# Mathematical analysis of Westwood+ TCP congestion control

L.A. Grieco and S. Mascolo

**Abstract:** TCP congestion control is based on an additive-increase/multiplicative-decrease (AIMD) probing paradigm aimed at adapting the sending rate of TCP data sources to match the Internet time-varying available bandwidth. Westwood+ TCP has been recently proposed to improve the tracking of available bandwidth of classic TCP. It is based on an end-to-end estimate of the available bandwidth, which is obtained by properly counting and filtering the stream of acknowledgement packets. The estimate is used to adaptively decrease the congestion window and slow start threshold after congestion so that it can be said that Westwood+ TCP substitutes the classic multiplicative decrease with an adaptive decrease paradigm. The authors propose a mathematical analysis of the additive-increase/adaptive-decrease (AIADD) paradigm to analyse the steady-state throughput provided by Westwood+ TCP and investigate the intra-protocol fairness of the AIADD paradigm and the inter-protocol friendliness between AIADD and AIMD algorithms. It is shown that (i) both classic and Westwood+ TCP provide a throughput that is proportional to $1/\sqrt{p}$, where $p$ is the segment drop probability, that is they are friendly to each other; and (ii) the throughput of Westwood+ TCP is proportional to $1/\sqrt{RTT}$, where $RTT$ is the round trip time, whereas the throughput of Reno TCP is proportional to $1/RTT$, i.e. Westwood+ TCP improves the intra-protocol fairness. Finally, N*s*-2 simulations are reported in order to validate the mathematical model in the presence of a wide range of network loads, loss probabilities and round trip times.

## 1 Introduction

Classic TCP congestion control, i.e. Tahoe [1], Reno [2], New Reno [3], implements a self-clocked additive increase multiplicative decrease (AIMD) sliding window algorithm to probe the network available bandwidth [2–8]. Two variables: (i) the congestion window (*cwnd*); and (ii) the slow start threshold (*ssthresh*), are used to throttle the TCP input rate in order to match the network's available bandwidth. In particular, TCP congestion control gradually increases the *cwnd* until a congestion episode occurs. The policy of increasing *cwnd* depends on the *ssthresh* value. In particular, when $cwnd < ssthresh$, the *cwnd* is incremented by one segment for every received acknowledgment, thus obtaining an exponential growth of the *cwnd* over the time. This is the slow start phase that aims at quickly grabbing the network's available bandwidth. On the other hand, when $cwnd \geq ssthresh$, the congestion avoidance phase is entered during which the *cwnd* is linearly incremented over time by one segment per round trip time (*RTT*). The aim of the congestion avoidance phase is to gently probe the network for extra available bandwidth.

When three three-duplicate acknowledgments (three DUPACKs) are received *ssthresh* is set to $cwnd/2$, *cwnd* is set equal to $(ssthresh + 3)$ and the first unacknowledged segment which sits in the retransmit queue is retransmitted. Moreover, during this phase, every time a DUPACK is received, *cwnd* is increased by one segment to take into account that another segment has left the network [2]. When an ACK finally acknowledges new data, *cwnd* is set to *ssthresh*, thus restarting the sender in the congestion avoidance phase.

When the retransmit timeout expires *ssthresh* is set to $cwnd/2$, *cwnd* is set equal to unity and the segment that appears to be lost is retransmitted. The *ssthresh* represents an estimate of the current available bandwidth. In fact, when the *cwnd* is less than *ssthresh*, the congestion control algorithm increases *cwnd* to quickly reach the *ssthresh* value and when $cwnd = ssthresh$, it increments the *cwnd* in a more gentle way in order to probe for extra available bandwidth.

Whereas AIMD algorithms can ensure that network capacity is not exceeded, they cannot ensure fair sharing of that capacity [1, 9, 10]. Moreover they provide a low throughput in the presence of losses not due to congestion, such as in the case of radio links, because they interpret losses caused by unreliable links as a symptom of congestion, thus reducing the transmission rate even if it is not due [11–13].

In order to improve the capability of classic TCP to track the available bandwidth, the recently proposed Westwood TCP congestion control substitutes the multiplicative decrease with an adaptive decrease phase, so that it can be said that Westwood TCP follows an additive increase adaptive decrease (AIADD) paradigm. In particular, Westwood TCP filters the stream of returning (ACKs). To obtain an estimate of the used bandwidth (*BWE*) so that when the network capacity is hit and a congestion episode occurs, the Westwood technique adaptively reduces the congestion window and the slow-start threshold to a value of *BWE* times the minimum RTT $RTT_{\min}$ [13].

In the case of wired networks, with losses due to congestion, the adaptive decrease phase allows Westwood

TCP flows to deplete connection buffers, which ensures a prompt congestion recovery and enhances statistical multiplexing and fairness. On the other hand, when losses are due to unreliable radio links and not to congestion, Westwood TCP does not lose ground because its setting takes into account the bandwidth used at the time of congestion.

Westwood TCP is heavily based on bandwidth estimation. The original bandwidth estimation algorithm proposed in [14] does not work properly in the presence of ACK compression [15, 16]. Westwood+ TCP differs from Westwood TCP in that it employs a new bandwidth estimation algorithm that works properly also in the presence of ACK compression [16–19].

We intend to propose a mathematical analysis of the AIADD paradigm. In particular, we intend to analyse the steady-state throughput of a Westwood+ TCP connection in order to investigate the intra-protocol fairness of the AIADD paradigm, i.e. how Westwood+ TCP flows share the network's available bandwidth, and inter-protocol friendliness, i.e. how flows controlled by an AIMD algorithm share the bandwidth with flows controlled by a AIADD algorithm.

Finally, Ns-2 simulations [20] will be used to validate the theoretical results in the presence of a wide range of network loads, loss probabilities and RTTs.

