

Using Radio Network Feedback to Improve TCP Performance over Cellular Networks

Niels Möller, Inés Cabrera Molero, Karl H. Johansson,
Justus Petersson, Robert Skog, and Åke Arvidsson

Abstract—A new control structure is proposed to improve user experience of wireless Internet. Information on radio bandwidth and queue length available in the radio network controller (RNC), close to the base station, is used in a proxy that resides between the Internet and the cellular system. The control algorithm in the proxy sets the window size according to event-triggered information on radio bandwidth changes and time-triggered information on the queue length of the RNC. A stability analysis of the hybrid control system is performed. In a number of realistic use cases evaluated in ns-2, the new control structure is compared to a nominal one that employs end-to-end TCP Reno. It is shown that the proposed solution reduces the time to serve users, increases the radio link utilization, and decreases the required buffer size in the RNC.

I. INTRODUCTION

The available bandwidth in wireless networks is rapidly varying and follows a multi-modal distribution. It is therefore difficult to compensate for the variations with the current end-to-end mechanisms in the wired Internet, such as TCP. The large end-to-end round-trip time (RTT) that is common in wireless networks makes the control problem even harder [1], [2], [3]. For high-bandwidth and long-delay networks, TCP throughput is limited by the maximum TCP window size, which is less than or equal to 64 KB. The exponential back-off, slow-start, and congestion avoidance mechanisms in TCP further limit the throughput and may also force over-dimensioned layer-2 buffers in the cellular system.

There are several proposals for improving TCP performance over wireless networks. One approach is to improve the TCP algorithm in the end points, e.g., Eifel [4] and TCP Westwood [5]. Deploying an improved TCP version, however, requires changes at the sending server or at the receiving terminal (or sometimes both), which might be less attractive in many situations. A different approach to improve TCP over wireless is to introduce a proxy between the server and the terminal, i.e., splitting the connection into one connection between the server and the proxy, and another one between the proxy and the terminal. This approach is often denoted split TCP [6]. It has the advantage that changes in the system are made only to the proxy and possibly the terminal; the server

will see an ordinary wired network. With split TCP, one can in many cases use an arbitrary transport protocol between the proxy and the terminal. This may be an attractive option if the operator has control over the networking software running on the terminals, but when supporting off-the-shelf terminals, deployment of a specialized transport protocol is difficult.

The main contribution of the paper is a new proxy-based scheme for improving both the user experience of wireless Internet, and the utilization of existing infrastructure. We propose a new custom protocol between the radio network controller (RNC) of the cellular system and a proxy that resides between the Internet and the cellular system. The data-link layer within the RNC provides information to the transport layer within the proxy, and we refer to this communication as radio network feedback (RNF). In the RNC, we use the layer-2 trigger mechanism presented in [7]. When the radio link bandwidth changes, the RNC sends an RNF message with the radio connection's new bandwidth to the proxy. The proxy then takes appropriate action by adjusting the TCP window size. The computation of a suitable window size in the proxy also takes into consideration the queue in the RNC, which needs to be controlled at a suitable level due to delay fluctuations and other uncertainties in the cellular system. The proposed scheme does not require any changes to the terminals or to the Internet server, but they may utilize any of the standard TCP protocols. We perform a stability analysis for the dynamics of the RNC queue and the window set by the proxy. It is shown that after a bandwidth change, the system is able to dynamically adjust the queue accordingly. It is also shown using LaSalle's Theorem and a flow model approximation of the feedback control system, that the queue and the window sizes tends to a unique equilibrium point.

The rest of the paper is organized as follows. Section II presents the system architecture and introduces notation. The new control structure, with an event-triggered feedforward mechanism based on the bandwidth changes and a time-triggered feedback mechanism based on the RNC queue size, is specified in Section III. A flow model is then used in Section IV to analyze stability of the control strategies. Section V discussed the discrete-time implementation of the algorithms. Qualitative results, obtained from ns-2 simulations, on slow start, outage recovery, bandwidth adaptation, and queue length transients are presented in Section VI. A quantitative evaluation on a set of use cases is also done, showing substantial improvements on time to serve user and link utilization compared to conventional end-to-end control.

The work was partially supported by European Commission through the Network of Excellence HYCON, by the Swedish Foundation for Strategic Research through an Individual Grant for the Advancement of Research Leaders, and by the Swedish Research Council.

