Performance Effects of Two-way FAST TCP

Fei Ge[a], Sammy Chan[b], Lachlan L. H. Andrew[c], Fan Li[b], Liansheng Tan[a], Moshe Zukerman[b]

[a] *Dept. of Computer Science, Huazhong Normal University, Wuhan, P.R.China*
[b] *Dept. of Electronic Engineering, City University of Hong Kong, HKSAR, P.R.China*
[c] *Centre for Advanced Internet Architectures, Swinburne University of Technology, Australia*

Abstract

In this paper we study the performance of delay-based congestion control in the presence of queueing in the reverse path. Specifically, we consider FAST TCP in a single-bottleneck network by considering two scenarios, each corresponding to an equilibrium where a different direction of the bottleneck link is saturated. We argue that the local stability properties of FAST are unchanged by reverse traffic, and present expressions for the throughput of each flow and link. Finally, we consider the effect of bidirectional traffic on the dynamics of the number of flows in the network. We find conditions on the traffic loads under which the bottleneck shifts between the forward and reverse links, and conditions under which a given direction is always the bottleneck.

Keywords:
Delay-based TCP, reverse path queueing, throughput, efficiency

1. Introduction

As the Internet has grown to its present size, network congestion has been avoided by applying congestion control algorithms such as TCP Reno. This is especially significant given the expected exponential increase in Internet traffic and transmission capacity. To meet the challenges associated with the growing Internet, new congestion control algorithms are often proposed. Many of these [1, 2, 3] consider increased delay as an indicator of congestion. A common concern with this approach is that delay experienced by acknowledgement packets (ACKs) cannot be distinguished from that experienced by data packets; because ACKs cause much less congestion than data packets, it is not necessarily appropriate to reduce the sending rate simply because ACKs are traversing a congested link.

In this paper, we investigate the performance of delay-based congestion control in the presence of reverse-path congestion. As an example protocol, we take the "Fast AQM Scalable TCP" (FAST) protocol, which has been extensively studied [4, 2, 5, 6, 7, 8]. These studies indicate that FAST is scalable to meet the challenges of the future Internet.

FAST estimates the amount of queueing its packets experience, by taking the difference between the current round trip delay and the minimum delay experienced. It adjusts its sending rate to try to maintain a constant number of packets in queues throughout the network. If ACKs never experience congestion, then this approach allocates rates to flows according to proportional fairness [2].

When TCP flows traverse a link in both directions, each direction carries not only data packets but also for ACK packets. It is well-known that queueing of ACKs in the reverse path has adverse effect on TCP Reno throughout [9, 10], and this also applies to FAST [11]. However, most of the performance analysis on FAST assumed one-way transmission and ignored effects of the reverse direction. This paper fills a gap by providing an analysis of FAST with bi-directional flows.

In this paper we study the performance and stability of two-way FAST considering a network consisting of a single bidirectional link. Either direction of this link may be the bottleneck, giving rise to two distinct but symmetric scenarios.

1.1. Related work

There exists a significant body of work on the performance of two-way TCP. Zhang *et al.* [10] studied the dynamics that result from two-way Tahoe TCP traffic. They observed two phenomena, ACK-compression and an out-of-phase queue-synchronization mode which adversely affect link utilization. Kalampoukas *et al.* [9] provided an understanding of ACK compression by quantitatively analyzing the periodic burst behavior of the source IP queue and suggested that the degradation in throughput due to two-way traffic can be significant. Balakrishnan and Padmanabhan [12] explained that the two-way traffic degraded performance due to bandwidth asymmetry because of adverse interaction between data packets in the forward direction and ACKs in the reverse direction. Truong *et al.* [13] derived an upper bound and an approximation for the two-way transmission capacity in wireless ad hoc networks. Lopez-Aguilera *et al.* [14] analyzed two-way transmission capacity of Asymmetric Access Point in wireless LANs. Lakshman *et al.*

[15] determined the throughput as a function of buffering round-trip times and normalized asymmetry, and showed how performance degradation can be alleviated by per-connection buffer and bandwidth allocation in the reverse path. Sun *et al.* [16] verified instability effects of two-way traffic in a TCP/AQM system by experiments. Fu *et al.* [17] improved TCP Vegas throughput over the forward path.

The works on two-way TCP cannot be directly applicable to the FAST flows in the two scenarios considered in this paper before they are carefully extended and revised, but give some useful enlightenment. As preliminary works of the topic, we have analyzed FAST throughput in the asymmetric single-bottleneck network where one FAST flow transmits in one direction and its ACK packets are transmitted in the reverse direction [18, 19].