## 2 Westwood+ TCP background

The key idea of TCP Westwood is to exploit the stream of returning acknowledgment packets to estimate the bandwidth *BWE* that is available for the TCP connection. When a congestion episode occurs at the end of the TCP probing phase, the used bandwidth corresponds to the definition of the best effort available bandwidth in a connectionless packet network. The bandwidth estimate is used to adaptively decrease the congestion window and the slow-start threshold after a timeout or three duplicate ACKs. In particular, when, three DUPACKs are received, the bandwidth estimate is used to set both the *cwnd* and the *ssthresh* control window values equal to $\max[2, BWE \times RTT_{\min}]$. On the other hand, when a timeout expires, the *cwnd* is set equal to unity whereas the *ssthresh* is set as in the previous case. The pseudo-code below summarises the Westwood TCP algorithm:

When three DUPACKs are received:

```
ssthresh = (BWE * RTTmin)/MSS;
  //adaptive decrease phase
if (ssthresh < 2) ssthresh = 2;
cwnd = ssthresh;
```

When coarse timeout expires:

```
ssthresh = (BWE * RTTmin)/MSS;
  //adaptive decrease phase
if (ssthresh < 2) ssthresh = 2;
cwnd = 1;
```

When ACKs are successfully received:

```
cwnd increases as stated in the
  New Reno algorithm.
```

The bandwidth estimation algorithm originally proposed with Westwood TCP does not work properly in the presence of ACK compression, which is an important phenomena when the ACK path is congested, because bandwidth samples are collected each time an ACK is received by the TCP sender [14, 17]. In order to overcome this limitation, Westwood+ TCP employs a bandwidth estimation algorithm that is able to works properly in the presence of ACK compression. In particular, to estimate the available bandwidth, Westwood+ TCP counts the amount of data $D_k$ acknowledged during the last RTT and computes the bandwidth sample $B_k = D_k/RTT$. Since buffers located at router interfaces encountered along the path act like integrators, only low-frequency components of the sending rate can provoke an overflow [21]. This assumption is valid when buffers are wide enough to absorb the burstiness of the TCP senders. For this reason, the bandwidth samples $B_k$ are filtered using a discrete time-varying low-pass filter, which is obtained by discretising a continuous-time single-pole filter by using a Tustin approximation. This results in a bandwidth estimate as of:

$$\hat{B}_k = \frac{2\tau - \Delta_k}{2\tau + \Delta_k}\hat{B}_{k-1} + \Delta_k \frac{B_k + B_{k-1}}{2\tau + \Delta_k}$$

where $1/\tau$ is the filter cut-off frequency (a typical value is $\tau = 0.5$ s), and $\Delta_k$ is the last RTT. To conclude, we say a few words on the sampling interval $RTT = \Delta_k$. In order to obtain a correctly working filter, it is necessary to satisfy the Nyquist sampling theorem, that is, $\Delta_k \leq \tau_f/2$ [19]. To be conservative, we assume that $\Delta_k \leq \tau_f/4$. Thus, if it happens that $\Delta_k > \tau_f/4$, then we interpolate and re-sample by creating $N = \mathrm{int}(4\, RTT/\tau_f)$, where $\mathrm{int}(\cdot)$ stands for the integer part of $(\cdot)$, virtual samples $b_k = D_k/\Delta_k$ that arrive with an interarrival time of $\Delta_k = \tau_f/4$ [15, 18].

To give an insight into the bandwidth estimation algorithm, Figs. 1 and 2 show the bandwidth estimate computed by one Westwood+ or one Westwood flow sending data over a 5 Mb/s bottleneck in the presence of reverse traffic contributed by ten TCP long-lived New Reno connections. Figure 1 shows that the Westwood+ connection estimates a best-effort available bandwidth that reasonably approaches the available bandwidth of 5 Mb/s. On the other hand, Fig. 2 highlights that Westwood

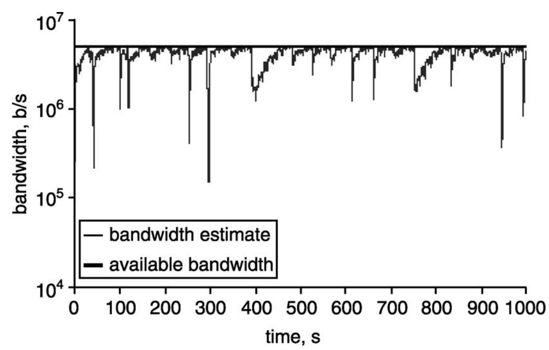


**Fig. 1** *Bandwidth estimate of one Westwood+ TCP flow in the presence of ACK compression*

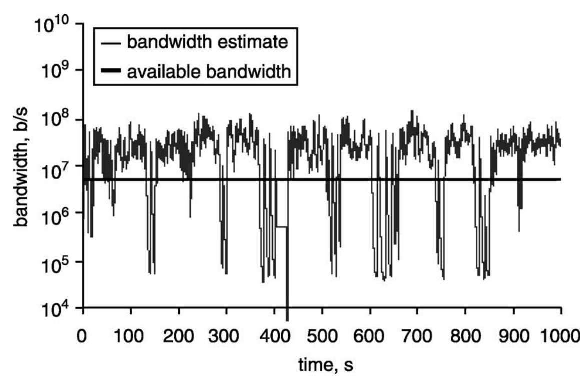


**Fig. 2** *Bandwidth estimate of one Westwood TCP flow in the presence of ACK compression*

overestimates by up to ten times the available bandwidth due to ACK compression.

It is worth noting that the adaptive decrease mechanism employed by Westwood+ TCP improves the standard TCP multiplicative decrease algorithm. In fact, the adaptive window shrinking provides a congestion window that is sufficiently decreased in the presence of heavy congestion and not excessively so in the presence of light congestion or losses that are not due to congestion, such as in the case of unreliable radio links. Moreover, the adaptive setting of the control windows increases the fair allocation of available bandwidth to different TCP flows. In fact, it could be noted that the setting $cwnd = BWE \times RTT_{\min}$ sustains a transmission rate $cwnd/RTT = BWE \times RTT_{\min}/RTT$ that is less than the bandwidth $BWE$ that is measured at the time of congestion: as a consequence, the TCP flow clears out its path backlog after the setting thus leaving room in the buffers for coexisting and joining flows, which improves statistical multiplexing and fairness. On the other hand, the setting $cwnd = BWE \times RTT$ would produce an input rate exactly equal to the one that has caused the congestion; thus, it would continue to provoke congestion by increasing backlog and causing instability.

## 3 A mathematical model for the AIADD paradigm

In this Section, we model the dynamics of the expected transmission rate of an AIADD controlled TCP flow as a function of the segment loss probability, the bandwidth estimate and the connection RTT. In order to obtain the dynamic model of the expected transmission rate of an AIADD controlled TCP flow we follow an approach similar to the one proposed in [9]. We will not model the propagation and queuing delays because we will focus on the steady-state behaviour of Westwood+ TCP. For the same reason, we will consider the connection RTT and the packet loss probability $p$ to be constant. We will validate the model using Ns-2 simulations [17] and we will highlight the conditions under which the proposed analysis provides valid results.