N. Möller, I. Cabrera Molero and K. H. Johansson are with the Department of Signals, Sensors and Systems, KTH, Stockholm, Sweden. {niels|inescm|kallej}@kth.se

J. Petersson, R. Skog and Å. Arvidsson are with Ericsson AB, Stockholm, Sweden. {Justus.Petersson|Robert.Skog|Ake.Arvidsson}@ericsson.com

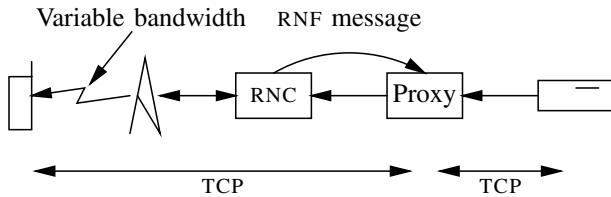


Fig. 1. Proposed architecture. The mobile terminal on the left downloads a file from the server on the right, via the proxy. During the transfer, the RNC generates RNF messages including information about the current bandwidth over the radio link, and the current RNC queue length. The proxy uses this information to adjust its sending rate.

The paper is concluded in Section VII. A simulation study motivating this work is presented in [8]. Further details and a general discussion on automatic control in TCP over wireless is given in [9].

II. ARCHITECTURE

Consider the transfer of a file, from a server on the Internet, to a mobile terminal attached via an operator's radio access network. Within the operator's network, there are two special nodes: An RNC, which among other things is responsible for allocating bandwidth to user connections, and a proxy, which acts as a gateway between the operator's network and the Internet. The endpoints use standard TCP to communicate with the proxy. The proxy, on the other hand, adapts its sending rate towards the terminal using a custom control algorithm, which is aided by extra information provided by the RNC. We make the reasonable assumptions that the bottleneck for the connection is the radio channel, and that the operator's network, between proxy and RNC, does not suffer congestion.

The architecture is illustrated in Figure 1, where two TCP connections have been established: one between the terminal and the proxy and one between the proxy and the server. The RNC sends RNF messages to the proxy, including information about the current bandwidth allocation and the queue length in the RNC. The RNF messages are sent every time the bandwidth changes, and also periodically with a relatively long period time, e.g., one second.

We will compare this system to the nominal setup, in which there is a direct TCP connection between the terminal and the server.

We use the following notation. The bandwidth of the radio link is b . τ_f is the time it takes for a packet that is sent by the proxy to reach the RNC, and τ_b is the time it takes for a packet that is forwarded by the RNC to reach the terminal, and for the corresponding acknowledgement to get back to the proxy. The corresponding RTT of the cellular network, excluding queueing delay at the RNC, is $\tau = \tau_f + \tau_b$, and the "pipe size", or bandwidth-delay product, is $b\tau$. We assume that b , τ_f , and τ_b are constant most of the time, with occasional step changes. The queue length at the RNC is denoted q , so that the RTT is $\tau + q/b$. The current window size at the proxy is denoted w .

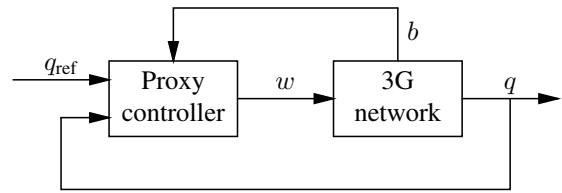


Fig. 2. Control structure. The controller uses event-triggered feedforward of the available radio link bandwidth b , and time-triggered feedback of the RNC queue length q .

III. CONTROL STRUCTURE

The objective is to achieve a high utilization of the radio link and maintain the RNC queue close to a reasonably small reference value q_{ref} , by controlling the sending rate of the proxy. Like in standard TCP, we use a window-based algorithm, but unlike standard TCP, we take advantage of explicit information provided by the RNC.

The control signal is the non-negative window size w , which indirectly determines the sending rate. The network provides the proxy controller with two kinds of information. The control structure, with feedforward of radio bandwidth b , and feedback of queue length q , is shown in Figure 2.

