q_t is the predetermined threshold, then the router sets the local CN (Congestion Notification) bit to 1.

Buffer Congestion: If the IP router's maximum queue length $Q_{max} > Q_{goal}$, where $2Q_t < Q_{goal} < C_{max}$ (buffer capacity), then the router's (Buffer Congestion Notification) $BCN := 1$.

The proposed α-control based buffer-congestion control is exercised only when the source rate control is in a "decrease-to-increase" transition based on the feedback $BCN(n-1, n)$ at n-th rate control cycle, and can be described by:

$$\alpha_{n+1} = \begin{cases} \alpha_n + p; & \text{if } BCN(n-1, n) = (0, 0), \\ q\alpha_n; & \text{if } BCN(n-1, n) = (x, 1), \\ \alpha_n/q; & \text{if } BCN(n-1, n) = (1, 0), \end{cases} \quad (1)$$

where $p > 0$ and $1 > q > 0$ are α-control increase-step size and decrease factor, respectively; the resultant $\alpha_n > 0$, $\forall n = 1, \cdots, \infty$; and $x \in \{0, 1\}$.

B. The Sliding Window Based Error-Control Mechanism

The proposed scheme uses both NACK error detection and selective-retransmission recovery, see Fig. 1. Combining with selective retransmission, an NACK contains a range of the sequence numbers of packets that were lost and will be selectively retransmitted. The combination of NACK and

---

[1]We focus mainly on the transport protocol connection from a stationary host to a wireless mobile terminal at the last link hop — the typical and widely targeted/addressed "last mile problem" model [4], [13] in the wireless Internet environments, but the proposed scheme is also applicable to the case of the mobile-host to mobile-host transport protocol connection, and the other cases where the wireless link (links) is (are) not at the last hop in a transport protocol connection (see the example described in Fig. 4(a) in Section IV).
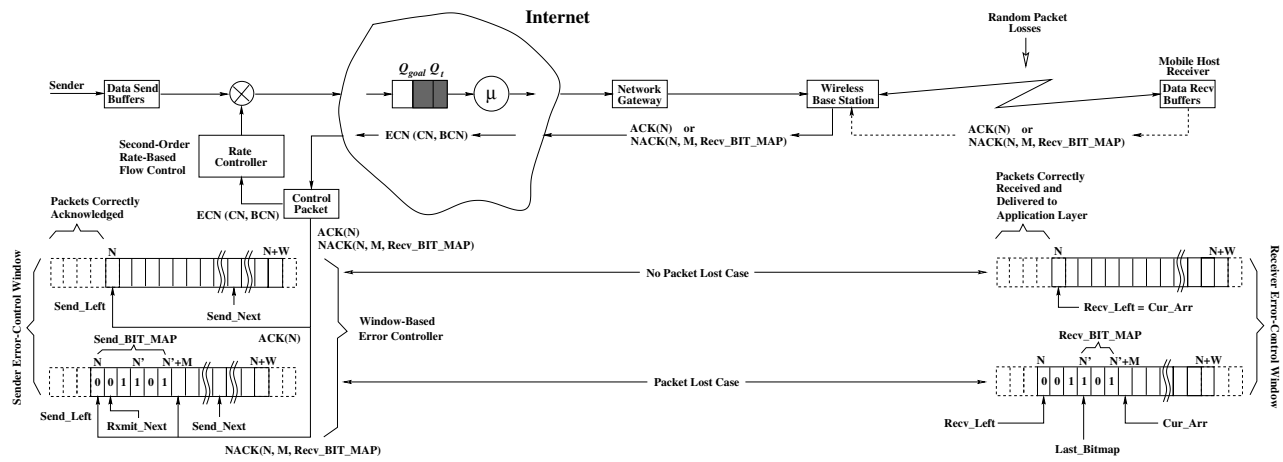
Fig. 1. The proposed flow and error control scheme for the transport protocol over wireless networks.

periodic control-packet feedback minimizes the dependency of error and flow-control performance on RTT and virtually eliminate the impact of wireless random loss on the flow control performance.

