Stability of window-based queue control with application to mobile terminal download*

Niels Möller, Karl Henrik Johansson and Krister Jacobsson†

Abstract

Window-based transmission control, such as TCP, is a cascaded control system with an inner and an outer loop. The inner loop works on a per-packet time scale, and is governed by the so called ACK-clock. The outer loop adjusts the sending window based on an estimate of the network state. In this paper, we analyze the behaviour of the inner loop in a bottleneck topology with constant cross traffic. It is shown that the inner loop is globally asymptotically stable and that the time constant for local convergence is smaller than four times the roundtrip time. These results are applied to the design of a new outer loop control mechanism for mobile terminal download. 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 is window-based and sets the window size according to event-triggered information on radio bandwidth changes and time-triggered information on the queue length of the RNC. The properties of this control scheme is analysed.

1 Introduction

An important objective for current research on transport protocols, such as TCP and related mechanisms, is to operate the bottleneck queues at a reasonably small queue size. If this can be achieved, we can gain better performance for any network traffic that is sensitive to delay. It also makes it possible to operate links closer to their actual capacity, i.e., to use less over-provisioning. It is argued in the paper that to improve both end-to-end performance and link usage a better understanding of the dynamical properties of both the inner and outer control loops of window-based transmission control is needed. When transport protocols are analyzed in the literature, the focus is mainly on the dynamics of the window size and of the queue lengths. Here we discuss also the properties of the inner control loop, which is heavily affected by the acknowledgement arrival process.

There is a sharp distinction between the transmission queues for bottleneck links and the queues for non-bottleneck links in the current Internet, with mostly drop-tail queues. The queue for a non-bottleneck link is naturally almost empty. The queue for a bottleneck link (or "bottleneck queue", for short) on the other hand, is almost full. This is a consequence of the simple drop-tail queueing discipline and the congestion avoidance mechanism of TCP. It has a fundamental influence on the achievable network performance. It is also important to note that the maximum queue capacity (i.e., the size of a full queue) is usually quite large. The rule of thumb, documented in RFC 3439 [3], is to use 250 ms times the link capacity, where 250 ms is a reasonable upper bound on end-to-end roundtrip time on the Internet. This recommendation implies that the delay across a path in the network is the propagation delay plus 250 ms for each bottleneck link on the path. Large delays are undesirable for several reasons, including degraded quality for real-time traffic (e.g., voice over IP) and interactive traffic (e.g., telnet sessions and multi-player games). Large delays in the core network are particularly undesirable, which is why most of the backbone

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

†School of Electrical Engineering, Royal Institute of Technology, SE-100 44 Stockholm, Sweden, {niels,kallej,kringlan}@s3.kth.se

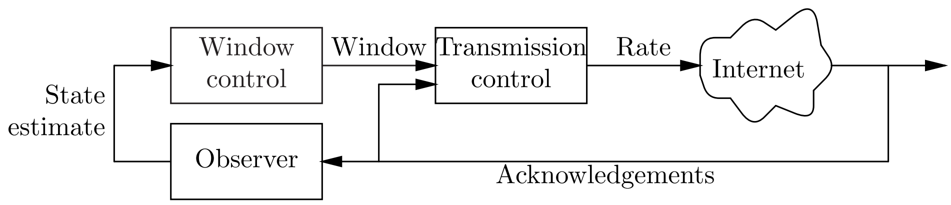

Figure 1: Window-based transmission control viewed as a cascaded control system: an inner loop based on the ACK-clock and an outer loop governed by a window controller. The focus of this paper is on the analysis of the inner loop.

links in the Internet are highly over-provisioned.

1.1 Window-based transmission control

The control structure for a traditional end-to-end window-based transmission control is illustrated in Figure 1. The controller consists of three blocks denoted transmission control, window control and observer. The transmission control sets the sending rate indirectly based on the transmission window and the acknowledgement (ACK) arrival process. Recall that the transmission window is the number of packets that are sent before waiting for an ACK. The window control sets the window size based on state estimates, such as the loss rate, the roundtrip time, the available bandwidth, and the level of congestion. These estimates are provided by the observer. Note that window-based transmission control has a cascaded control structure with an inner and an outer loop. Such control structures are common in many control applications and their properties are fairly well understood.

The main control objectives of the window-based transmission control are to avoid network congestion, to maintain a high utilization of available network resources, and to fairly share the available resources between users. It is also desirable to maintain small bottleneck queues, and hence small queueing delays, but this not achieved in the current Internet. Let us discuss some of the proposals in the literature to reduce the delays.

A simple approach to decrease queue delay is to continue to operate the queues close to maximum capacity, but significantly cut down the buffer sizes. The rule of thumb in RFC 3439 [3] for buffer provisioning is questioned in [1], where it is argued that backbone routers should use buffers that are an order of magnitude smaller than given by the rule of thumb.

It has been pointed out that keeping the queue size of bottleneck queues small can be seen as a distributed control problem, and that control theory provides some of the tools needed for designing network and end-to-end mechanisms for this problem [5, 8]. A special case of this control problem is illustrated in Figure 1. At a sending node, the measurements available to the controller are the arrival and timing of acknowledgements (ACKs). The control signal is the sending rate, which is usually controlled indirectly via the size of the transmission window. At a bottleneck router, the available measurements are the queue size and the timing of arriving packets. The possible control actions are limited. The router can decide which packets to drop. With active queue management (AQM), packets are dropped randomly, and the control signal is the packet drop probability.

Explicit congestion notifications (ECN) provides an explicit, though still severely limited, information channel between routers and end nodes. ECN provides a single mark bit per packet, shared by all routers along the path, which can be used for signalling back to the sending node. When AQM is used, and packets are dropped even though the queue is not full, setting the ECN bit is an attractive alternative to actually discarding the packets. It seems that use of ECN is essential for achieving good end-to-end quality with AQM [9].