When the available bandwidth over the radio channel is changed, the RNC sends an RNF message to the proxy to inform it about the new bandwidth. The controller uses an event-triggered feedforward mechanism that takes advantage of this information, and resets the window size to a new value,

$$w = \hat{b}\hat{\tau} + q_{ref} \quad (1)$$

Here $\hat{\tau}$ is an estimate of the propagation delay τ , and \hat{b} is the new bandwidth from the RNC's RNF message.

Between bandwidth updates, the RNC periodically sends RNF messages with the current queue length. The controller compares this feedback information to the reference value and adjusts its window,

$$w_{k+1} = w_k + c(q_{ref} - q_{k+1}) \quad (2)$$

The feedback loop uses a quite long sampling time, on the order of one second, and it is designed to compensate for the bias caused by a feedforward controller based on uncertain information. On shorter time scales, the transmission window is fixed, and transmission is governed by the usual ACK clock.

IV. STABILITY ANALYSIS

To investigate the stability of the control mechanisms, we use a flow level model. The system input signal is $w(t)$ and the output signal is $q(t)$. The sending rate is related to the window size by

$$r(t) = \frac{w(t)}{\tau + q(t - \tau_b)/b}$$

where $\tau + q(t - \tau_b)/b$ is the RTT corresponding to the queue length at the time the most recently acknowledged packet was forwarded by the RNC. The rate of change of the queue

(as long as it is neither full nor empty) equals the difference between the received rate and the bandwidth,

$$\dot{q}(t) = r(t - \tau_f) - b = \frac{w(t - \tau_f) - b\tau - q(t - \tau)}{\tau + q(t - \tau)/b} \quad (3)$$

A. Feedforward control

With feedforward control, the sender uses estimates \hat{b} and $\hat{\tau}$ to compute a constant control signal according to (1). Substituting this control into the queue dynamics (3), gives

$$\dot{q}(t) = \frac{\hat{b}\hat{\tau} + q_{\text{ref}} - b\tau - q(t - \tau)}{\tau + q(t - \tau)/b}$$

This time-delayed differential equation has an equilibrium point $q^* = q_{\text{ref}} + \hat{b}\hat{\tau} - b\tau$, and can be rewritten as

$$\dot{q}(t) = -\frac{q(t - \tau_f) - q^*}{\tau + q(t - \tau_f)/b}$$

Define $\tau^* = \tau + q^*/b$ as the RTT corresponding to the equilibrium. The linearized dynamics are

$$\dot{q}(t) = -\frac{1}{\tau^*}(q(t - \tau) - q^*)$$

The Nyquist criterion gives asymptotic stability since

$$\frac{\tau}{\tau^*} < 1 < \frac{\pi}{2}$$

To estimate the convergence time, consider the poles of the linearized system, i.e., the solutions to

$$s + \frac{e^{-\tau s}}{\tau^*} = 0$$

Since the system is stable, all solutions lie in the half plane $\text{Re } s < 0$. Note that for any such pole, it holds that $|s| = e^{-\tau \text{Re } s} / \tau^* > 1/\tau^*$, so the poles lie outside of the disk of radius $1/\tau^*$ around the origin. Let us derive a bound $\lambda > 0$ for the stability exponent of the system, i.e., a bound such that $\text{Re } s < -\lambda$ for all solutions to the characteristic equation above. The trajectories will locally converge to the equilibrium faster than $e^{-\lambda t}$, and $1/\lambda$ is an upper bound for the system time constant. We use the following result [9].

Lemma 1: The equation $\alpha x = e^x$, with $\alpha > 0$, has a solution $x > 0$ if and only if $\alpha \geq e$. Furthermore, the solutions satisfy $1/(\alpha - 1) < x < 2\alpha$.

First consider a real pole, $s = -x$. Then $x = e^{\tau x} / \tau^*$. With $\alpha = \tau^* / \tau$, the first part of the lemma yields $\tau^* \geq e\tau$, and the second part gives $\tau x > 1/(\tau^* / \tau - 1)$, or $x > 1/(\tau^* - \tau)$. So in this case, we have the bound $\lambda > 1/(\tau^* - \tau)$.