As shown in Fig. 1, a transmitted packet is not removed from the buffer until its sequence number is correctly acknowledged. When there are packet losses, the sender host need to handle three sender-buffer pointers: (i) $Send\_Left$ — the maximum packet sequence number below which all packets have been correctly acknowledged; (ii) $Send\_Next$ — the sequence number of the packet to be sent next; (iii) $Rxmit\_Next$ — the sequence number of the packet to be retransmitted. Associated with the error-control window at the sender is a sender-bitmap vector, $Send\_BIT\_MAP$ where bit 1 (0) indicates that the corresponding packet has (not) been acknowledged within the retransmission error-control window ($Send\_BIT\_MAP$) at the sender host. If no packet is lost, the sender only need to update two pointers $Send\_Left$ and $Send\_Next$.

The mobile host receiver maintains three buffer pointers (see Fig. 1): (i) $Recv\_Left$ — the maximum packet sequence number below which all packets have been correctly received; (ii) Cur_Arr — the immediate-next packet sequence number that follows the packet received most recently; (iii) $Last\_Bitmap$ — the value of Cur_Arr when sending the last feedback control packet in the last error-control cycle. If all packets are received correctly, then $Recv\_Left = Cur\_Arr$ and the receiver sends ACK($N := Recv\_Left$) to the source (see the no packet lost case shown in Fig. 1). When some packets are lost or received in error before Cur_Arr, a receiver-bitmap vector $Recv\_BIT\_MAP$ (see Fig. 1) for the current error-control cycle is used at the mobile receiver to record which packet has (not) been received correctly during the current error-control cycle. The length $M := Cur\_Arr - Last\_Bitmap$ is the increment $Recv\_BIT\_MAP$ and an NACK($N := Recv\_Left$, M, $Recv\_BIT\_MAP$) is sent to the source where $Send\_BIT\_MAP$ is concatenated with the returned $Recv\_BIT\_MAP$. The detailed pseudocode of the error-control algorithms is omitted for lack of space, but

available on-line at [15].

III. THE SYSTEM MODEL AND ANALYSIS

Using the fluid analysis [16], [17], we model a transport-layer connection under the proposed flow-control scheme as a dynamic feedback control system, which is shown in Fig. 2. We assume the existence of only a single bottleneck with queue length $Q(t)$ and a "persistent" source,[2] always having data packets to send at the rate of $R(t)$.

A. System Description and State Equations

The connection model is characterized by a set of flow-control parameters. T_f represents the "forward" delay from the source to the bottleneck, and T_b the "backward" delay from the bottleneck to the source via the mobile receiver. Clearly, $T_b = \tau - T_f$, where τ is the connection's RTT. $R(t)$ is dictated by the bottleneck's currently-available bandwidth μ. When $R(t) > \mu$, the bottleneck queue builds up, and newly-arriving packets are dropped after $Q(t)$ reaches buffer capacity ξ. The bandwidth congestion (set $CN = 1$) or buffer congestion (set $BCN = 1$) is detected if $Q(t) > Q_t$ or $Q(t) > Q_{goal}$.

The first-order (AIMD) rate control algorithm can be modeled by the following state equations:

$$R(t) = \begin{cases} R(t_0) + \alpha(t - t_0); & \text{If } Q(t - T_b) < Q_t \\ R(t_0)e^{-(1-\beta)\frac{(t-t_0)}{\Delta}}; & \text{If } Q(t - T_b) \geq Q_t \end{cases} \quad (2)$$

$$Q(t) = \int_{t_0}^{t} [R(v - T_f) - \mu]dv + Q(t_0). \quad (3)$$

where "additive increase" and "multiplicative decrease" are modeled by "linear increase" and "exponential decrease", respectively, in a continuous-time domain [17]; α (controlled by Eq. (1)) and β are rate increase and decrease factor, respectively, for a rate-adjustment interval Δ, i.e., control packet interval; t_0 is the last rate-update time point; and Q_t queue-size threshold indicating the bandwidth congestion.
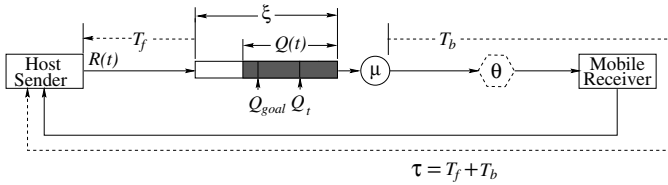
3724

Fig. 2. The system model for a transport-protocol connection with the wireless link at the last hop.