TCP variants that try to control the queue sizes by adjusting the congestion window based on RTT measurements, without relying on ECN or AQM inside the network, are called delay-based congestion avoidance algorithms (DCA). TCP Vegas [2] and FAST TCP [16] are two well-known algorithms in this class. In [10], it is argued that the correlation between delay changes and losses is too weak for delay-based congestion avoidance to be effective, and that the control is severely disturbed by short-term queue fluctuations.

1.2 ACK-clock

The TCP algorithms control the sending rate indirectly, via the window size. The window size is the number of packets that are sent, but which for no ACK has yet been seen. A new packets is transmitted when an ACK for the oldest outstanding packet is received, which is referred to as the ACK-clock: The transmission of new packets is controlled or "clocked" by the stream of received ACKs.

The original motivation for the ACK-clock was robustness in the presence of network congestion:

> *... the packet flow is what a physicist would call 'conservative': A new packet isn't put into the network until an old packet leaves. The physics of flow predicts that systems with this property should be robust in the face of congestion.*
>
> Jacobson [6]

The ACK-clock has other useful properties, too. The focus of this paper is to view the ACK-clock as a transmission control inner loop, as illustrated in Figure 1, and analyze its stability properties.

The design of the outer loop, i.e., the window adjustment mechanism, is the object of much of the research in transport protocols. The ACK-clock is then often ignored:

> *These models do not adequately capture the self-clocking effect where a packet is sent only when an old one is acknowledged, except briefly and immediately after the congestion window is changed. This automatically constrains the input rate at a link to the link capacity, after a brief transient, no matter how large the congestion windows are set.*
>
> Wang et al. [16]

It is sometimes unfortunate not to consider the ACK-clock mechanism, especially, since it has stabilizing properties in itself. The ACK-clock operates at a per-packet time-scale and is therefore better suited to handle short-term queue fluctuations, than the outer-loop control that adjusts the window and operates at an RTT-timescale.

1.3 Contribution

The main contribution of this paper is a proof that the inner loop of window-based congestion control is globally asymptotically stable, with a local convergence time constant smaller than four roundtrip times. It is argued that when stabilizing the queue sizes, we can gain better performance if we take advantage of the ACK-clock inner loop, and divide the work to be done between inner and outer loop in an appropriate way. The inner loop should be responsible for stabilizing the queues. The outer loop, working at a slower time scale, should be responsible for fairness and for adaptation to changes in the network state. To illustrate this design paradigm, the results of the paper are applied to the design of a new outer loop control mechanism for mobile terminal download. 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 is window-based and sets the window size according to event-triggered information on radio bandwidth changes and time-triggered information on the queue length of the RNC. The properties of this control scheme is analysed.

The rest of the paper is organized as follows. Section 2 describes the fluid model for the system dynamics, and in Section 3 we show that it is a well-posed model. The main stability result is proved in Section 4, and a bound for the convergence time constant is derived in Section 5. In Section 6 we consider an application to mobile terminal download, with explicit cross-layer communication between the TCP sender and the radio network controller (RNC), which manages the wireless channel. The RNC sends a radio network feedback (RNF) message whenever the capacity of the channel is changed, and the TCP sender uses this information for feedforward control of the window size. Finally, in Section 7, we summarize our results, and draw some conclusions on how the control work should be divided between inner and outer loop in general.

2 System model

As illustrated in Figure 1, the transmission control consists of an inner loop, the ACK-clock, and an outer loop, which adjusts the window size based on the estimated network state. In this and the next few sections, we analyze the system behavior when closing the *inner* loop. This closed-loop system has the window size as the input signal, and the resulting queue size as the output signal.

We study a flow through the network, with a single bottleneck on the path. The capacity of the bottleneck link is denoted c, and the queue size $q(t)$. The forward and backward delays are denoted τ_{f} and τ_{b}, where τ_{f} is the time it takes for a packet that is transmitted by the sender to reach the queue at the bottleneck link, and τ_{b} is the time it takes for a packet that leaves the queue to reach the receiver, and for the corresponding ACK to travel back to the sender. The end-to-end roundtrip time is thus $\tau_{\mathrm{f}} + q(t)/c + \tau_{\mathrm{b}}$. We also define $\tau = \tau_{\mathrm{f}} + \tau_{\mathrm{b}}$, which simply is the roundtrip time excluding the queueing delay at the bottleneck link.

2.1 Sending rate

The average sending rate is one window of data per roundtrip time. In the fluid model, this leads to the sending rate

$$r(t) = \frac{w(t)}{\tau + q(t - \tau_{\mathrm{b}})/c} \tag{1}$$

There is an important simplification in this equation, besides the use of a fluid model: We assume that an ACK carries information about the queue length at the time the corresponding packet left the queue. This simplification gives constant time delays. It may in some cases be more realistic to assume that an ACK carries information about the queue length at the time the corresponding packet *arrived* to the queue, but this case is not covered in the paper.

2.2 Queue dynamics

The arrival rate at the queue is given by $r(t-\tau_{\mathrm{f}}) + r_{\mathrm{x}}(t)$. Here r_{x} denotes the cross-traffic, i.e., all packets that arrive to the queue and which do not belong to the flow under consideration. The queue evolves according to

$$\dot{q}(t) = \begin{cases} r(t-\tau_{\mathrm{f}}) + r_{\mathrm{x}}(t) - c & q(t) > 0 \\ \max\left(0, r(t-\tau_{\mathrm{f}}) + r_{\mathrm{x}}(t) - c\right) & q(t) = 0 \end{cases} \tag{2}$$

where the derivative should be interpreted as the right-hand derivative. To investigate the stability when closing the inner loop, we assume that the window size w and the cross-traffic $r_{\mathrm{x}}(t) < c$ are constant. Note that we can then replace c by $c - r_{\mathrm{x}}$ and r_{x} by 0, without any loss of generality.