Next, consider a complex pole in the left half plane, $s = -x + iy$, with $x, y > 0$. Again we want to find a bound $\lambda > 0$ such that all poles are to the left of $\text{Re } s = -\lambda$. So assume that $x \leq \lambda$. Denote $z = e^{-\tau s} = e^{\tau x}(\cos \tau y - i \sin \tau y)$, and assume that $\tau^*(x - iy) = -\tau^* s = z$, so that s is a pole of the system. First, note that

$$\tau^* y \leq |\tau^* s| = |z| = e^{\tau x} \leq e^{\tau \lambda}$$

Require λ to be such that $e^{\tau \lambda} \tau / \tau^* < \pi/2$, which is always possible for some $\lambda > 0$. Then \cos is decreasing on the interval $(\tau y, \pi/2)$. We get for the real part

$$\tau^* \lambda > \tau^* x = \text{Re}(-\tau^* s) = \text{Re } z = e^{\tau x} \cos \tau y > \cos \frac{\tau e^{\tau \lambda}}{\tau^*}$$

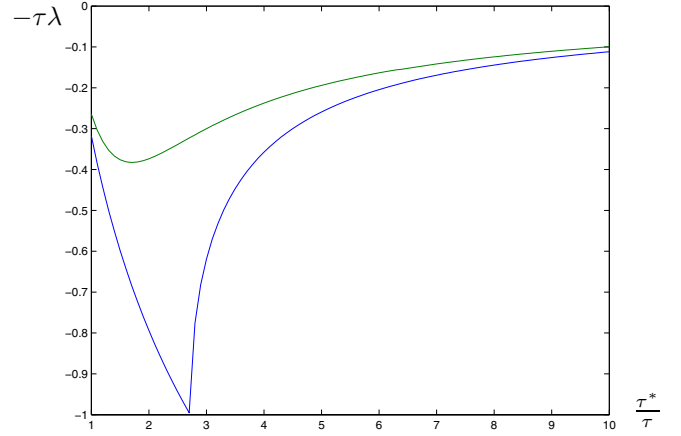


Fig. 3. Stability exponent $-\lambda$ for $\tau = 1$. The upper curve is the derived conservative estimate $-\lambda = \min(1/(\tau^* - \tau), \lambda')$. The lower curve is computed numerically using an order seven Padé approximation for the delay.

This implies that $\lambda > (1/\tau^*) \cos(e^{\tau \lambda} \tau / \tau^*)$. So if we choose λ small enough, this inequality is violated and then there can be no poles with $x \leq \lambda$. Hence, define

$$\lambda' = \{\text{smallest positive solution } \lambda, \text{ to } \tau^* \lambda = \cos \frac{\tau e^{\tau \lambda}}{\tau^*}\}$$

We obtain the general bound for the stability exponent as

$$\lambda = \min\left(\frac{1}{\tau^* - \tau}, \lambda'\right)$$

Note that $\tau\lambda$ depends only on the ratio τ^*/τ . Figure 3 shows $\tau\lambda$ as a function of τ^*/τ . The convergence speed peaks for $\tau^* = e\tau$, which is when the system has a real double pole at $-1/\tau$. It is easy to obtain the bound $\lambda > 1/(4\tau)$ for the interval $\tau < \tau^* < e\tau$. To summarize, we have the following result.

Proposition 1: The feedforward control results in an asymptotically stable system, with an equilibrium, which corresponds to a deviation from the reference value q_{ref} that equals the estimation error for the pipe size, $\hat{b}\tau - b\tau$. As long as $\tau^* < e\tau$, the convergence time constant for the linearized system is smaller than 4τ .

B. Feedback control

Next, consider feedback control of the window size, using queue length samples. The primary objective of the feedback control is to cancel the bias that results when using feedforward control from estimates \hat{b} and $\hat{\tau}$ that are uncertain. A natural approach is to use an integrating controller,

$$\dot{w}(t) = -c'(q(t - \tau_b) - q_{\text{ref}}) \quad (4)$$

With this control law, the closed loop system is described by the equations

$$\begin{cases} \dot{q}(t) = \frac{w(t - \tau_f) - b\tau - q(t - \tau)}{\tau + q(t - \tau)/b} \\ \dot{w}(t) = -c'(q(t - \tau_b) - q_{\text{ref}}) \end{cases} \quad (5)$$

Since q and w are non-negative, these equations are valid only in the interior of the domain $q, w \geq 0$. On the border $w = 0$ we have instead

$$\dot{w}(t) = \max\left(0, -c'(q(t - \tau_b) - q_{\text{ref}})\right)$$

and similarly for \dot{q} on the line $q = 0$.