Three different models of the steady-state throughput have been obtained by assuming different approximations of the bandwidth estimate used by Westwood+. The analysis will be initially carried out without considering any upper bound on the $cwnd$ size, then, this assumption will be relaxed, and the behaviour of Westwood+ will be modelled also in the presence of an upper bound $W_{\max}$ on the $cwnd$.

### 3.1 Dynamics of the AIADD paradigm

Let us consider a Westwood+ TCP flow and assume that: (i) the drop probability of a segment is $p$; (ii) the segment losses are independent; (iii) the bandwidth estimate is $\hat{B}$; (iv) the mean round trip time is $RTT$; and the minimum round trip time is $RTT_{\min}$.

The congestion window is updated upon ACK reception. The receiver transmits one ACK packet for each data segment received. Moreover, each time the sender receives an ACK packet, it increases the $cwnd$ by $1/cwnd$. Therefore, each time a segment is successfully received by the receiver, the $cwnd$ is increased by $1/cwnd$, whereas, after a segment loss, the congestion window is set equal to $\hat{B} \times RTT_{\min}$ so that the change in $cwnd$ is $\hat{B} \times RTT_{\min} - cwnd$. Thus, the variation of the congestion window $\Delta cwnd$ is a discrete random variable with the following probability function:

$$p(\Delta cwnd) = \begin{cases} 1-p & \text{when } \Delta cwnd = 1/cwnd \\ p & \text{when } \Delta cwnd = \hat{B} \times RTT_{\min} - cwnd \end{cases} \quad (1)$$

The expected increment of the congestion window $cwnd$ per update step is then:

$$E[\Delta cwnd] = \frac{1-p}{cwnd} + (\hat{B} \times RTT_{\min} - cwnd)p \quad (2)$$

Since the time between update steps is $RTT/cwnd$ and the average transmission rate is $r = cwnd/RTT$, by recalling (2), the expected change in the rate $r$ per unit time is approximately given by:

$$\frac{dr(t)}{dt} = \frac{1-p}{RTT^2} + pr(t)\frac{\hat{B} \times RTT_{\min}}{RTT} - r^2(t)p \quad (3)$$

### 3.2 Steady-state behaviour of the AIADD paradigm

This Section analyses the equilibrium point of (3), i.e. the steady-state values of the transmission rate ($\dot{r}(t) = 0$). We will show that for any constant value of $p$ and $\hat{B}$, (3) has an equilibrium point that is exponentially globally stable. Exponential stability means that the system converges to the equilibrium point faster that an exponential function. Global stability means that the system converges to the equilibrium point when starting from any initial condition.

*Theorem 1:* For any constant value of the loss probability $p$ and available bandwidth $\hat{B}$ the steady-state throughput $r^{\text{West}}$ achieved by a Westwood+ flow has the following equilibrium point:

$$r^{\text{West}} = \frac{\hat{B} \times RTT_{\min}}{2\,RTT} + \sqrt{\left(\frac{\hat{B} \times RTT_{\min}}{2\,RTT}\right)^2 + \frac{1-p}{p(RTT)^2}} \quad (4)$$

which is exponentially globally stable.

*Proof:* By considering $p$ and $\hat{B}$ as constant inputs, the equilibrium point (4) can be easily found by imposing $dr/dt = 0$ in (3). To show that the equilibrium point (4) is globally exponentially stable, we show that the following autonomous system:

$$\frac{dR(t)}{dt} = pR(t)\frac{\hat{B} \times RTT_{\min}}{RTT} - R^2(t)p - 2R(t)r^{\text{West}}p \quad (5)$$

which is obtained using the change of coordinate $r = R + r^{\text{West}}$ in (3), has a globally exponentially stable equilibrium point in $R = 0$.

Equation (5) is a separable variable differential equation. By separating variables, it can be written as:

$$\frac{dR(t)}{R^2(t) + R(t)a} = -pdt \quad (6)$$

where, accordingly to (4):

$$a = 2\,r^{\text{West}} - \frac{\hat{B} \times RTT_{\min}}{RTT} \geq 0$$

By integrating each member of (6) the following solution can be obtained:

$$R(t) = R(0)e^{-apt}/(1 + R(0)/a(1 - e^{-apt}))$$

It is easy to see that the following inequality holds:

$$R(t) \leq R(0)e^{-apt}$$

which means that the equilibrium point (4) is globally exponentially stable [20]. □

### 3.3 A comparison of fairness and friendliness of AIADD and AIMD

Equation (4) models the throughput as a function of the bandwidth estimate $\hat{B}$. However, it does not take into account that the bandwidth estimate is a function of the Westwood+ sending rate and of the segment loss probability. In particular, when the loss probability $p \ll 1$ and the bandwidth estimation algorithm is fast enough with respect to the network dynamics, then, the following approximation holds $\hat{B} \cong cwnd/RTT$, because of the flow conservation principle. Now, by assuming a constant packet drop rate we are ready to demonstrate the following theorem.

*Theorem 2:* By assuming a loss probability $p \ll 1$, and the approximation $\hat{B}(t) = cwnd/RTT = r(t)$, the steady-state throughput $r^{\text{West}}$ achieved by a Westwood+ flow, has the following globally exponentially stable equilibrium point:

$$r^{\text{West}} = \frac{1}{\sqrt{RTT \times T_{\text{q}}}} \sqrt{\frac{1-p}{p}} \qquad (7)$$

where $T_{\text{q}} = RTT - RTT_{\min}$ is the mean queuing time experienced by the segments of the connection.

*Proof:* By assuming the following estimate of the available bandwidth:

$$\hat{B}(t) = cwnd/RTT = r(t) \qquad (8)$$

and by substituting (8) into (3), the following differential equation is obtained:

$$\frac{dr(t)}{dt} = \frac{1-p}{RTT^2} + p\, r^2(t) \frac{RTT_{\min}}{RTT} - r^2(t)p \qquad (9)$$

The condition $dr/dt = 0$ in (9) provides the equilibrium point. To show that the equilibrium point (7) is globally exponentially stable, we show that the following autonomous system:

$$\frac{dR(t)}{dt} = -p\left(1 - \frac{RTT_{\min}}{RTT}\right)(R^2(t) + 2\, r^{\text{West}})$$
$$= -p\frac{T_{\text{q}}}{RTT}(R^2(t) + 2\, r^{\text{West}}) \qquad (10)$$

which is obtained using the change of coordinate $r = R + r^{\text{West}}$ in (9), has a globally exponentially stable equilibrium point in $R = 0$.