B. Wireless Link Loss Modeling

I. Data-Packet Random Losses: As shown in Fig. 2, our system model focuses on the mobile networking scenario where the only wireless link is located at the last hop as specified by Fig. 1. In such a case, the packets not dropped by the buffer overflow at the congested bottleneck link, which can be either a wired link (not at the last hop) or a wireless link (at the last hop), still face the possible random loss while traversing the wireless link at the last hop. In our loss model, we also consider the wireless link as the *erasure channel* [18] characterized by *Bernoulli(θ) loss model* [4], where each packet is lost with a *fixed constant* probability $0 < \theta < 1$ and each loss is *independent* of all other packet losses. Even though the Bernoulli(θ) loss model is not appropriate for the Internet traffic, where congestion losses can be highly correlated and bursty, the Bernoulli(θ) loss model is shown to be a highly accurate model to capture the random loss behavior of the wireless links while facilitating the analysis [4].

II. Control-Packet Random Losses: The proposed scheme regularly transmits the control packets, through the connection, with the period which is independent of data packet loss (thanks to the error and flow control decoupling) as described in Section II. Even if any feedback control packet gets lost over the random-loss wireless link, the missed flow-/error-control information in the currently lost control packet is promptly made up by the following feedback control packets arrived periodically, because the error- and flow-control information in control packet is *cumulative* — in terms of the transport protocol connection's *running sum values* of the error-control window pointer and bitmap: $Recv\_Left$ and $Recv\_BIT\_MAP$; and flow control information: CN and BCN, see Section II. Moreover, the control packets are much smaller (about 40 bytes) as compared to the data packet (560~1500 bytes), having much smaller random-loss probability than the data packets over wireless link. Thus, we can assume no control-packet loss in our analysis, which virtually doesn't affect the modeling accuracy.

C. Rate-Control Performance Analysis

Using Eqs. (2) and (3) for the case of $Q_{max} < \xi$, we derive a set of rate-control performance measures. We only list some of them to be used in this paper. The maximum queue size

---

[2]The single bottleneck and persistent-source assumption is only needed for the fluid modeling analysis [16], [17], but is not necessary for the simulations.

---

is derived as:

$$Q_{max} = \int_0^{\sqrt{2Q_t/\alpha}+\tau} \alpha t \; dt$$
$$+ \int_0^{-[\Delta/(1-\beta)]\log(\mu/R_{max})} \left(R_{max} e^{-(1-\beta)\frac{t}{\Delta}} - \mu \right) dt, \quad (4)$$

where $R_{max} = \mu + \alpha(\sqrt{2Q_t/\alpha}+\tau)$. Then, we obtain

$$Q_{max} = \frac{\alpha}{2} \left(\sqrt{\frac{2Q_t}{\alpha}}+\tau \right)^2 + \frac{\alpha\Delta}{1-\beta} \left(\sqrt{\frac{2Q_t}{\alpha}}+\frac{\mu}{\alpha}\log\frac{\mu}{R_{max}}+\tau \right). \quad (5)$$

We also obtain the *rate-control cycle T* which is determined by

$$T = \sqrt{\frac{2Q_t}{\alpha}} - \frac{\Delta}{1-\beta}\log\frac{\mu}{R_{max}} + \frac{\mu - R_{min}}{\alpha^*} + 2\tau + T_l, \quad (6)$$

where $R_{min} = \mu e^{-(1-\beta)\frac{(T_l+\tau)}{\Delta}}$, α^* is the new rate control parameter specified by Eq. (1), and T_l is the non-negative real-valued root of the following non-linear equation:

$$e^{-(1-\beta)\frac{T_l}{\Delta}} + \frac{1-\beta}{\Delta}T_l - \left(\frac{Q_{max}-Q_t}{\mu} \right)\left(\frac{1-\beta}{\Delta} \right) - 1 = 0. \quad (7)$$

Similarly, we can derive the average throughput $\overline{R}$ over T as

$$\overline{R} = \frac{1}{T}\left\{ \left[\mu + \frac{\alpha}{2}\left(\sqrt{\frac{2Q_t}{\alpha}}+\tau \right) \right]\left(\sqrt{\frac{2Q_t}{\alpha}}+\tau \right) \right.$$
$$\left. + \frac{\Delta R_{max}\left(1-e^{-(1-\beta)\frac{T_e}{\Delta}} \right)}{1-\beta} + \frac{\mu^2 - R_{min}^2}{2\alpha^*} \right\} \quad (8)$$

where $T_e = -[\Delta/(1-\beta)]\log(\mu/R_{max}) + T_l + \tau$.