In the following of the paper, we first build a model of FAST in both scenarios in Section 2. It is argued in Section 3 that reverse queueing does not fundamentally change the stability issues associated with FAST. The throughput of FAST flows in the two scenarios is obtained in Section 4.1, and the utilization of each link by data packets (excluding ACKs) is derived in Section 4.2. Then we discuss two additional examples with balanced parameters to have higher network efficiency in Section 5.1. In Section 5.2, we study the impact of VBR cross traffic on FAST. In Section 6, we consider the probability of being in each of the scenarios, by constructing a Markov chain representing the number of flows in each direction.

2. Modeling a Network with Two-way FAST Traffic

In this section, we will consider a network with two-way FAST flows. A duality model [20], [21] of end-to-end congestion control has been applied to study the performance of one-way FAST in various publications [4, 2, 6, 7]. In a network with two-way FAST flows, the data packets of FAST flows in one direction share the same router queue and capacity with the ACK packets of flows in the opposite direction. This sharing not only has capacity implications but also affects delay of both data packets and ACK packets. Therefore, there is a need to extend the model of [21] to be fit for two-way FAST flows.

2.1. Network model with two-way traffic

A network consists of a set L of unidirection links; bidirectional links are represented by two separate unidirectional links. Each link $l \in L$ has a capacity of c_l [bit/s].

There is a set S of FAST flows. Each flow $s \in S$ consists of two subflows, denoted s^+ and s^-, which represent the forward traffic consisting of data packets and the reverse traffic consisting of ACKs. Throughout, a superscript $^+$ denotes a quantity pertaining to a forward flow s^+, and a superscript $^-$ denotes a quantity pertaining to a reverse flow s^-. Each subflow has the same window size, w_s, measured in packets.

At time t, flow s sends at rate $x_s(t) = w_s(t)/(d_s + q_s(t))$ packet/s, inducing rates where $q_s(t)$ is the queueing delay experienced by flow s. Let $x_s^+ = x_s P_s^+$ and $x_s^- = x_s P_s^-$ [bit/s] on the subflows s^+ and s^-, where P_s^+ and P_s^- [bit/packet] are the sizes of packets on the forward flow s^+ and the reverse flow s^-. Note that the units of x_s^+ and x_s differ by a factor of bit/packet, the unit of P_s^+.

The use of links by flows is characterized by the routing matrix $R = (R^+ \quad R^-)$, where $R^+ = (r_{ls}^+)$ with $r_{ls}^+ = 1$ if flow s^+ uses link l and 0 otherwise, and R^- defined analogously.

Let $y_l(t)$ be the aggregate source rate at link l, then $y_l(t) = \sum_s r_{ls}^+ x_s^+(t) + \sum_s r_{ls}^- x_s^-(t)$. In vector notation,

$$y(t) = \begin{pmatrix} R^+ & R^- \end{pmatrix} \begin{pmatrix} x^+(t) \\ x^-(t) \end{pmatrix}, \tag{1}$$

where $x^+(t) = (x_1^+(t), x_2^+(t), \ldots, x_S^+(t))^T$ and $x^-(t) = (x_1^-(t), x_2^-(t), \ldots, x_S^-(t))^T$ are S dimensional vectors, $y(t) = (y_1(t), y_2(t), \ldots, y_L(t))^T$ is a L dimensional vector, and $()^T$ denotes matrix transposition.

Each link l is associated with a price $p_l(t)$ as its congestion measure at time t, representing queueing delay. Let $q_s^+(t)$ be the end-to-end congestion measure for flow s^+, $q_s^+(t) = \sum_l r_{ls}^+ p_l(t)$. Let $q_s^-(t)$ be the end-to-end congestion measure for flow s^-, $q_s^-(t) = \sum_l r_{ls}^- p_l(t)$. The total congestion measure for source s is $q_s(t) = q_s^+(t) + q_s^-(t)$. In vector notation, we have

$$q(t) = (R^+ + R^-)^T p(t),$$

where $p(t) = (p_1(t), p_2(t), \ldots, p_L(t))^T$, is a L dimensional vector, $q(t) = (q_1(t), q_2(t), \ldots, q_S(t))^T$ is a S dimensional vector.