By separating variables, (10) can be written as:

$$\frac{dR(t)}{R^2(t) + 2r^{\text{West}}R(t)} = -p\frac{T_{\text{q}}}{RTT}dt$$

and by integrating each member the solution is:

$$R(t) = R(0)\exp\left(-2r^{\text{West}} \times p \cdot \frac{T_{\text{q}}}{RTT}t\right) \Big/ \left(1 + \frac{R(0)}{2r^{\text{West}}}\left[1 - \exp\left(-2r^{\text{West}} \times p\frac{T_{\text{q}}}{RTT}t\right)\right]\right)$$

It easy to prove that the following inequality holds:

$$R(t) \leq R(0)\exp\left(-2r^{\text{West}} \times p\frac{T_{\text{q}}}{RTT}t\right) \qquad (11)$$

which means that the equilibrium point (7) is globally exponentially stable [23]. □

#### 3.3.1 Analytical comparison of Reno and Westwood+:

By assuming $p \ll 1$, the Reno throughput has been approximated as follows in [9] and [10]

$$r^{\text{Reno}} \propto \frac{1}{RTT\sqrt{p}} \qquad (12)$$

By comparing (7) and (12) it can be noticed that both the throughputs of Westwood+ and Reno depend on $1/\sqrt{p}$, that is they are friendly towards each other. Moreover, from (7) it can be noticed that flows with different $RTT$ values experience the same mean queuing time $T_{\text{q}}$. Therefore, the throughput of Westwood+ depends on the RTT as $1/\sqrt{RTT}$ whereas the throughput of Reno varies as $1/RTT$. Thus, with reference to fairness it can be concluded that Westwood+ increases the fair sharing of network capacity between flows with different $RTT$ values.

### 3.4 A dynamic model for the bandwidth estimate of Westwood+

In the proceeding Section, we used the approximation $\hat{B} = cwnd/RTT = r(t)$ for the bandwidth estimate; however, this approximation only works if: (i) the filter is fast enough with respect to the system dynamics; and (ii) the segment drop probability $p$ is very small.

This Section models the dynamics of Westwood+ TCP and analyses the equilibrium stability of the AIADD paradigm by taking into account the dynamics of the bandwidth estimate and relaxing the assumption of a low loss rate made in the preceeding section. Filtering is useful to obtain the low-frequency components of the available bandwidth but it also affects the dynamics of the transmission rate $r(t)$. In our model, we consider that Westwood+ employs a first-order low-pass filter with a time constant of $\tau$. To take into account the loss probability, we consider that the low-pass filter receives the rate $[1 - p(r(t))]r(t)$ of ACK packets. Therefore, the bandwidth estimate is the convolution of $[1 - p(r(t))]r(t)$ and $h(t)$, where $h(t) = e^{-t/\tau}/\tau$ is the impulse response of a first-order low-pass filter:

$$\hat{B}(t) = \{[1 - p(r(t))]r(t)\} \times h(t)$$
$$= \int_0^t [1 - p(r(\vartheta))]r(\vartheta)h(t-\vartheta)d\vartheta \qquad (13)$$

By noticing that $p$ is a function of the sending rate $r(t)$, (13) can be rewritten as follows:

$$\frac{d\hat{B}(t)}{dt} = -\frac{\hat{B}(t)}{\tau} + \frac{1-p(r)}{\tau}r(t) \qquad (14)$$

We are ready to show the following result:

*Theorem 3:* By assuming the bandwidth estimate dynamics (14) and letting $r^{\text{West}}$ be the steady-state throughput of a Westwood+ flow,

$$\frac{\partial p}{\partial r_{|r=r^{\text{West}}}} \geq 0$$

(3) has the following asymptotically stable equilibrium point under the assumption

$$r^{\text{West}} = \frac{1}{\sqrt{RTT[RTT - (1-p)RTT_{\min}]}}\sqrt{\frac{(1-p)}{p}} \qquad (15)$$

where $p$ is the loss probability obtained for $r = r^{\text{West}}$.

*Proof:* By considering (3) and (14), the following state equations are obtained:

$$\begin{cases} \dfrac{dr(t)}{dt} = f(r, \hat{B}, p) = \dfrac{1 - p(r)}{RTT^2} + p(r) r(t) \hat{B}(t) \cdot \dfrac{RTT_{\min}}{RTT} - p(r) r^2(t) & (16a) \\ \dfrac{d\hat{B}(t)}{dt} = g(r, \hat{B}, p) = -\dfrac{\hat{B}(t)}{\tau} + \dfrac{1 - p(r)}{\tau} r(t) & (16b) \end{cases}$$

By considering $p = p_0 =$ constant and by imposing the equilibrium conditions $dr/dt = 0$ and $d\hat{B}/dt = 0$, it is easy to derive the equilibrium point $\bar{Q} = (\bar{r}, \bar{B})$ as follows:

$$\begin{cases} \bar{r} = r^{\text{West}} \\ \bar{B} = (1 - p_0) r^{\text{West}} \end{cases} \tag{17}$$