D. Packet-Loss and Link-Transmission Efficiency Analysis

For a given rate-control cycle T, the **number of total packet losses**, denoted by $\mathcal{L}$, through a transport connection is the summation of the amount of congestive packet losses on the upstream wired-links and the amount of the random packet losses over the downstream wireless link, during T. So, $\mathcal{L}$ is random variable, and thus needs to be handled using statistical approach. To quantitatively evaluate the loss-control performance of the proposed scheme, we introduce the following definitions in terms of the means/expectations of a group of random variables:

Definition 1: Under the flow control system model described by Fig 2, the **mean of packet-loss ratio**, denoted by γ, and the **mean of the link-transmission efficiency**, denoted by η, are defined by

$$\gamma \triangleq \frac{\rho_t}{T\overline{R}} \quad \text{and} \quad \eta \triangleq 1 - \gamma = 1 - \frac{\rho_t}{T\overline{R}}, \quad (9)$$

respectively, where $\rho_t \triangleq E[\mathcal{L}]$ is the mean of total number of lost packets $\mathcal{L}$ (including congestive and random losses)
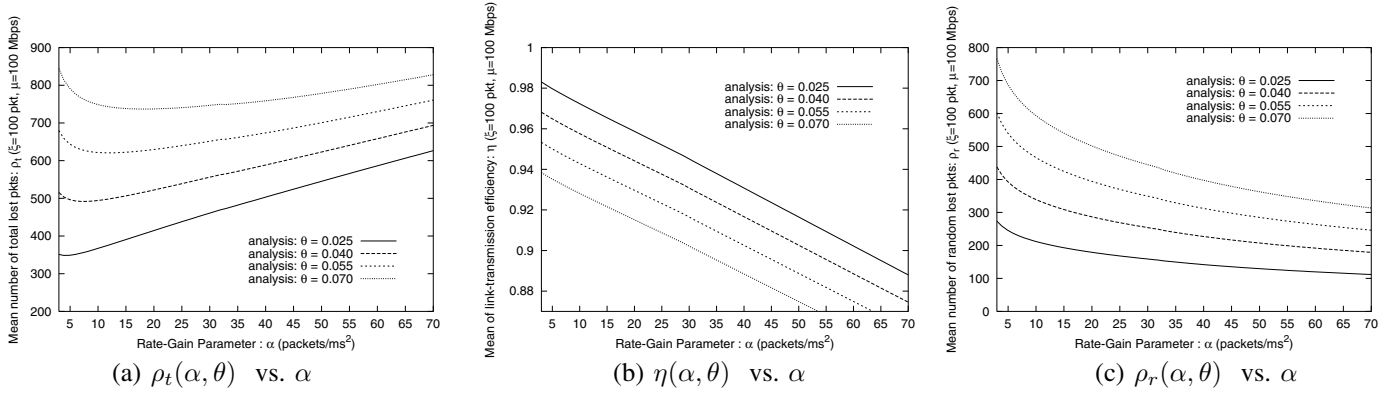
3725

(a) $\rho_t(\alpha, \theta)$ vs. α

(b) $\eta(\alpha, \theta)$ vs. α

(c) $\rho_r(\alpha, \theta)$ vs. α

Fig. 3. Numerical evaluations of the proposed scheme in wireless networks.

during T, T is the rate-control cycle specified by Eq. (6), and $\overline{R}$ is the average throughput given by Eq. (8) within T. ∎

The mean η of link-transmission efficiency is an important metric for flow and error control since it measures the average percentage of link bandwidth used by successfully-transmitted packets (without retransmitting them). The following theorem gives an explicit formula for ρ_t, from which both η and γ can be derived.