2.2. Dynamic model of FAST

FAST adjusts its congestion window [2] by

$$\dot{w}_s(t) = \gamma_s\left(\frac{-q_s(t)}{d_s + q_s(t)} w_s(t) + \alpha_s\right) \tag{2}$$

where α_s is the number of the data packets FAST attempts to maintain in buffers throughout the network, $\gamma_s = 1/2$ is a step-size parameter, and $d_s = d_s^+ + d_s^-$ is the total propagation delay, where d_s^+ and d_s^- are the propagation delays of s^+ and s^-, respectively. The throughput of source s can be modeled by

$$x_s(t) = \frac{w_s(t)}{d_s^+ + q_s^+(t) + d_s^- + q_s^-(t)}.$$

The aggregated throughput on links can be obtain by (1). We model the queueing delay of a link as a truncated integral of the difference between incoming traffic and the capacity [5]:

$$\dot{p}_l(t) = \begin{cases} \frac{y_l(t)}{c_l} - 1 & \text{if } p_l(t) > 0 \\ \left(\frac{y_l(t)}{c_l} - 1\right)^+ & \text{otherwise,} \end{cases} \tag{3}$$

where $(x)^+ = \max(x, 0)$.

3. Stability

This section shows that a single-link network with bidirectional FAST flows can usually be mapped to one with unidirectional flows. This allows the existing large body of stability results to be applied; see for example [6, 7, 22]. Note that we do not give specific conditions under which FAST is stable with bidirectional flows; it was shown in [6] that such conditions are extremely sensitive to assumptions of the models, and so such a result gives limited insight. It is more useful to see whether or not bidirectional traffic causes a fundamental change to the problem.

Consider a network with bidirectional FAST flows, in which all links except for one duplex link between nodes 1 and 2 are non-bottlenecks. The (possibly bottleneck) link from node 1 to node 2, arbitrarily considered the "forward" link, is denoted l_F and the link from node 2 to node 1 is denoted as l_B. (When used in subscripts, these will be denoted simply F and B.) The capacities are c_F and c_B respectively. The non-bottleneck links with very large capacity are ignored. Let F^+, F^-, B^+ and B^- be the subsets of FAST flows such that $r_{Fs}^+ = 1$, $r_{Fs}^- = 1$, $r_{Bs}^+ = 1$ and $r_{Bs}^- = 1$, respectively. Intuitively, a flow $s \in F^+$ sends data packets through link l_F, and the other groups are analogous. If routing is symmetric, then $F^+ = B^-$ and $F^- = B^+$, but this need not be the case if routing is asymmetric.

One or both of l_F and l_B must be a bottleneck. If there is some cross-traffic not bottlenecked at either of these links, then it is very unlikely that both links will be bottlenecks. We will thus only consider the case that only one is a bottleneck.

Consider without loss of generality the case that l_F is the bottleneck. The dynamics of this case are exactly those of a network consisting of $|F^+| + |F^-|$ unidirectional flows, where flows $s \in |F^+|$ have packets of size P_s^+ and flows $s \in |F^-|$ have packets of size P_s^-. The α_s of each flow in the unidirectional network is the same as that in the bidirectional network; since l_B is not a bottleneck, the size of the packets traversing l_B does not affect the dynamics (except for serialization delay). Existing stability results for such unidirectional results apply directly to the stability of the bidirectional FAST flows.

Stability analysis does not help us to obtain how much throughput FAST flows can achieve in different scenarios. In the next section, we will focus on performance including throughput and efficiency of FAST flows in two scenarios.

4. Performance Analysis

4.1. Throughput

The throughput of FAST flows in equilibrium can be obtained by the Theorem 1 of [2], which is applicable in any networks. In the symmetric single-bottleneck network with one-way traffic, every FAST flow can achieve a throughput of $x_s^+ = (\alpha_s / \sum_i \alpha_i) c_l$ because FAST flows achieve α_s-weighted proportional fairness. However, reverse queueing complicates the throughput of FAST.

The throughput of FAST in a simple asymmetric link with one-way FAST flows is considered in [19], but the conclusions are not applicable to two-way traffic. It is necessary to reconsider two-way FAST throughput in such a network.

Consider again the dumbell topology of Section 3, consisting of two potential bottlenecks l_F and l_B. We begin with FAST throughput analysis from Theorem 1 of [2], that is, throughput of FAST source in equilibrium is given by $x_s = \alpha_s / q^*$, where q^* is the equilibrium sum of the queueing delays on l_F and l_B.

Theorem 1. *Link l_F is a bottleneck if and only if both*

$$|F^+| + |F^-| > 0$$

and

$$\frac{\sum_{s \in F^+} \alpha_s P_s^+ + \sum_{s \in F^-} \alpha_s P_s^-}{c_F} \geq \frac{\sum_{s \in F^-} \alpha_s P_s^+ + \sum_{s \in F^+} \alpha_s P_s^-}{c_B}. \quad (4)$$

The conditions under which l_B is a bottleneck are analogous.