The equations have an equilibrium point given by $q^* = q_{\text{ref}}$ and $w^* = b\tau + q_{\text{ref}}$. The RTT in stationarity is $\tau^* = \tau + q_{\text{ref}}/b$. With this notation, the system can be written as

$$\begin{cases} \dot{q}(t) = \frac{w(t - \tau_f) - w^* - (q(t - \tau) - q^*)}{\tau^* + (q(t - \tau) - q^*)/b} \\ \dot{w}(t) = -c'(q(t - \tau_b) - q^*) \end{cases}$$

Note that the queue delay influences the system dynamics via the gain in the expression for \dot{q} , but the queue is *not* in the signaling path. In the RNF architecture, the RNF messages say how long the queue is, but the RNF messages themselves do not suffer any queueing delay. In other words, the time delays in the differential equations are constant and not depending on the system state.

In the following analysis, we ignore the propagation delay $\tau = \tau_f + \tau_b$. We will show that in this case the system is globally asymptotically stable. Introduce $\tilde{q} = q - q^*$ and $\tilde{w} = w - w^*$. Then, the non-linear system without propagation delay can be written as

$$\begin{cases} \dot{\tilde{q}}(t) = \frac{\tilde{w} - \tilde{q}}{\tau^* + \tilde{q}/b} \\ \dot{\tilde{w}}(t) = -c'\tilde{q} \end{cases}$$

We make a change of variables by introducing a “virtual time” s , defined by $dt = (\tau^* + \tilde{q}/b)ds$. The virtual time corresponds to the number of RTTs. Then

$$\begin{cases} \frac{d\tilde{q}}{ds} = \tilde{w} - \tilde{q} \\ \frac{d\tilde{w}}{ds} = -c'\tau^*\tilde{q} - \frac{c'}{b}\tilde{q}^2 \end{cases}$$

These differential equations are still valid only in the interior of the region $w, q \geq 0$, non-negativity has to be enforced at the borders. The phase portrait of this system (for unit parameters) is shown in Figure 4. Our goal is to show that this system is globally asymptotically stable in the region $w, q \geq 0$. The Lyapunov function

$$V(\tilde{q}, \tilde{w}) = \tilde{q}^2 \left(\tilde{q} + \frac{3b\tau^*}{2} \right) + \frac{3b}{2c'}\tilde{w}^2$$

and LaSalle’s theorem we can prove the following result:

Proposition 2: The feedback control system, with negligible propagation delay in the cellular network, is globally asymptotically stable.

V. IMPLEMENTATION

In practice, the sender does not have access to the continuous-time function $q(t)$, but rely on samples sent from the RNC. Denote the sampling time by T . For the flow model to be valid, an averaging over at least one RTT is needed, so T should not be shorter than the RTT. We next consider

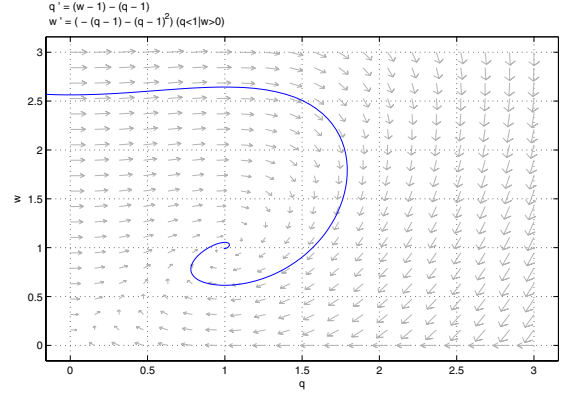


Fig. 4. Phase portrait of q - w dynamics for the feedback control system.

the case of a T several times larger than the RTT. (In the simulations presented in next section, the sampling time is one second.)

Between samples, the controller keeps w constant. The queue dynamics will converge to the stationary value within about twice the system time constant. From the analysis for a constant window size, in Section IV-A, we know that the stationary value is $q^* = w - b\tau$. Hence, if T is sufficiently large, and $w(t) = w_k$ for $kT < t \leq (k+1)T$, then

$$q_{k+1} = q((k+1)T) \approx w_k - b\tau$$

The continuous-time integrating controller corresponds to the discrete-time controller

$$w_{k+1} = w_k + c(q_{\text{ref}} - q_{k+1})$$

It turns out that $c = 1$ is a good choice for the controller gain, because then

$$q_{k+2} \approx w_{k+1} - b\tau \approx w_k + q_{\text{ref}} - (w_k - b\tau) - b\tau = q_{\text{ref}}$$

The queue will thus converge to the target value within two samples, $2T$.