By linearising (16) around the equilibrium point $\bar{Q} = (\bar{r}, \bar{B})$ and by considering the change of variables $\Delta r(t)$, $\Delta\hat{B}(t)$ and $\Delta p(\Delta r)$ with respect to the equilibrium point we get:

$$\begin{cases} \dfrac{d\Delta r(t)}{dt} = a\Delta r(t) + b\Delta\hat{B}(t) + cm\Delta r(t) \\ \qquad = (a + cm)\Delta r(t) + b\Delta\hat{B}(t) \\ \dfrac{d\Delta\hat{B}(t)}{dt} = d\Delta r(t) + e\Delta\hat{B}(t) + lm\Delta r(t) \\ \qquad = (d + l \cdot m)\Delta r(t) + e\Delta\hat{B}(t) \end{cases} \tag{18}$$

where

$$\begin{aligned} a &= \frac{\partial f}{\partial r}_{|\bar{Q}} = r^{\text{West}} p_0 \left[ (1 - p_0) \frac{RTT_{\min}}{RTT} - 2 \right] \\ b &= \frac{\partial f}{\partial \hat{B}}_{|\bar{Q}} = r^{\text{West}} p_0 \frac{RTT_{\min}}{RTT} \\ c &= \frac{\partial f}{\partial p}_{|\bar{Q}} = -\frac{1}{RTT^2} + (r^{\text{West}})^2 \left[ (1 - p_0) \frac{RTT_{\min}}{RTT} - 1 \right] \\ d &= \frac{\partial g}{\partial r}_{|\bar{Q}} = \frac{1 - p_0}{\tau} \\ e &= \frac{\partial g}{\partial \hat{B}}_{|\bar{Q}} = -\frac{1}{\tau} \\ l &= \frac{\partial g}{\partial p}_{|\bar{Q}} = -\frac{r^{\text{West}}}{\tau} \\ m &= \frac{\partial p}{\partial r}_{|\bar{Q}} \end{aligned} \tag{19}$$

Equations (18) can be rewritten in compact form using the state matrix:

$$\boldsymbol{F} = \begin{bmatrix} a + cm & b \\ d + lm & e \end{bmatrix}$$

as follows:

$$\begin{bmatrix} \dfrac{d\Delta r(t)}{dt} \\ \dfrac{d\Delta\hat{B}(t)}{dt} \end{bmatrix} = \boldsymbol{F} \begin{bmatrix} \Delta r(t) \\ \Delta\hat{B}(t) \end{bmatrix}$$

The eigenvalues of $\boldsymbol{F}$ are:

$$\lambda_{1,2} = \frac{a + e + cm \pm \sqrt{(a + e + cm)^2 + 4b(d + lm) - 4e(a + cm)}}{2} \tag{20}$$

By considering that $RTT_{\min}/RTT \le 1$, $p \le 1$ and consequently:

$$\left[ (1 - p_0) \cdot \frac{RTT_{\min}}{RTT} - 1 \right] \le 0$$

then (19) becomes:

$$\begin{aligned} a + cm + e &= r^{\text{West}} p_0 \left[ (1 - p_0) \frac{RTT_{\min}}{RTT} - 2 \right] - \frac{1}{\tau} + (r^{\text{West}})^2 \\ &\quad \left[ (1 - p_0) \frac{RTT_{\min}}{RTT} - 1 \right] m - \frac{m}{RTT^2} < 0 \quad \text{for } m \ge 0 \end{aligned} \tag{21}$$

and

$$\begin{aligned} & b(d + lm) - e(a + cm) \\ &= bd - ea + m(bl - ec) \\ &= \frac{2}{\tau} r^{\text{West}} p_0 \left[ (1 - p_0) \frac{RTT_{\min}}{RTT} - 1 \right] \\ &\quad + \frac{m}{\tau} \left\{ -(r^{\text{West}})^2 p_0 \frac{RTT_{\min}}{RTT} - \frac{1}{RTT^2} + (r^{\text{West}})^2 \right. \\ &\quad \left. \times \left[ (1 - p_0) \frac{RTT_{\min}}{RTT} - 1 \right] \right\} < 0 \quad \text{for } m \ge 0 \end{aligned} \tag{22}$$

Therefore, from (20), (21) and (22) it turns out that:

$$\text{Re}(\lambda_{1,2}) < 0 \quad \text{for } m \ge 0 \tag{23}$$

From known results of nonlinear control theory, (23) implies that the equilibrium point $\bar{Q}$ is locally asymptotically stable for $m \ge 0$ [23]. □

### 3.5 Modelling the impact of window limitation

So far we have assumed no bound on the *cwnd* size. In order to provide a more complete analysis of the AIADD paradigm, this Section models the behaviour of Westwood+ under the assumption that a maximum bound $W_{\max}$ on the *cwnd* does exist. For that purpose, we assume that during the data transfer the *cwnd* reaches its maximum value so that the available bandwidth estimated at the time a loss is experienced is equal to $\hat{B} = W_{\max}/RTT$. Thus, the *cwnd* is set to be equal to:

$$cwnd = \hat{B} \times RTT_{\min} = \frac{W_{\max}}{RTT} RTT_{\min}$$

after a loss is experienced. Under this assumption we obtain the time-domain behaviour for *cwnd* pictured in Fig. 3.

Let us adopt the following notation:

$S_1$: number of segment sent during the time interval $\Delta T_1$
$S_2$: number of segment sent during the time interval $\Delta T_2$

From Fig. 3, for $0 \le t \le \Delta T_1 + \Delta T_2$, it follows, that:

$$cwnd(t) = \begin{cases} \frac{W_{\max} \times RTT_{\min}}{RTT} + \frac{t}{RTT} & \text{if } cwnd(t) \le W_{\max} \\ W_{\max} & \text{otherwise} \end{cases} \tag{24}$$

From (24) we can compute $\Delta T_1$ if we use the relation:

$$\frac{W_{\max} \times RTT_{\min}}{RTT} + \frac{\Delta T_1}{RTT} = W_{\max}$$

then we can obtain:

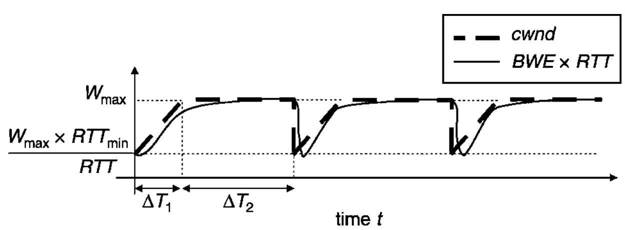


**Fig. 3** *Congestion window time behaviour in the presence of an upper bound on cwnd*

$$\Delta T_1 = W_{\max}(RTT - RTT_{\min}) \tag{25}$$

By considering that $r = cwnd/RTT$ is the connection input rate, the number of segments sent during the time interval $\Delta T_1$ is given by the integral:

$$S_1 = \int_0^{\Delta T_1} r(\tau)d\tau$$

Then from Fig. 3 it holds that:

$$S_1 = \frac{W_{\max}^2}{2RTT^2}\left(RTT^2 - RTT_{\min}^2\right) \tag{26}$$

Following the same arguments used to compute $S_1$, the number of segment sent during the time interval $\Delta T_2$ is:

$$S_2 = \frac{W_{\max}}{RTT}\Delta T_2 \tag{27}$$

By considering that a segment drop probability $p$ means that a segment is dropped approximately after that $1/p$ segments have been sent, it follows that:

$$\frac{1}{p} = S_1 + S_2 = \frac{W_{\max}^2}{2RTT^2}\left(RTT^2 - RTT_{\min}^2\right) + \frac{W_{\max}}{RTT}\Delta T_2 \tag{28}$$