If l_F is a bottleneck, then the rate of each flow is

$$x_s = \frac{\alpha_s}{\sum_{\sigma \in F^+} \alpha_\sigma P_\sigma^+ + \sum_{\sigma \in F^-} \alpha_\sigma P_\sigma^-} c_F; \tag{5}$$

otherwise

$$x_s = \frac{\alpha_s}{\sum_{\sigma \in B^+} \alpha_\sigma P_\sigma^+ + \sum_{\sigma \in B^-} \alpha_\sigma P_\sigma^-} c_B. \tag{6}$$

Proof. The total data rate on link l_F is

$$y_F = \sum_{s \in F^+} \frac{\alpha_s}{q^*} P_s^+ + \sum_{s \in F^-} \frac{\alpha_s}{q^*} P_s^-,$$

and l_F is a bottleneck if and only if this equals the capacity, c_F, making the left hand side of (4) equal q^*. To satisfy the feasibility constraint that $y_l \le c_l$ for all l, the right hand side is at most q^*. At least one of l_F and l_B must be a bottleneck, or else q^* would be 0, making x_s infinite for all s. This implies that (4) is necessary and sufficient for l_F to be a bottleneck.

If l_F is a bottleneck, then q^* is given by the left hand side of (4), whether or not l_B is also a bottleneck. Substituting this into $x_s = \alpha_s / q^*$ gives (5), and (6) is derived analogously. □

Note that the weak inequality in (4) implies that it is possible for both l_F and l_B to be bottlenecks simultaneously.

Theorem 1 clearly shows how the reverse traffic affects the throughput of forward traffic under FAST. It also admits a range of simple special cases.

If all flows have equal α_s and equal packet sizes P^+ and $P^- = bP^+$, then it reduces to l_F being a bottleneck if

$$\frac{|F^+| + b|F^-|}{|F^-| + b|F^+|} \ge \frac{c_F}{c_B}.$$

or equivalently,

$$|F^+| \left(1 - b\frac{c_F}{c_B}\right) + |F^-| \left(b - \frac{c_F}{c_B}\right) \ge 0. \tag{7}$$

If moreover the number of flows in each direction is equal, then l_F is a bottleneck if $c_F < c_B$. Clearly (7) leads to degenerate cases if the asymmetry between the capacities, c_F/c_B, exceeds that between the data packets and acknowledgements, b; for example, if $1 > b > c_B/c_F$ then l_F is never a bottleneck, since flows using it are always bottlenecked by l_F instead.

Conversely, if the link is symmetric, with $c_F = c_B$, and if all flows have equal sized packets and equal sized acknowledgements, with $P^+ > P^-$, then (4) reduces to

$$\sum_{s \in F^+} \alpha_s \geq \sum_{s \in F^-} \alpha_s.$$

4.2. Efficiency

The efficiency of the network can be measured by the utilization of the bottleneck link. We define the *utilization* U_l of a link l as the ratio of the aggregated throughput of all FAST data packets (excluding ACKs) to the link capacity c_l.

This can easily be calculated by summing the throughputs of the individual flows. Specifically, if l_F is the bottleneck then

$$U_F = \frac{\sum_{s \in F^+} \alpha_s P_s^+}{\sum_{s \in F^+} \alpha_s P_s^+ + \sum_{s \in F^-} \alpha_s P_s^-} \tag{8}$$

and

$$U_B = \frac{\sum_{s \in F^-} \alpha_s P_s^+}{\sum_{s \in F^+} \alpha_s P_s^+ + \sum_{s \in F^-} \alpha_s P_s^-} \frac{c_F}{c_B}, \tag{9}$$

and similarly if l_B is the bottleneck.

Note that the throughput of the bottleneck link depends explicitly only on the TCP parameters, while the throughput of the link in the other direction depends also on the ratio $k_c = c_F/c_B$. The first half of that sentence must be interpreted carefully: In a network in which l_F is the bottleneck, it cannot be concluded that the throughput of l_F cannot be changed by altering k_c. Specifically, for a given set of TCP parameters, k_c will determine whether or not l_F is indeed the bottleneck. This will be studied in greater detail in the next section.