Substituting (1) into (2) yields

$$\dot{q}(t) = \begin{cases} \dfrac{q^* - q(t-\tau)}{\tau + q(t-\tau)/c} & q(t) > 0 \\ \max\left(0, \dfrac{q^* - q(t-\tau)}{\tau + q(t-\tau)/c}\right) & q(t) = 0 \end{cases} \tag{3}$$

where $q^* = w - \tau c$. If $w \le \tau c$, then $\dot{q} \le 0$ for all t, hence $q(t)$ converges to some non-negative value. This value must be the stationary point $q = 0$. In the following, we assume that $w > \tau c$, and in this case, q^* is the equilibrium queue size.

3 Well-posedness

In this section we show that (3) is a well-posed model, by showing existence and uniqueness of its solutions.

Equation (3) is a delayed differential equation [4, 14]. The state of a system is the information needed at a time t_0, in order to predict the trajectory at time $t > t_0$. For a delay differential equation, the state is infinite dimensional since we need to know the complete trajectory over an interval to predict its future. For the system (3), that interval is $[t_0 - \tau, t_0]$. The state vector is denoted $q_t(\theta)$ and defined by

$$q_t(\theta) = q(t+\theta) \qquad \theta \in [-\tau, 0] \tag{4}$$

The initial state is denoted ϕ_0, corresponding to the initial condition $q(t) = \phi_0(t)$, for $t \in [-\tau, 0]$. We have the following well-posedness result.

Theorem 1 *For any non-negative and right-hand continuous function ϕ_0 on $[-\tau, 0]$, there exists a uniquely determined function $q(t)$ such that $q(t) = \phi_0(t)$ for $t \in [-\tau, 0]$ and $q(t)$ satisfies* (3) *for $t \ge 0$.*

Proof: The existence of a solution follows from the "step method", since the derivative in (3) are interpreted as the right-hand derivative. The solution on any interval $[k\tau, (k+1)\tau]$ is found by integrating (3) over the previous interval. The non-negativity constraint makes the argument non-trivial, see Appendix A for details.

For uniqueness, assume that x and $\tilde{x}$ are distinct functions, both satisfying (3) with the same initial state ϕ_0. Let $k \geq 0$ be the smallest integer such that $x(t) \neq \tilde{x}(t)$ somewhere in the half-open interval $k\tau < t \leq (k+1)\tau$. Then, the derivatives of x and $\tilde{x}$ must differ for some $s \in (k\tau, t)$. It follows that $x(s-\tau) \neq \tilde{x}(s-\tau)$, contradicting the minimality of k. □

The trajectory can also be uniquely extended backwards in time, as long as the trajectory is differentiable, non-zero, and with appropriately bounded derivative. If $q(t) > 0$ and $-c < \dot{q}(t) < q^*/\tau$, then

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

When $q(t) = 0$, however, the past trajectory is not uniquely determined.

4 Stability

The main result of the paper is the following stability result for the queue dynamics (3).

Theorem 2 *For any non-negative and right-hand continuous function ϕ_0 on $[-\tau, 0]$, the solution to* (3)*, with initial condition $q(t) = \phi_0(t)$ for $t \in [-\tau, 0]$, satisfies $q(t) \to q^*$ as $t \to \infty$.*

The proof of this result follows from the following three lemmas, which are proved in Appendix. First define S as the following bounded subset of the state space:

$$S = \{\phi \in C^1(-\tau, 0); -c \leq \dot{\phi}(\theta) \leq q^*/\tau \\ \phi(\theta) \geq 0, \phi(0) \leq 2q^*, \theta \in (-\tau, 0)\} \quad (5)$$

Lemma 1 (Invariance) *The set S is invariant under the differential equation* (3)*, i.e.,*

$$q_T \in S \implies q_t \in S, \quad t \geq T$$

Lemma 2 (Boundedness) *For any initial state ϕ_0, there exists a finite time T such that for all $t \geq T$, $q_t \in S$ and $q(t) > 0$*

Lemma 3 (Stability) *If $\phi_0 \in S$ and $q(t) > 0$ for all $t \geq 0$, then $q(t) \to q^*$ as $t \to \infty$.*

The proof of the theorem follows from that by Lemma 2, there exists a T such that $q_T \in S$ and $q(t) > 0$ for all $t \geq T$. By Lemma 3, the initial state q_T then results in a trajectory that converges to q^*.

5 Convergence rate

When using cascade control, it is essential that the inner loop is significantly faster than the outer loop. This section derives an upper bound for the time constant of the inner loop. It can be used to aid the design of the window update mechanism of the outer loop. The main result of the section can be summarized as, for arbitrary capacities and delays, transmission control based on the ACK-clock ensures that the sending rate and the bottleneck queue size converge within a small number of roundtrip times.

We concentrate on the convergence rate for small variations, using a linearization of the queue dynamics (3). Note that since the link serving the queue is of finite capacity, we can not expect to have exponential convergence for arbitrary large initial values. For $q(t)$ close to q^*, the denominator in (3) can be approximated by the stationary value for the roundtrip delay

$$\tau^* = \tau + q^*/c$$

This leads to the linearized dynamics

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

For a simple feedback system consisting of an integrator, a proportional gain, and a delay, the Nyquist criterion yields the stability condition that gain × delay $< \pi/2$. In our case, this gives that

$$(1/\tau^*)\tau = \tau/(\tau + q^*/c) < 1$$

Note that this stability condition is satisfied for any $\tau, q^* > 0$. Let

$$\gamma = \frac{\tau}{\tau^*} = \frac{\tau c}{\tau c + q^*}$$

so that $0 < \gamma < 1$. The poles of the closed-loop system are the solutions to the equation

$$se^{s\tau} + \frac{1}{\tau^*} = 0$$

Let $z = \tau s$, and rewrite the equation as

$$ze^z + \gamma = 0 \tag{6}$$

Since the system is stable, we know that $\operatorname{Re} z < 0$ for all solutions to (6). Our next goal is to find a bound λ such that all $\operatorname{Re} z \le -\lambda$ for all solutions. Then all transients of the system are bounded by the exponential $Ce^{-\lambda t/\tau}$, which corresponds to that the convergence time constant is at most τ/λ.