By solving (28) with respect to $\Delta T_2$ we obtain:

$$\Delta T_2 = \frac{RTT}{pW_{\max}} - \frac{W_{\max}}{2RTT}\left(RTT^2 - RTT_{\min}^2\right) \tag{29}$$

By considering (26), (27) and (29), the mean throughput of Westwood+ TCP can be computed as:

$$r^{\mathrm{West}} = \frac{S_1 + S_2}{\Delta T_1 + \Delta T_2} = \frac{1}{p(\Delta T_1 + \Delta T_2)}$$
$$= \frac{(W_{\max}/RTT)}{1 + (pW_{\max}^2/2)(1 - (RTT_{\min}/RTT))^2} \tag{30}$$

## 4 Simulation results and model validation

This Section validates the models of first-in (4), (7) and (15) using the Ns-2 network simulator [20, 24]. For that purpose, a network topology with a single first-in first-out bottleneck shared by $N$ Westwood+ TCP greedy flows with different $RTT$ values has been used (see Fig. 4). In particular, we consider $RTT$ values ranging uniformly from $[2 + (250/N)]$ ms to 252 ms, and the number of connections sharing the bottleneck, $N$, ranges from ten to 80. Simulations last 500 s. The bottleneck bandwidth is set equal to 10 Mb/s and the bottleneck buffer size is set equal to the bottleneck bandwidth times the maximum $RTT$ value so that it is equal to 210 segments. Segments are 1500 bytes long. Westwood+ TCP implements the New Reno feature, which avoids multiple *cwnd* settings when several or more segments from the same window of data are lost [3]. On the backward path ten Reno TCP connections send data in order to provoke ACK compression. We have validated the models (4), (7) and (15) when the backward traffic is off and on, respectively.

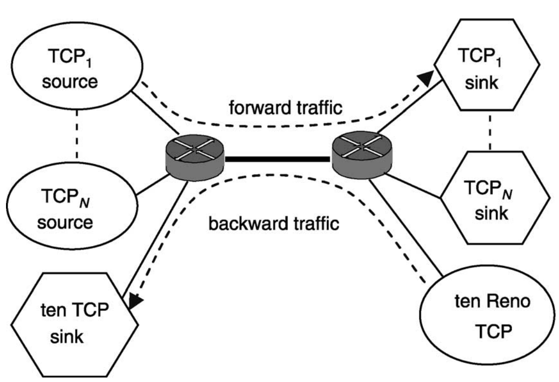


**Fig. 4** *Single bottleneck topology used for model validation*

In order to validate the models of (4), (7) and (15), we have measured for each Westwood+ TCP flow, the following variables: (i) the drop probability $p$ computed as the ratio between the number of retransmitted segment and the total number of sent segments; (ii) the mean $RTT$ value; and (iii) the mean minimum $RTT_{\min}$. To validate (4) we have also computed the mean value of the bandwidth estimate $\hat{B}$ computed by each Westwood+ TCP sender. Finally, we have compared the measured throughputs with the steady-state throughputs predicted by (4), (7) and (15). The validation has been carried out by collecting data during the last 400 s of the simulation time in order to avoid the initial transients. For each Westwood+ TCP flow, we have computed the *relative error*:

$$relative\_error = \left|\frac{predicted\_throughput - measured\_throughput}{measured\_throughput}\right|$$

and for each equation model we have computed the *average relative error*:

$$average\_relative\_error = \frac{\sum_{i=1}^{N} relative\_error}{N}$$

In the following we denote model 1, model 2, model 3 to indicate the throughput models given by (4), (7) and (15), respectively. The proposed models do not take into account the slow start phase and assume an independent packet loss probability. Thus, we expect that the measured throughputs will differ from the throughputs predicted using (4), (7) and (15).

In order to summarise the simulation results, Figs. 5 and 6 plot the average errors as a function of the ratio of the retransmitted segments when $N$ ranges from ten to 80 and the backwards traffic is turned on and off, respectively. In particular, they show that when the reverse traffic sources are turned off (on) the average relative errors are below 15% (35%), which are comparable with the prediction errors shown in [10]. Model 1 provides the best throughput prediction, especially at low retransmission ratios. On the other hand model 2 and model 3 exhibit similar average errors at low retransmission ratios whereas model 3 exhibits

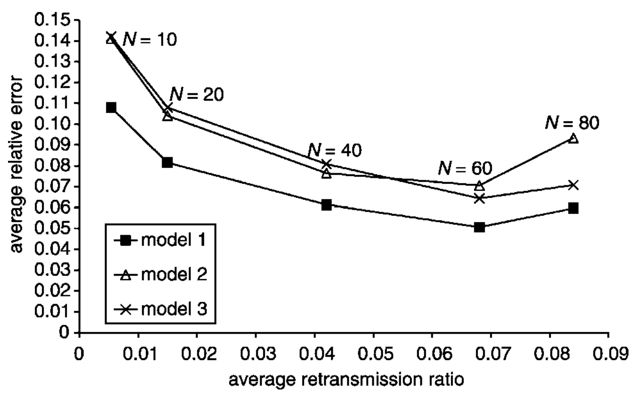


**Fig. 5** *Average errors against retransmission ratios obtained for values of N ranging from ten to 80 when the reverse traffic sources are turned off*