Theorem 1: If a transport connection specified by Fig. 2, with a buffer capacity $Q_t < \xi < \infty$ at the bottleneck of wired-links and a wireless link characterized by the **Bernoulli(θ) loss model** (i.e., independent random losses with the packet loss probability equal to θ) at the last hop,[3] is controlled by the proposed scheme described by Eqs. (2)–(3) and the α-control law by Eq. (1), then the mean ρ_t of total number of lost packets and the mean ρ_r of the number of randomly lost packets over the wireless link, during T, are determined by:

$$\begin{cases} \rho_t = \rho_c + \rho_r = (1-\theta)\rho_c + \theta T\overline{R}; \\ \rho_r = \theta\left(T\overline{R} - \rho_c\right); \end{cases} \quad (10)$$

where ρ_c is the number of lost packets due to the congestion at the bottleneck of wired-links during T which is given by

$$\rho_c = \begin{cases} \frac{1}{2}\,\alpha\left[\left(\sqrt{2Q_t/\alpha} + \tau\right)^2 - t_\xi^2\right] + \frac{\mu\Delta}{(1-\beta)}\log\frac{\mu}{R_{max}} \\ \quad + \frac{\Delta}{1-\beta}\left(R_{max} - \mu\right), \qquad \text{if } t_\xi \le \sqrt{2Q_t/\alpha} + \tau; \\ \mu\left[t_\xi - \tau - \sqrt{2Q_t/\alpha} + \frac{\Delta}{(1-\beta)}\left(\log\frac{\mu}{R_{max}} - 1\right)\right] \\ \quad + R_{max}e^{-\frac{1-\beta}{\Delta}\left(t_\xi - \tau - \sqrt{2Q_t/\alpha}\right)}, \quad \text{if } t_\xi > \sqrt{2Q_t/\alpha} + \tau; \end{cases} \quad (11)$$

where all variables are the same as defined in Section III-C, except that $t_\xi = \sqrt{(2\xi)/\alpha}$ if $\xi \le \left(\sqrt{Q_t} + \tau\sqrt{\alpha/2}\right)^2$ (i.e., $t_\xi = \sqrt{(2\xi)/\alpha} \le \sqrt{2Q_t/\alpha} + \tau$, which determines the value of variable t_ξ for the condition used in the first part of Eq. (11)); else t_ξ is the non-negative real-valued root of the following non-linear equation, which determines the value of variable t_ξ

[3]The wireless link can also be the transport-protocol connection-bottleneck link.

for the condition used in the second part of Eq. (11):

$$\left(\sqrt{Q_t} + \tau\sqrt{\frac{\alpha}{2}}\right)^2 + \frac{R_{max}\Delta}{1-\beta}\left(1 - e^{-(1-\beta)\frac{t_\xi - \tau - \sqrt{2Q_t/\alpha}}{\Delta}}\right) - \xi$$
$$- \mu\left(t_\xi - \tau - \sqrt{2Q_t/\alpha}\right) = 0, \text{ if } \xi > \left(\sqrt{Q_t} + \tau\sqrt{\alpha/2}\right)^2. \quad (12)$$

Proof: The proof is omitted for lack of space, but available on-line at [15]. ∎

IV. PERFORMANCE EVALUATIONS

Consider a transport connection with wired bottleneck-link bandwidth $\mu = 100$ Mbps, buffer capacity $\xi = 100$ packets; $Q_t = 50$ packets, $\tau = 4$ ms, and $q = 0.6$ for the α-control. Fig. 3(a) plots the mean ρ_t of the total number of lost packets obtained from Eqs. (10) and (11), against the rate-gain parameter α — the key parameter of the α-control — for different random-loss probabilities θ's at the wireless link (at the last hop of) the connection. $\rho_t(\alpha, \theta)$ is found to be a monotonic increasing function of θ for a given α, which is expected as the number of random lost packets increases as θ gets larger. However, interestingly $\rho_t(\alpha, \theta)$ is not a monotonically function of α for a given θ. There is the minimizer $\alpha_{min}^* \triangleq \arg\min_\alpha \rho_t(\alpha, \theta)$, which is the unique real-valued root of $\partial\rho_t(\alpha, \theta)/\partial\alpha = 0$ specified by Eq. (10). This is because the total number of packets $T(\alpha)\overline{R}(\alpha)$ transmitted during $T(\alpha)$ is a monotonic-decreasing function [19] of α while $\rho_c(\alpha)$ is a monotonic-increasing function of α (since Q_{max} is a monotonic-increasing function [19] of α). Thus, $\rho_r(\alpha, \theta) = \theta[T(\alpha)\overline{R}(\alpha) - \rho_c(\alpha)]$ specified in the second part of Eq. (10) in Theorem 1 is a monotonic decreasing function of α, as also verified in Fig. 3(c) for any given θ. Thus, if $\alpha < \alpha_{min}^*$, $\rho_r(\alpha, \theta)$ gets larger, thus dominating $\rho_t(\alpha, \theta) \triangleq \rho_c(\alpha) + \rho_r(\alpha, \theta)$. On the other hand, if $\alpha > \alpha_{min}^*$, $\rho_t(\alpha, \theta)$ also increases because $\rho_c(\alpha)$ contributes to $\rho_t(\alpha, \theta)$ significantly as α gets larger.