Assume that $z = -x + iy$, $x > 0$, is a pole of the system. Substitution into (6) yields

$$-x + iy = z = -\gamma e^{-z} = -\gamma e^x e^{iy} \tag{7}$$

First, we derive a bound for $|y|$ in terms of x. We have

$$|y| \le |z| = \gamma e^x \tag{8}$$

Assume that $x \le \log(\pi/(2\gamma))$, so that both left and right hand side of (8) lie in the interval $[0, \pi/2]$, where the cos function is decreasing. Now, consider the real part of Equation (7),

$$x = \gamma e^x \cos y = \gamma e^x \cos|y| \ge \gamma e^x \cos(\gamma e^x) \tag{9}$$

Define λ as the smallest positive solution to

$$x = \gamma e^x \cos(\gamma e^x) \tag{10}$$

To get a bound for x, note that (9) implies that $x \ge \lambda$, so we have the implication

$$x \le \log(\pi/(2\gamma)) \implies x \ge \lambda$$

It follows that $x \ge \min(\lambda, \log(\pi/(2\gamma)))$. However, we also have that $\lambda < \log(\pi/(2\gamma))$, since the right hand side of (10) is zero for $x = \log(\pi/(2\gamma))$. Hence, $x \ge \lambda$ for every pole. Figure 2 shows λ as a function of γ.

The bound on the real part of the poles, λ, translates to a lower bound τ/λ on the convergence time constant. From numerical evaluation, we have $\lambda(\gamma) > 0.298$ for all $\gamma > 0.24$, see Figure 2. This bounds is equivalent to the bound $q^*/c < 3.17\tau$ on the average queueing delay. With this bound on the average queue size, the convergence time constant is at most $3.35\ \tau$.

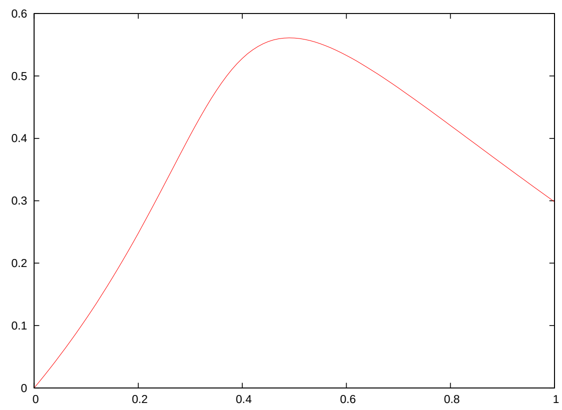

Figure 2: The bound λ, as a function of γ.

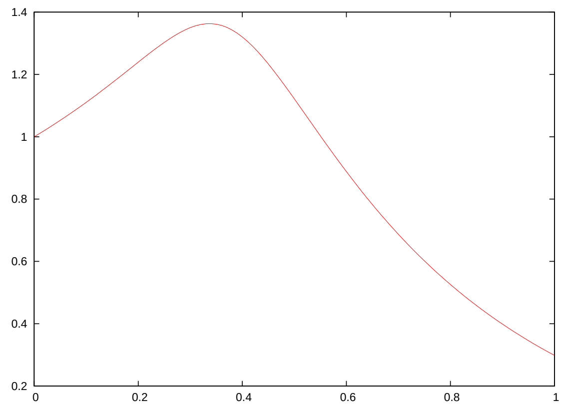

Figure 3: The bound μ, as a function of γ.

The time constant can also be expressed in terms of the average roundtrip time τ^*. Put $\mu = \lambda/\gamma$, then μ is the smallest positive solution to

$$\mu = e^{\gamma\mu} \cos(\gamma e^{\gamma\mu}) \tag{11}$$

and the time constant is bounded by τ^*/μ. Figure 3 shows μ as a function of γ. We see that $\mu > 0.298$ for all γ, which implies that that the convergence time constant is at most $3.35\ \tau^*$.

We summarize the results in the following theorem.

Theorem 3 *The linearized system corresponding to* (3) *is stable, and has a convergence time constant smaller than* τ/λ*, where* τ *is the roundtrip time excluding the queueing delay at the bottleneck,* *and λ is the smallest positive solution to*

$$x = \gamma e^x \cos(\gamma e^x) \qquad (12)$$

Furthermore, for all γ, the time constant is smaller than 3.35 τ^*. The same bound also holds if the total roundtrip time τ^* is replaced by τ, unless the queueing delay is extreme (more than 3.17 τ).

Due to the simplifications in the fluid model, these bounds need to be taken with a grain of salt. For large queueing delays, e.g., $\gamma = 0.3$, we get $\mu > 1$ and a time constant significantly smaller than the roundtrip time. We would probably get a larger time constant if we did not assume that ACKs carry information about the queue state at the time they leave the queue, as discussed in the beginning of Section 2.1. On the other hand, in the case of no cross-traffic, the sending rate and queue length converges exactly in one roundtrip time, since the ACK-clock forces the sending rate to equal the link capacity, with a single roundtrip delay and no interesting dynamics. So in this case the flow model turns out to be conservative.

For typical network paths, with a roundtrip time of at most a few 100 ms, we can expect convergence time on the order of 1 s. This can be compared to the convergence time of proposed outer loops, such as TCPwith AQM with a convergence time of 20–60 s [5], and FAST TCP with convergence times up to several minutes [7].

6 Mobile terminal download

In this section, we apply the previous results to a window control mechanism (i.e., transmission control outer loop) designed for download to a mobile terminal. For further details as well as background material, see [11, 12, 13].