VI. RESULTS

The proposed proxy controller takes advantage of RNF messages and thus gets better performance out of the link compared to standard TCP. The gain has three main components: Connection startup, outage recovery, and bandwidth adaptation. They are discussed in detail below. After that, the RNC queue transients are examined, and at the end of the section, we describe a quantitative study for three realistic use cases.

Several advantages with the proxy controller is illustrated by the ns-2 simulations shown in Figure 5, which depicts available bandwidth (dotted), sending rate for proxy controller (dashed), and sending rate for nominal end-to-end TCP (solid). In this figure, note that the transmission via the proxy is completed in about 51 s, while the end-to-end TCP transmission lasts approximately 10 s longer.

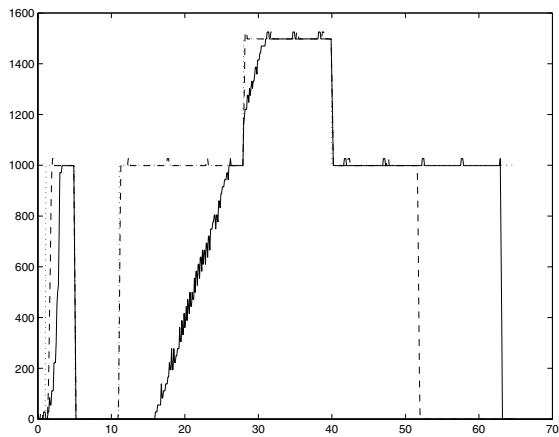


Fig. 5. Available bandwidth (dotted), sending rate for proposed proxy controller (dashed), and sending rate for nominal end-to-end TCP (solid). The link under-utilization is visible as the area between the bandwidth and the sending rate curves. The sending rates in this and the other figures in this chapter are in kbit/s, averaged over 300 ms. The time axis is in seconds. (That the sending rates sometimes seem to exceed the available bandwidth is an artifact of the sampling of the ns-2 simulation.)

A. Connection startup

The connection startup phase is illustrated in the very beginning of Figure 5. A zoom in is shown at the top of Figure 6. The initial delay before the sending rate starts to rise is the RTT between terminal and server. The rate curve for nominal TCP is governed by the slow start mechanism of TCP. The flow from the proxy to the terminal does not use slow start at all. However, the flow from the server to the proxy does use slow start, which explains why the rate curve for the proxy can not jump instantaneously to 1 Mbit/s. The reason that the proxy solution outperforms nominal TCP is that the the RTT between the proxy and the server is much shorter than the RTT between the terminal and the server.

B. Outage recovery

When an outage occurs, TCP goes into timeout and retransmits lost packets repeatedly using exponential backoff. It will not notice that the outage has ended until it receives an ACK for one of the retransmitted packets, which in the worst case can be 60 seconds after the outage ended. This is illustrated in the middle of Figure 6, which zoom in the second section (after slow start) of Figure 5. Furthermore, when the outage has ended, TCP goes into congestion avoidance mode, where the sending rate is increased only linearly. The proxy solution does not suffer from this additional delay, since it gets explicit information from the RNC when the outage starts and ends. After the outage ends, the proxy immediately starts sending at the right rate.

C. Bandwidth adaptation

When the bandwidth of the radio channel is increased, a TCP sender in congestion avoidance mode increases its sending rate slowly. When the bandwidth is decreased, the TCP sender will not notice until some packet is lost due to buffer overflow in the RNC (in our simulations this does not