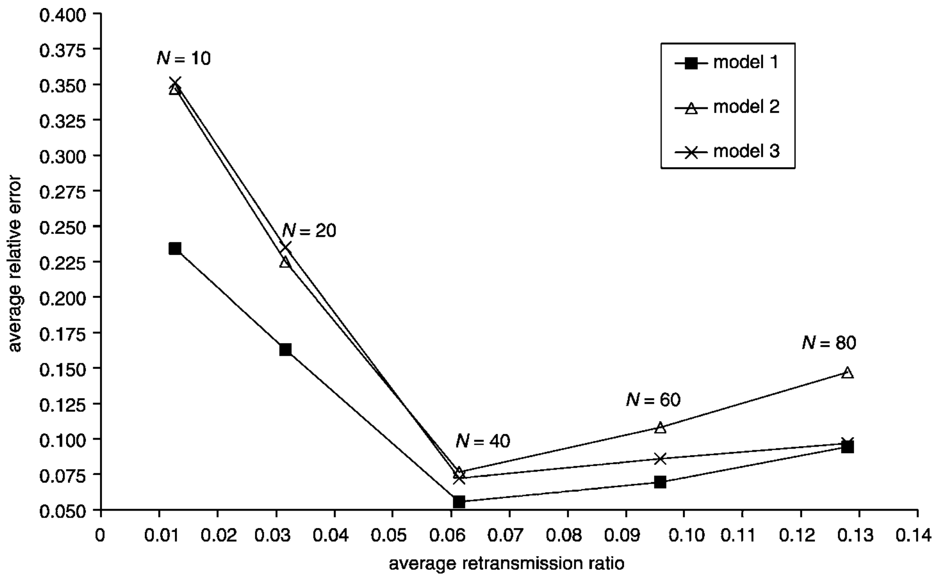


**Fig. 6** *Average errors against retransmission ratios obtained for values of N ranging from ten to 80 when the reverse traffic sources are turned on*

better performances than model 2 when the retransmission ratio increases. This result is not surprising since model 2 has been obtained under the assumption of a low loss rate whereas model 3 has been introduced to relax this assumption. Model 1 exhibits the best prediction since it explicitly uses the bandwidth estimate in the throughput equation (4), whereas models 2 and 3 approximate the bandwidth estimate as a function of the sending rate and of the loss probability using (8) and (12), respectively. Finally it is worth noting that the very low retransmission ratios obtained for $N = 10$ and $N = 20$ cause large transients (see also (11)), which results in higher prediction errors, since models 1–3 are only able to predict the steady-state behaviour of the Westwood + algorithm. In order to sustain this interpretation, Figs. 7 and 8 show the bottleneck queue dynamics obtained when $N = 10$ or $N = 60$ connections share the 10 Mb/s bottleneck and the reverse traffic is turned off (Figs. 9 and 10 show analogous results, obtained when the reverse traffic sources are turned on). By looking at these Figures, it is straightforward to note that, when $N = 10$, the

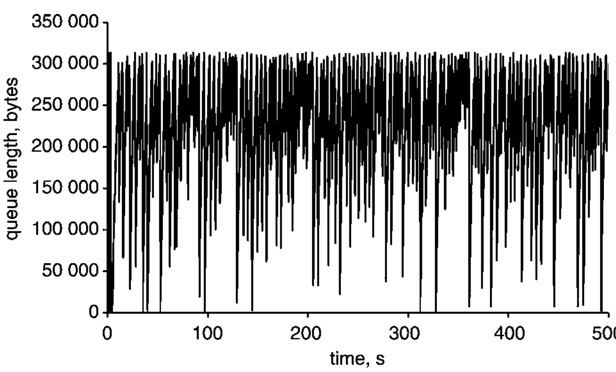


**Fig. 7** *Bottleneck queue dynamics obtained when N = 10 Westwood+ TCP connections share a 10 Mb/s bottleneck and the reverse traffic sources are turned off*

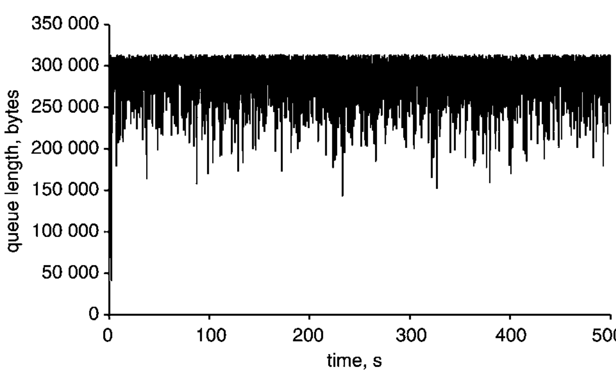


**Fig. 8** *Bottleneck queue dynamics obtained when N = 60 Westwood+ TCP connections share a 10 Mb/s bottleneck and the reverse traffic sources are turned off*

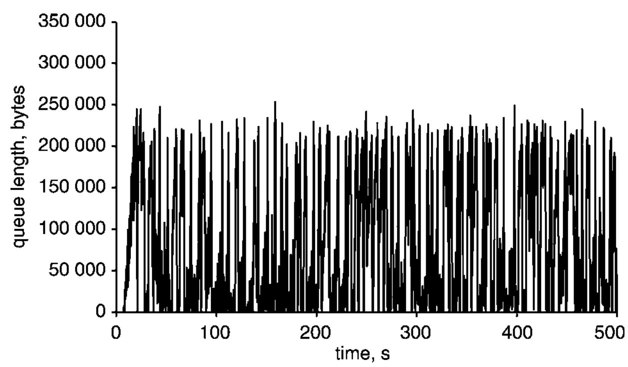


**Fig. 9** *Bottleneck queue dynamics obtained when N = 10 Westwood+ TCP connections share a 10 Mb/s bottleneck and the reverse traffic sources are turned on*

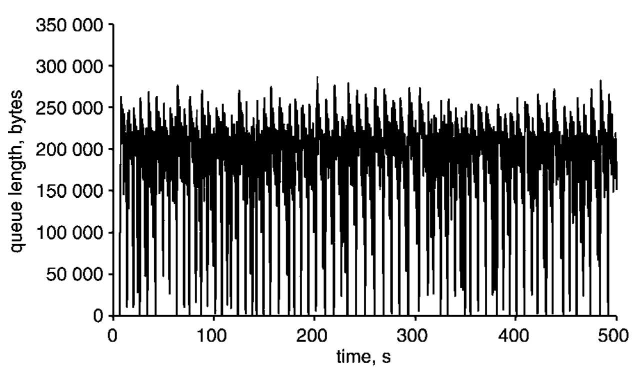


**Fig. 10** *Bottleneck queue dynamics obtained when N = 60 Westwood+ TCP connections share a 10 Mb/s bottleneck and the reverse traffic sources are turned on*

bottleneck queue exhibits very large oscillations, which means that the TCP sources are not operating under steady-state; conditions, on the other hand, when $N = 60$, the oscillations are considerably reduced, which means that the TCP sources are operating near the predicted equilibrium point (this phenomenon has also been shown in [22] and [23]). Thus, the large prediction errors obtained for $N = 10$ in Figs. 5 and 6 are due to the fact that the network system does not operate close to steady-state conditions because of the low levels of statistical multiplexing.