Using Eqs. (8) and (9), Fig. 3(b) plots the mean $\eta(\alpha, \theta)$ of link-transmission efficiency versus α-control parameter α. We observe that $\eta(\alpha, \theta)$ monotonically decreases as either α, or θ, increases, which is also expected because either random, or congestion, loss requires the lost packets to be retransmitted, thus degrading the mean link-transmission efficiency. So, the

3726

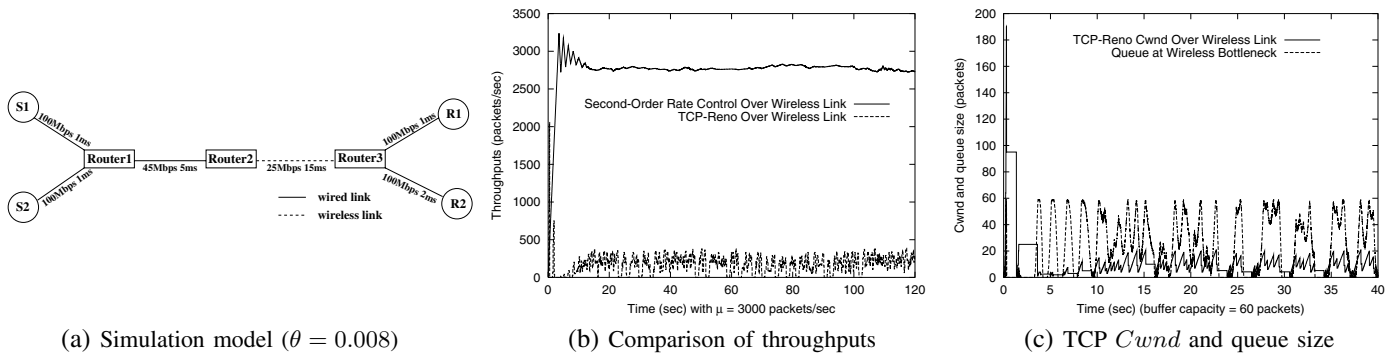(a) Simulation model ($\theta = 0.008$)   (b) Comparison of throughputs   (c) TCP $Cwnd$ and queue size

Fig. 4.   Performance comparison between α-control over wireless networks.

above numerical analysis is consistent with the valid systems dynamics of the proposed flow and error control scheme.

Using the *ns*-2 [20], we simulate the proposed scheme to validate the analyses and evaluate its performance. The network model and parameters are given by Fig. 4(a), where the only wireless link is the shared bottleneck link between Router2 and Router3 (note this wireless link is not at the last hop), shared by two types of connections: (1) TCP-Reno connection from S1 to R1, and (2) α-controlled connection from S2 to R2. As shown in Fig. 4(b), the simulated throughput under α-control is over 10 times higher than that of TCP-Reno connection. This is expected since α-control adjusts its sending rate via the explicit feedback, and is decoupled from the error control, making the throughput independent of the random loss over the shared wireless link. By contrast, as shown in Fig. 4(c), any loss at the bottleneck wireless link, whether it's random loss or congestion loss, is always inferred as the congestion by TCP source, and thus the TCP congestion control window $Cwnd$ is often unnecessarily reduced by random losses even when bottleneck queue size is much lower than buffer size (60 packets), see Fig. 4(c), i.e., no congestion, which degrades TCP-Reno's throughput significantly.