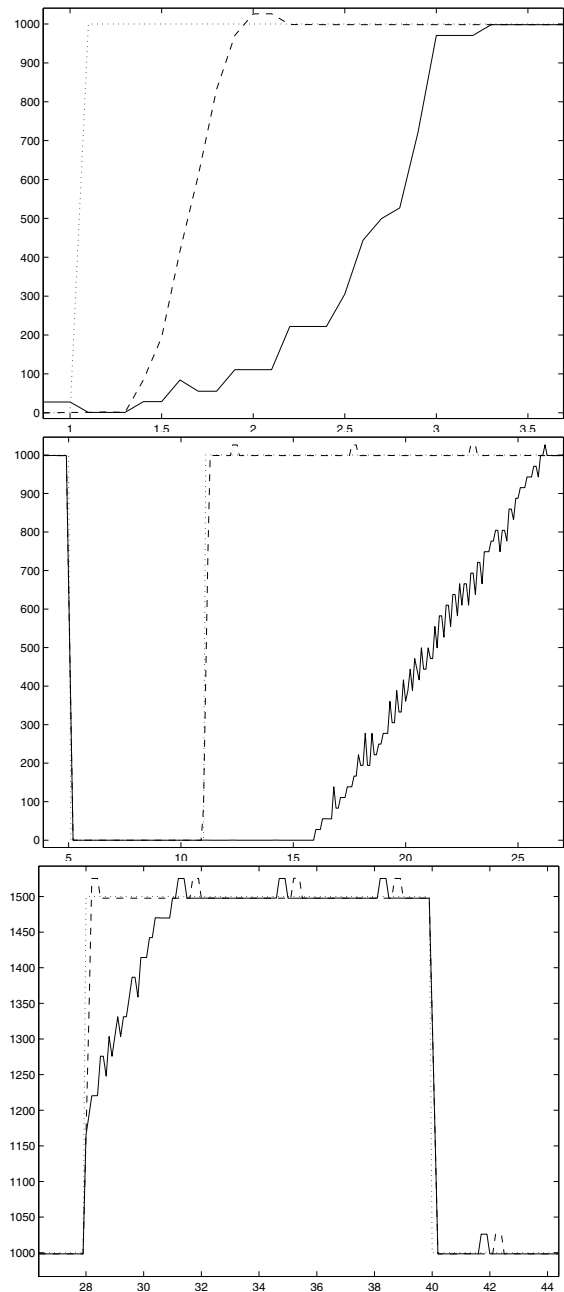


Fig. 6. Zoomed in details of Figure 5. Top: Confection startup. The response for the proxy controller is limited by the slow start of the server-proxy connection. It is still much faster than the response of end-to-end TCP. Middle: Outage recovery. Since the proxy controller does not use exponential backoff or congestion avoidance, it recovers much faster than the nominal TCP from an outage. Bottom: Bandwidth adaptation. The proxy controller reacts faster to radio bandwidth changes than the nominal TCP controller.

happen, since we have a very large RNC buffer). Thanks to the feedforward mechanism in the proxy controller, it reacts more quickly, as illustrated at the bottom of Figure 6. The performance gain is not as significant as the gains in the slow start and outage recovery phases.

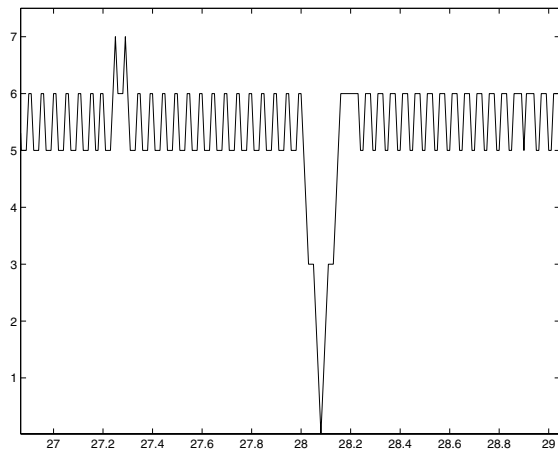


Fig. 7. Evolution of the queue length, when the bandwidth is increased at 28 s in the simulation (see Figure 5). The target value for the queue length is 6 packets.

D. RNC queue transient

The size of the RNC queue is time-varying. When the bandwidth of the radio channel is changed, the queue will initially drift off from the reference value, cf., the analysis in Section IV. The magnitude of the variation depends on the propagation delay for the RNF messages. A typical value of the propagation delay is 50 ms, but can depend on the implementation as well as dynamic behavior. Figure 7 shows the evolution of the queue length when the bandwidth is changed. It indicates what is a reasonable size of the RNC buffer.