The analysis of the preceding section shows that for a fixed window size w, the sending rate and the queue size will converge to stationary values. A key question for window-based transmission control is obviously what window size should we use. In general, the appropriate window size is related to the bandwidth–delay product. More precisely, the ideal window size is

$$w_{\text{ideal}} = \tau c_{\text{fair}} + q_{\text{ref}}$$

where c_{fair} is the flow's fair share of the capacity, and q_{ref} is a small stationary queue size. It is the job of the outer loop and the network state estimator to find a proper window size.

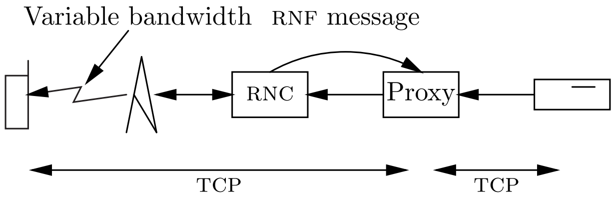

Figure 4: Radio network feedback architecture. The mobile terminal on the left downloads a file from the server on the right, via the proxy. During the transfer, the RNC generates RNF messages including information about the current bandwidth over the radio link, and the current RNC queue length. The proxy uses this information to adjust its window size.

One application where the implementation of an ideal window control is relatively easy , is for web browsing and file download in a mobile terminal. This system is illustrated in Figure 4. The wireless link is the bottleneck of the path, and the time varying capacity of this link is under the supervision of the radio network controller (RNC) associated with the base station. The RNC can be configured to notify the TCP sender, which is a web proxy within the operator's network, whenever the capacity of the radio link changes. This signalling is called radio network feedback (RNF).

To improve performance, the standard TCP window control in the proxy is replaced by a specialized controller that takes advantage of the RNF signalling. With estimates $\hat{c}$ and $\hat{\tau}$ of the capacity and delay, the TCP sender uses the following event-triggered feedforward control law for the window size: $w = \hat{\tau}\hat{c} + q_{\text{ref}}$. By applying Theorems 2 and 3, we have the following result.

Fact 1 *When the window size is set to*

$$w = \hat{\tau}\hat{c} + q_{ref}$$

the queue size converges to

$$q^* = q_{ref} + \hat{\tau}\hat{c} - \tau c$$

within a few roundtrip times.

We see that any error in the estimate of the bandwidth–delay product leads to a control error.

To compensate for this error, we use a feedback control mechanism. The RNC periodically sends RNF messages with the actual queue size. We use a sampling time significantly larger than the roundtrip time, e.g., one or a few seconds. The proxy controller uses this information in the time-triggered feedback control law:

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

We now argue that this controller removes the control error. Assume $w_0 = \hat{\tau}\hat{c} + q_{\text{ref}}$ was set by the feedforward controller. If the sampling time is large compared to the roundtrip time, then by Theorem 3, the sampling time is also a couple of times longer than the convergence time constant. We get

$$\begin{aligned} q_1 &\approx w_0 - \tau c && \text{Convergence} \\ w_1 &= w_0 + q_{\text{ref}} - q_1 \approx \tau c + q_{\text{ref}} && \text{Feedback} \\ q_2 &\approx w_1 - \tau c \approx q_{\text{ref}} && \text{Convergence} \end{aligned}$$

where the final line says that the queue has converged to the desired value. To summarize, the following fact holds.

Fact 2 *If the sampling time is large compared to the roundtrip time, then the feedback law $w_{k+1} = w_k + q_{ref} - q_{k+1}$ almost eliminates the bias after two sample periods.*

Compared to end-to-end TCP, the RNF controller improves both user response time, and radio link utilization. The changes are localized to the RNC and the proxy, which both reside in the operator's network. See [12, 13] for detailed evaluations of the proposed scheme.

7 Conclusions

In this paper, we investigate the stability properties of the ACK-clock, which is the inner loop in all window-based transmission control schemes. This inner loop is globally asymptotically stable. Furthermore, the convergence time constant for the corresponding linearized system is bounded by a small number of roundtrip times. Further analysis is needed to find out to which degree this stability result can be generalized to networks with multiple window-controlled sources and multiple bottlenecks. As an example outer loop, we consider a radio network feedback architecture, where explicit information about the bottleneck link is available to the outer loop window controller.

We would like to revisit design of window control and AQM mechanisms, leaving most of the queue stabilization work to the inner loop. It is argued that window control and AQM should not be designed for queue stabilization, since that is taken care of by the inner loop. They should be designed to meet the other objectives: Fairness, and adaptation to changes in the network state such as capacity, delay, and cross traffic intensity; these features all operate on a slower time scale than the per-packet time scale of the inner loop.

There are a couple of drawbacks when using feedback control of an open-loop stable system. The feedback can make the system unstable; we should design the outer loop controller so that the system either remains stable, or exhibits oscillations in queue sizes and other state variables that are reasonably small. Furthermore, the feedback will amplify disturbances, which can be expected to be a major concern for transmission control, where cross traffic induces stochastic queue variations that disturb the system and the measurements.

Let us end with a discussion of fairness and AQM. What information is needed in order to ensure fairness between TCP flows? It seems that we must have some information about what our fair share of the available bandwidth is (for some appropriate definition of fairness, e.g., proportional fairness). For example, compare the case of a few large cross-traffic flows, to the case of a large number of smaller flows. The fair share per flow is quite different in these two cases, hence, to select the right sending rate for our flow, we need to be able to distinguish between these cases. Estimating the fair share from the ACK timing alone seems non-trivial.

So how come that standard TCP with drop-tail queues exhibits any fairness? The reason is the packet loss signal. Bottleneck queues are full, and hence drop packets. And among all the flows that share a bottleneck link, the packet drops are distributed between flows in proportion to the size of the flows. Note that TCPs response, multiplicative decrease, is *not* proportional to the packet loss probability, which is equal for all flows, but rather it is related to the packet loss *frequency*.

Assume that some improved congestion control mechanism achieves the goal of maintaining small bottleneck queues, without packet losses. Then the packet loss signal, which is directly related to relative flow size, is no longer available, and the control must work with ACK timing alone. This makes fairness a more challenging problem. One motivation for the use of AQM is that it reintroduces a signal, the ECN marks, which is proportional to flow size.