V. CONCLUSION

We proposed and analyzed an efficient flow and error control scheme for high-throughput transport protocols over wireless networks. It is built on the α-control, a second-order rate control, as well as a separate sliding-window error-control scheme. The α-control minimizes packet losses due to congestion and retransmissions by adjusting the rate-gain parameter to the variations in the number and RTTs of cross-traffic flows sharing the bottleneck. Using NACKs and selective retransmissions, our error-control scheme recovers packet losses caused by either congestion or wireless random noise. Using the fluid analysis, we modeled the proposed scheme, and derived various performance measures. Our simulation experiments confirmed the analyses, and demonstrated the superiority of the α-control to the TCP-Reno in terms of dealing with random loss, increasing the average throughput, and enhancing the robustness to the variation of wireless-link random-loss probability.

REFERENCES

[1] R. Caceres and L. Iftode, "Improvingd the performance of reliable transport protocols in mobile computing environments," *IEEE Journal on Selected Areas in Comm.,* vol. 13, no. 5, pp. 850-857, June 1995.

[2] P. Manzoni, D. Ghosal, and Serazzi, "Impapct of mobility on TCP/IP: an integrated performance study," *IEEE Journal on Selected Areas in Communications,* vol. 13, no. 5, pp. 858-867, June 1995.

[3] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Trans. on Networking,* pp. 756-769, 1997.

[4] A. Kumar, "Comparative performance analysis of versions of TCP in a local network with a lossy link," *IEEE/ACM Trans. on Networking,* vol. 6, no. 4, pp. 485-498, Aug. 1998.

[5] S. Floyd, "TCP and explicit congestion notification," A*CM SIGCOMM Computer Communication Review,* vol. 24, no. 5, pp. 10-23, Oct. 1994.

[6] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. on Networking,* vol. 1, no. 4, pp. 397-413, Aug. 1993.

[7] , H. Balakrishnan, V. N. Padmanabhan, and R. H. Katz, "The effects of asymmetry on TCP performance," in *MobiCom' 1997.*

[8] F. Kelly, "Models for a self-managed Internet," http://www.statslab.cam.ac.uk/∼frank/smi.html, Aug. 2000.

[9] V. Jacobson, "Congestion avoidance and control," *ACM SIGCOMM' 88.*

[10] W. R. Stevens, *TCP/IP Illustrated, Vol. 1,* Addison-Wesley, 1994.

[11] L. S. Brakmo and L. L. Peterson, "TCP Vegas: End to end congestion avoidence on a global Internet," *IEEE Journal on Selected Areas in Communications,* vol. 13, no. 8, pp. 1465-1480, Oct. 1995.

[12] Steven H. Low, Larry Peterson, and Limin Wang, "Understanding TCP Vegas: A duality model," in *Proc. of ACM SIGMETRICS,* 2001.

[13] H. M. Chaskar, T. V. Lakshman, and U. Madhow, "TCP over wireless with link level error control: analysis and design methodology," *IEEE/ACM Trans. on Networking,* vol. 7, no. 5, pp. 605-615, Oct. 1999.

[14] X. Zhang, K. G. Shin, D. Saha, and D. Kandlur, "Scalable flow control for multicast ABR services in ATM networks," *IEEE/ACM Trans. on Networking,* vol. 10, no. 1, pp. 67-85, Feb. 2002.

[15] X. Zhang, "Second-order rate-control based transport protocols over mobile wireless networks (full paper version)," *Networking and Information Systems Laboratory,* Department of Electrical Engineering, Texas A&M University, College Station, Technical Report, [Online.] Available: http://ece.tamu.edu/∼xizhang/papers/rw-wireless.pdf, Sept. 2006.

[16] J. Bolot and A. Shankar, "Dynamical behavior of rate-based flow control mechanism," *ACM SIGCOMM Computer Communication Review*, vol. 20, no. 4, pp. 35-49, Apr. 1990.

[17] N. Yin and M. G. Hluchyj, "On closed-loop rate control for ATM cell relay networks," in *IEEE INFOCOM,* 1994, pp. 99-109.

[18] P. Elias, "Coding for two noisy channels," in *Information Theory,* 3rd London Symposium, 1995, pp. 61-76.

[19] X. Zhang, K. G. Shin, and Q. Zheng, "Integrated rate and credit feedback control for ABR services in ATM networks," in *Proc. of IEEE INFOCOM,* Apr. 1997, pp. 1297-1305.

[20] S. McCanne and S. Floyd, "NS-LBNL network simulator," http://www-nrg.ee.lbl.gov/ns, 1996.