E. Quantitative evaluation

To evaluate the the performance of the proposed controllers, ns-2 simulations were performed for three realistic use cases. The proxy setup in Figure 1 is compared to a nominal setup, with a single direct TCP connection between the terminal and the server. The links between the server and the RNC are set to 2 Mbit/s. The wireless link between the RNC and the terminal has a lower time-varying bandwidth and a one-way delay of 60 ms. We also use an average RTT (server-terminal-server) of 300 ms. The slow start threshold was set to 50, to ensure that the TCP sender does not enter the congestion avoidance phase prematurely.

The performance measures we use are the average link utilization and the time to serve a user (TTSU). The TTSU is defined as the time elapsed from when the connection is established (SYN packet sent by the requester) until the last data packet is received at the terminal. The three use cases model common services to be provided over a WCDMA system. Use case 1 models a 4 Mbyte file transfer. Use case 2 imitates web browsing and is based on statistical data [10], [11]. In this use case, three files (whose sizes vary between 51.5 Kbyte and 368.5 Kbyte) are downloaded to a terminal during one web session. Use case 3 models a large (25 Mbyte) file transfer. The results are summarized in the following table:

		TTSU [s]	Utilization [%]
Use case 1	Nominal	47.3	77%
	Proxy	39.2	98%
Use case 2	Nominal	3.9	51%
	Proxy	3.0	78%
Use case 3	Nominal	233.6	61%
	Proxy	220.5	98%

The TTSU is particularly improved with the proxy controller for relatively small file downloads (such as web browsing). Link utilization is close to 100% for the proxy controller for large files.

VII. CONCLUSIONS

TCP-based applications in a WCDMA system were studied in two setups: a nominal one that employs end-to-end TCP Reno and a new one with and intermediate proxy. The proxy solution uses RNF-messages from the data-link layer in the RNC to the transport layer in the proxy. The proxy uses feedforward control to adjust to a varying bandwidth, and feedback control to maintain the RNC queue length close to a desired value. It was shown that the resulting closed-loop system is stable and that the convergence time is linked only to the propagation delay in the cellular system. The performance gain was estimated using ns-2 simulations. In these simulations, we considered three realistic use cases. The proxy controller was able to reduce the time to serve users and increase the radio link utilization. The proposed control architecture might substantially improve the user experience of wireless Internet. The proposal is transparent to both end points, so no modifications need to be done to the terminal or the server.

REFERENCES

- [1] M. Kazantzidis and M. Gerla, "End-to-end versus explicit feedback measurement in 802.11 networks."
- [2] D. Dutta and Y. Zhang, "An early bandwidth notification (EBN) architecture for dynamic bandwidth environment," in *IEEE International Conference on Communications*, vol. 4. IEEE, April 2002.
- [3] H. Lim, K. Xu, and M. Gerla, "TCP performance over multipath routing in mobile ad hoc networks."
- [4] R. Ludwig and R. H. Katz, "The Eifel algorithm: Making TCP robust against spurious retransmissions," *ACM Computer Communication Review*, vol. 30, no. 1, January 2000.
- [5] S. Mascolo, C. Casetti, M. Gerla, M. Y. Sanadidi, and R. Wang, "TCP Westwood: bandwidth estimation for enhanced transport over wireless links," in *MobiCom*, Rome, Italy, 2001.
- [6] J. Border, M. Kojo, J. Griner, G. Montenegro, and Z. Shelby, "Performance enhancing proxies intended to mitigate link-related degradations," RFC 3135, June 2001.
- [7] S. Chemiakina and J. Pettersson, "Radio network based adaptation method for packet switched services over 3G mobile systems." *IST Mobile and Wireless Communications*, June 2004.
- [8] I. C. Molero, N. Möller, J. Pettersson, A. A. R. Skog, O. Flärdh, and K. H. Johansson, "Cross-layer adaptation for TCP-based applications in WCDMA systems," in *IST Mobile & Wireless Communications Summit*, Dresden, 2005.
- [9] N. Möller, "Automatic control in TCP over wireless," Licentiate Thesis, September 2005.
- [10] C. H. Muntean, J. McManis, and J. Murphy, "The influence of Web page images on the performance of Web servers," *Lecture Notes in Computer Science*, vol. 2093, 2001.
- [11] B. A. Huberman, P. L. T. Piroli, J. E. Pitkow, and R. M. Lukose, "Strong regularities in World Wide Web surfing," *Science*, vol. 280, no. 5360, pp. 95–97, 1998.