References

[1] G. Appenzeller, I. Keslassy, and N. McKeown. Sizing router buffers. In *SIGCOMM*, Portland, September 2004. ASM.

[2] L. S. Brakmo and L. L. Peterson. TCP Vegas: end-to-end congestion avoidance on a global Internet. *IEEE Journal on Selected Areas in Communications*, 13(8):1465–1480, 1995.

[3] R. Bush and D. Meyer. Some internet architectural guidelines and philosophy. RFC 3439, December 2002.

[4] J. K. Hale and S. M. V. Lunel. *Introduction to functional differential equations*. Springer-Verlag, 1993.

[5] C. Hollot, V. Mistra, D. Towsley, and W. B. Gong. A control theoretic analysis of red. In *IEEE Infocom 2001*, 2001.

[6] V. Jacobson. Congestion avoidance and control. *ACM Computer Communication Review*, 18:314–329, 1988.

[7] C. Jin, D. X. Wei, and S. H. Low. FAST TCP: motivation, architecture, algorithms, performance. In *Proceedings of IEEE Infocom*. IEEE, March 2004.

[8] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan. Rate control in communication networks: shadow prices, proportional fairness and stability. *Journal of Operational Research Society*, 49:237–252, 1998.

[9] L. Le, J. Aikat, K. Jeffay, and F. D. Smith. The effects of active queue management on Web performance. In *SIGCOMM*, volume 33 of *Computer Communication Review*, pages 265–276, Karlsruhe, October 2003. ACM.

[10] Jim Martin, Arne Nilsson, and Injong Rhee. Delay-based congestion avoidance for tcp. *IEEE/ACM Trans. Netw.*, 11(3):356–369, 2003.

[11] I. Cabrera Molero, N. Möller, J. Petersson, Å. Arvidsson 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.

[12] N. Möller, I. Cabrera Molero, K. H. Johansson, J. Petersson, and Å. Arvidsson R. Skog. Using radio network feedback to improve TCP performance over cellular networks. In *IEEE Conference on Decision and Control and European Control Conference*, Seville, 2005. IEEE CSS.

[13] Niels Möller. Automatic control in TCP over wireless. KTH Licentiate Thesis, September 2005.

[14] S.-I. Niculescu. *Delay effects on stability*, volume 269 of *Lecture notes in control and information sciences*. Springer, 2001.

[15] M. Vidyasagar. *Nonlinear systems analysis*. Prentice-Hall, second edition, 1993.

[16] J. Wang, D. X. Wei, and S. H. Low. Modelling and stability of FAST TCP. In *Proceedings of IEEE Infocom*, Miami, March 2005. IEEE.

A Existence argument

Assume that f is continuous, and ϕ_0, defined on $[-1, 0]$, is non-negative and right-hand continuous.[1] Consider the equation

$$\begin{aligned} x(t) &= \phi_0(t) && \text{for } -1 \le t \le 0 && (13)\\ \dot{x}(t) &= f(x(t-1)) && \text{if } x(t) > 0 && (14)\\ \dot{x}(t) &= \max(0, f(x(t-1))) && \text{if } x(t) = 0 && (15) \end{aligned}$$

[1] In the proof, the continuity of ϕ_0 is not used in the construction of $x(t)$, only for proving that the constructed function actually is a solution. The results can be extended to an arbitrary non-negative, measurable and essentially bounded initial state ϕ_0, if the notion of a "solution" is generalized appropriately.

Equation (3) is of this form. The time derivatives should be interpreted as right-hand derivatives.

It is sufficient to show that there exists a solution for $0 \leq t \leq 1$, satisfying the same requirements as ϕ_0; then existence for all $t \geq 0$ follows by induction. Define

$$\psi(t) = \phi_0(0) + \int_0^t f(\phi_0(s-1))\mathrm{d}s$$

for $t \in [0,1]$. If ψ is non-negative, we can set $x(t) = \psi(t)$ for $t \in [0,1]$, but in the general case, we have to modify it via the following construction.

Let $c = \sup_s |f(\phi_0(s))|$, and define the set A as the set of all functions $z \in C(0,1)$ such that

1. $z(t) \geq 0$ for $t \in [0,1]$, with equality for $t = 0$.
2. $0 \leq s \leq t \leq 1 \implies z(s) \leq z(t)$.
3. $z(t) + \psi(t) \geq 0$ for $t \in [0,1]$.
4. $|z(s) - z(t)| \leq c|s-t|$ for all $s, t \in [0,1]$.

The set is non-empty, since $\int_0^t |f(\phi_0(s-1))|\mathrm{d}s \in A$, and the functions in A are bounded below by zero. Define

$$Z(t) = \inf_{z \in A} z(t)$$

Requirement 4 implies that $Z(t)$ is continuous. Define

$$x(t) = Z(t) + \psi(t)$$

Next, we verify that this $x(t)$ is a solution to the differential equation. Let $t \in [0,1]$ be arbitrary. There are two cases, depending on whether or not the non-negativity constraint is active or not at this point.

Active In this case $x(t) = 0$ and $\dot{\psi}(t) = f(\phi_0(t-1)) < 0$. It follows that for some interval $[t, t+\epsilon]$, ψ is decreasing. Then, on this interval, Z is increasing, and it follows from the minimality of Z that x is identical zero. Hence the right-hand derivative satisfies $\dot{x}(t) = 0$.

Not active In this case, either $x(t) > 0$ or $\dot{\psi}(t) = f(\phi_0(t-1)) \geq 0$. Then it follows from the continuity properties of x, f and ϕ_0, and the minimality of Z, that Z is constant in some interval $[t, t+\epsilon]$. Hence the right hand derivatives satisfy $\dot{Z}(t) = 0$ and $\dot{x}(t) = \dot{\psi}(t) = f(\phi_0(t-1))$.