## 5 Conclusions

We have proposed a mathematical analysis of the AIADD paradigm to analyse the steady-state throughput provided by Westwood+ TCP. We have shown that both classic and Westwood+ TCP provide a throughput that is proportional to $1/\sqrt{p}$, that is they are friendly to each other. It has also been demonstrated that the throughput of Westwood+ TCP is proportional to $1/\sqrt{RTT}$ whereas the throughput of Reno TCP is proportional to $1/RTT$, i.e. Westwood+ TCP improves intra-protocol fairness. Finally, Ns-2 simulations have shown that the proposed models predict the throughput of an AIADD controlled TCP flow with an acceptable reasonable average for the prediction errors.

## 6 Acknowledgments

We thank the anonymous reviewers whose suggestions considerably improved ours presentations. This research was supported by the MIUR-FIRB project n. RBNE01BNL5 'Traffic Models and Algorithms for Next Generation IP networks Optimization (TANGO)'. coordinated by Prof. M. Ajmone Marsan.

## 7 References

1 Jacobson, V.: 'Congestion avoidance and control'. Proc. ACM Sigcomm, Stanford, CA, August 1988, pp. 314–329
2 Allman, M., Paxson, V., and Stevens, W.: 'TCP congestion control'. RFC 2581, April 1999
3 Floyd, S., and Henderson, T.: 'NewReno modification to TCP's fast recovery'. RFC 2582, April 1999
4 Clark, D.: 'The design philosophy of the DARPA internet protocols'. Proc. ACM Sigcomm, Stanford, CA, August 1988, pp. 106–114
5 Floyd, S., and Fall, K.: 'Promoting the use of end-to-end congestion control in the Internet', *IEEE/ACM Trans. Netw.*, 1999, **7**, (4), pp. 458–472
6 Mascolo, S.: 'Congestion control in high-speed communication networks', *Automatica*, 1999, **45**, (2)
7 Chiu, D.-M., and Jain, R.: 'Analysis of the increase and decrease algorithms for congestion avoidance in computer networks', *Computer Netw. ISDN Syst.*, 1989, **17**, (1), pp. 1–14
8 Mascolo, S.: 'Modeling the Internet congestion control as a time delay system: a robust stability analysis'. Presented at the IFAC Workshop on Time-delay Systems, Inria, Rocquencourt, 8–10 September 2003
9 Kelly, F.: 'Mathematical modeling of the Internet'. Proc. 4th Int. Cong. on Industrial and Applied Mathematics, July 1999
10 Padhye, J., Firoiu, V., Towsley, D., and Kurose, J.: 'Modeling TCP throughput: a simple model and its empirical validation'. Proc. ACM Sigcomm, Vancouver, BC, Canada, September 1998, pp. 303–314
11 Krishnan, R., Allman, M., Partridge, C., and Sterbenz, J.P.G.: 'Explicit transport error notification (ETEN) for error prone wireless and satellite networks'. BBN Technical Report No. 8333, 22 March 2002
12 Barakat, C., and Altman, E.: 'Bandwidth tradeoff between TCP and link-level FEC', *Comput. Netw.*, 2002, **39**, pp. 133–150
13 Balakrishnan, H., Padmanabhan, V.N., Seshan, S., and Katz, R.H.: 'A comparison of mechanisms for improving TCP performance over wireless links', *IEEE/ACM Trans. Netw.*, 1997, **5**, (6), pp. 756–769
14 Mascolo, S., Casetti, C., Gerla, M., Sanadidi, M., and Wang, R.: 'TCP Westwood: End-to-End bandwidth estimation for efficient transport over wired and wireless networks'. Presented at ACM Mobicom, Rome, Italy, July 2001
15 Grieco, L.A., and Mascolo, S.: 'Performance evaluation and comparison of Westwood+, New Reno and Vegas TCP congestion control', *ACM Comput. Commun. Rev.*, 2004
16 Mogul, J.C.: 'Observing TCP dynamics in real networks'. Proc. ACM Sigcomm, Baltimore, MD, August 1992, pp. 305–317
17 Grieco, L.A., and Mascolo, S.: 'End-to-End Bandwidth Estimation for Congestion Control in Packet Networks'. Presented at the 2nd Int. Workshop, QoS-IP, Milano, Italy, February 2003
18 Grieco, L.A., and Mascolo, S.: 'Westwood TCP and easy RED to improve fairness in high speed networks'. Proc. IEEE 7th Int. Workshop on Protocols For High-Speed Networks, Berlin, Germany, 22–24 April 2002
19 Dell'Aera, A., Grieco, L.A., and Mascolo, S.: 'Linux 2.4 Implementation of Westwood+ TCP with rate-halving: A Performance Evaluation over the Internet'. To presented at the IEEE Int. Conf. on Communications (ICC), 2004
20 Ns-2 network simulator (ver 2). LBL, URL: http://www-mash.cs.berkeley.edu/ns
21 Li, S.Q., and Hwang, C.: 'Link capacity allocation and network control by filtered input rate in high speed networks', *IEEE/ACM Trans. Netw.*, 1995, **3**, (1), pp. 10–25
22 Aström, K.J., and Wittenmark, B.: 'Computer controlled systems' (Prentice Hall, Englewood, Cliffs, NJ, 1997)
23 Halil, H.K.K.: 'Nonlinear systems' (Prentice Hall)
24 *ns*-2 Westwood+ implementation, available online at http://www-ictserv.poliba.it/mascolo/tcp%20westwood/modules.htm
25 Towsley, D., Gong, W.B., Hollot, C.V., and Misra, V.: 'Analysis and design of controllers for AQM routers supporting TCP flows', *IEEE Trans. Autom. Control*, 2002, **47**, (6), pp. 945–959
26 Chen, J., Paganini, F., Wang, R., Sanadidi, M.Y., and Gerla, M.: 'Fluid-flow Analysis of TCP Westwood with RED'. Presented at Globecom, San Francisco, CA, November 2003