In both cases, x satisfies the differential equation at the point t, and since t was arbitrary, it is a solution on the interval $[0,1]$, as required.

Note that the construction in this proof avoids enumerating the intervals where $x(t)$ is zero, since they may be infinitely many. For example, consider a solution of the form $x(t) = (\max(0, (t-1/2)^2 \sin(1/(t-1/2))))^2$, which is continuously differentiable, but is zero on infinitely many intervals accumulating at $t = 1/2$.

B Proof of Lemma 1

It is sufficient to show that $x_t \in S$ for $t \in [0,\tau]$; then the result follows by induction. The main idea of the proof is to use bounds on the derivative $\dot{q}$. For all $t \in [0,\tau]$ we have

$$\dot{q}(t) \leq \max\left(0, \frac{q^* - q(t-\tau)}{\tau + q(t-\tau)/c}\right) \leq \frac{q^*}{\tau} \qquad (16)$$

$$\dot{q}(t) \geq \frac{q^* - q(t-\tau)}{\tau + q(t-\tau)/c} \geq -c \qquad (17)$$

Hence, x_t satisfies the $\dot{\phi}$ bounds in (5) for all t. These bounds imply that

$$q(0) - tc \leq q(t) \leq q(0) + tq^*/\tau \qquad (18)$$

for all $t \in [0,\tau]$. Since $q_0 \in S$, similar inequalities hold also for $t \in [-\tau, 0]$; in this interval we get

$$q(0) + tq^*/\tau \leq q(t) \leq q(0) - tc \qquad (19)$$

Next, we show that $q(0) \leq 2q^*$ implies that $q(t) \leq 2q^*$ for $t \in [0,\tau]$. If $q(0) \leq q^*$, this follows directly from (18). So assume $q(0) = (1+\alpha)q^*$, with $0 < \alpha \leq 1$. The bound (19) implies that $q(t) > 0$ for $t \in [-\tau, 0]$. Then for $t \in [0,\tau]$, we have

$$\begin{aligned} q(t) &= q(0) + \int_0^t \dot{q}(s)\mathrm{d}s \\ &= (1+\alpha)q^* + \int_0^t \frac{q^* - q(s-\tau)}{\tau + q(s-\tau)/c}\mathrm{d}s \end{aligned} \qquad (20)$$

Using (19) again, $q(s-\tau) \geq q(0) + (s-\tau)q^*/\tau = (\alpha + s/\tau)q^*$. Inserting this expression into the numerator of the integrand, and replacing the integration interval by the subset where the new integrand is positive, we find

$$\begin{aligned} q(t) &\le (1+\alpha)q^* + q^* \int_0^t \frac{1-\alpha-s/\tau}{\tau + q(s-\tau)/c} \mathrm{d}s \\ &\le (1+\alpha)q^* + q^* \int_0^{(1-\alpha)\tau} \frac{1-\alpha-s/\tau}{\tau + q(s-\tau)/c} \mathrm{d}s \\ &\le (1+\alpha)q^* + q^* \int_0^{(1-\alpha)\tau} \frac{1-\alpha-s/\tau}{\tau} \mathrm{d}s \\ &= q^*(1+\alpha+(1-\alpha)^2/2) \\ &= q^*(3/2+\alpha^2/2) \le 2q^* \end{aligned} \tag{21}$$

This concludes the proof that the set S is invariant.

C Proof of Lemma 2

Like in the proof for the previous lemma, the $\dot{\phi}$ bounds in (5) are satisfied for all $t \ge 0$, so it remains to find a T such that $0 < q(t) \le 2q^*$ for all $t \ge T$. We handle the upper and lower limit separately, resulting in times T_u and T_l, and then $T = \max(T_u, T_l)$.

Upper bound: If $t \ge \tau$ and $q(t) \ge 2q^*$, then (16) implies that $q(t) \ge q^*$ for $t \in [t-\tau, t]$, and hence $\dot{q}(t) \le 0$. If $q(t) \ge 2q^*$ for *all* $t \ge \tau$, then $q(t)$ would be decreasing and bounded below by $2q^*$, hence converging to some limit no smaller than $2q^*$. But that is impossible, since q^* is the only stationary point.

Therefore there is some finite $T_u \ge \tau$ such that $q(T_u) \le 2q^*$. By Lemma 1, then $x_t \in S$ for all $t \ge T_u$.

Lower bound: Fix the initial condition ϕ_0. Then either $q(t) > 0$ for all $t \ge 0$, so we can take $T_l = 0$ and there is nothing to prove. Or there is some minimum $t_0 \ge 0$ such that $q(t_0) = 0$.

Then since $\dot{q}(t) \le q^*/\tau$, we have $0 \le q(t) \le q^*$ for $t \in [t_0, t_0+\tau]$. Assume for the moment that there are infinitely many points t_k, $k \ge 1$, such that $t_k > t_0$ and $q(t_k) = 0$. We can order the points so that $t_k < t_{k+1}$.

Since $q(t) - q^*$ changes sign at t_k, $\dot{q}(t)$ changes sign at $t_k + \tau$. It follows that the points are separated by at least τ. For all $k = 0, 1, 2, \ldots$, we have $t_{k+1} > t_k + \tau$ and

$$\begin{aligned} q(t) &< q^* \quad \text{for } t_{2n} < t < t_{2n+1},\ n = 0, 1, 2, \ldots \\ q(t) &> q^* \quad \text{for } t_{2n-1} < t < t_{2n},\ n = 1, 2, 3 \ldots \end{aligned}$$

This separation makes it straight-forward to bound the sequence of extreme values.

First consider the intervals $[t_{2n-1}, t_{2n}]$, $n = 1, 2, 3, \ldots$ where $q(t) \ge q^*$. The maximum is attained where $\dot{q}$ changes sign, at $q(t_{2n-1}+\tau)$. The bound (16) implies

$$q(t_{2n-1}+s-\tau) \ge q^* + (s-\tau)\frac{q^*}{\tau} = s\frac{q^*}{\tau} \tag{22}$$

and

$$\begin{aligned} q(t_{2n-1}+\tau) &= q^* + \int_0^\tau \dot{q}(t_{2n-1}+s)\mathrm{d}s \\ &= q^* + \int_0^\tau \frac{q^* - q(t_{2n-1}+s-\tau)}{\tau + q(t_{2n-1}+s-\tau)/c}\mathrm{d}s \\ &\le q^* + \int_0^\tau \frac{q^* - sq^*/\tau}{\tau}\mathrm{d}s \\ &= q^* + \frac{q^*}{\tau}\int_0^\tau (1-s/\tau)\mathrm{d}s \\ &= \frac{3}{2}q^* \end{aligned} \tag{23}$$

This shows that $q(t) \le 3q^*/2$ for all $t \ge t_0$. From this, we also get an improved lower bound. Consider the interval $[t_{2n}, t_{2n+1}]$, for $n = 1, 2, 3, \ldots$, where $q(t) \le q^*$. From (23) we get $q(t_{2n}+s-t) \le 3q^*/2$, and

$$\begin{aligned} q(t_{2n}+\tau) &= q^* + \int_0^\tau \dot{q}(t_{2n}+s)\mathrm{d}s \\ &= q^* + \int_0^\tau \frac{q^* - q(t_{2n}+s-\tau)}{\tau + q(t_{2n}+s-\tau)/c}\mathrm{d}s \\ &\ge q^* + \int_0^\tau \frac{q^* - 3q^*/2}{\tau + q(t_{2n}+s-\tau)/c}\mathrm{d}s \\ &= q^* - \frac{q^*}{2}\int_0^\tau \frac{1}{\tau + q(t_{2n}+s-\tau)/c}\mathrm{d}s \\ &\ge q^* - \frac{q^*}{2}\int_0^\tau \frac{1}{\tau}\mathrm{d}s \\ &= \frac{q^*}{2} \end{aligned} \tag{24}$$

It follows that $q(t) \in [q^*/2, 3q^*/2]$ for all $t \ge t_1$, and we can take $T_u = t_1$.

Finally, consider the case that there is no infinite sequence of points t_k where the trajectory intersects q^*. Then, let t_N be the final intersection (or $t_N = t_0$ if there are no intersections at all). Then for $t > t_N + \tau$, the trajectory is either non-decreasing and below q^*, or non-increasing and above q^*. In either case, take $T = t_N + \tau$.

D Proof of Lemma 3

We use the circle criterion to prove asymptotic stability under the additional assumption that the trajectory satisfies $0 < q(t) \leq 2q^*$, which is guaranteed by Lemma 2. We first make a change of variables

$$x(t) = (q(t\tau) - q^*)/(\tau c + q^*) \quad (25)$$

which transforms Equation (3) to

$$\dot{x}(t) = -\gamma \frac{x(t-1)}{1 + x(t-1)} \quad (26)$$

where $\gamma = \tau c/(\tau c + q^*) < 1$. The assumption $q(t) > 0$ translates to $x > \gamma - 1$ and the upper bound $q(t) \leq 2q^*$ translates to $x(t) \leq 1 - \gamma$.

Equation (26) can be decomposed as a linear system with a static non-linear feedback $\psi(x)$.

$$\dot{x}(t) = -\gamma x(t-1) - \gamma u(t) \quad (27)$$

$$u(t) = \psi(x(t-1)) \quad (28)$$

$$\psi(x) = -\frac{x^2}{1+x} \quad (29)$$

The transfer function for the linear system with $u(t)$ as input and $x(t-1)$ as output is

$$G(s) = \frac{\gamma}{\gamma + se^s}$$

which is obviously stable, cf., Section 5.

For $|x| \leq 1 - \gamma$, the non-linearity ψ satisfies the sector inequalities

$$\alpha x^2 \leq x\psi(x) \leq \beta x^2 \quad (30)$$

with

$$\alpha = -\frac{1-\gamma}{2-\gamma} \qquad \beta = \frac{1-\gamma}{\gamma} \quad (31)$$

Let $D(a, b)$ denote the circle in the complex plane, centered on the real axis, which intersects the real axis at a and b. According to the circle criterion, a sufficient condition for stability of the feedback system is that the Nyquist curve $G(i\omega)$ stays inside the circle $D(-1/\beta, -1/\alpha)$, with a positive margin [15]. In our case, the circle is

$$D(-\gamma/(1-\gamma), (2-\gamma)/(1-\gamma))$$

with center at 1 and radius $r = 1/(1-\gamma) > 1$.

To see that the Nyquist curve is inside this circle, with a positive margin, first consider the smaller circle $D(1-r, 1)$, which is tangent to the original circle at $1-r$, and tangent to the Nyquist curve at 1. Define a linear fractional transformation which maps $D(1-r, 1)$ onto the unit circle, and the origin onto itself:

$$h(z) = \frac{z}{(1-2\gamma)z + 2\gamma} \quad (32)$$

Compute the image of the Nyquist curve under h, $h(G(i\omega))$,

$$\begin{aligned} h(G(i\omega)) &= \frac{\gamma}{(1-2\gamma)\gamma + 2\gamma(\gamma + i\omega e^{i\omega})} \\ &= \frac{1}{1 + 2i\omega e^{i\omega}} \\ |h(G(i\omega))|^2 &= \frac{1}{|1 + 2i\omega e^{i\omega}|^2} \\ &= \frac{1}{1 + 4\omega^2 - 4\omega \sin \omega} \leq 1 \end{aligned}$$

where the final inequality follows from $|\sin \omega| \leq |\omega|$. We also see that the curve touches the unit circle only when $\omega = 0$. Hence, the Nyquist curve is inside $D(1-r, 1)$, getting close only at 1 (for $\omega = 0$), and it follows that the Nyquist curve is inside $D(1-r, 1+r)$ with a positive margin.