5. Numerical Results

To verify our analyses, we use ns2 [23, 24] to simulate several example networks. We first study three examples with different configurations of bidirectional FAST flows and network parameters, then investigate the performance improvement of the network in balanced parameters by two additional examples. Afterwards, we study the performance impact of Variable Bit Rate traffic on FAST flows. Finally, we summarize the insight gained by the numerical studies.

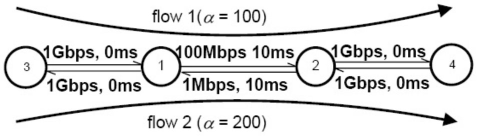

Figure 1: Topology of Example 1

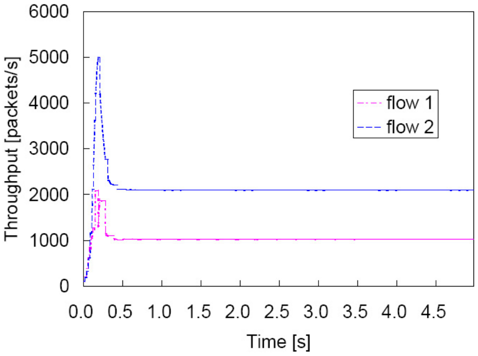

Figure 2: Throughput rates of flows 1 and 2 of Example 1

5.1. Model Validation

We present here five examples that demonstrate the accuracy of our analyses. In the first three examples, FAST traffic is transmitted over the three links between two nodes designated 3 and 4, in which (one direction of) the middle link is the bottleneck. The packet size of FAST is set to 1000 bytes, the ACK size is set to 40 bytes, and the γ parameter of every FAST flow is set to 0.5.

Example 1. *Consider the topology shown in Fig. 1. The bottleneck capacities are* $c_F = 100\,Mb/s$ *and* $c_B = 1\,Mb/s$*. The round trip propagation time is 20 ms. Two FAST flows,* $s \in \{1, 2\}$*, transmit data packets over the links from node 3 to node 4, ACK packets over the links from node 4 to node 3. Flow* s *has* $\alpha_s = 100s$ *packets.*

The throughput of each flow is plotted in Fig. 2. Here, $|F^-|$ is empty but $(1 - bc_F/c_B) < 0$, and so by (7) l_F is not a bottleneck, which implies link l_B must be one. Therefore by (6) and the fact that $x_s^+ = x_s P_s^+$, the

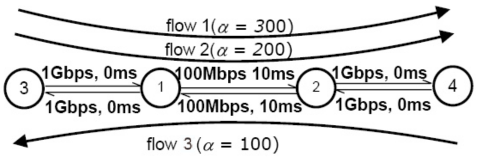

Figure 3: Topology of Example 2

throughputs of flows 1 and 2 are

$$x_1 = \frac{\alpha_1}{(\alpha_1 + \alpha_2)P^-} c_B \approx 1042 \text{ packet/s} \qquad (x_1^+ \approx 8.3\,\text{Mb/s}), \tag{10}$$

$$x_2 = \frac{\alpha_2}{(\alpha_1 + \alpha_2)P^-} c_B \approx 2083 \text{ packet/s} \qquad (x_2^+ \approx 16.7\,\text{Mb/s}). \tag{11}$$

Although l_B is saturated, $U_B = 0$ because only ACK packet flows on the link. The utilization of the link l_F is $U_F = 0.25$. This shows that the network is not efficient because of link asymmetry.

Example 2. *In this example, the bottleneck link is symmetric with capacity of $c_F = c_B = 100\,Mb/s$ and round trip propagation time of 20 ms, as shown in Fig. 3. Two FAST flows transmit data packets over the links from node 3 to node 4 and one from node 4 to node 3. They set α parameters at 300, 200, 100 packets, respectively.*

Here $\sum_{s \in F^+} \alpha_s P_s^+ = 5 \times 10^5$ bytes and $\sum_{s \in F^-} \alpha_s P_s^+ = 10^5$ bytes, while the contributions to delay from the ACKs are negligible, whence l_F is bottleneck by (4). By (5),

$$x_1 = \frac{\alpha_1}{(\alpha_1 + \alpha_2)P^+ + \alpha_3 P^-} c_F \approx 7440 \text{ packet/s} \quad (x_1^+ = 59.5\,\text{Mb/s}), \tag{12}$$

$$x_2 = \frac{\alpha_2}{(\alpha_1 + \alpha_2)P^+ + \alpha_3 P^-} c_F \approx 4960 \text{ packet/s} \quad (x_2^+ = 39.7\,\text{Mb/s}), \tag{13}$$

$$x_3 = \frac{\alpha_3}{(\alpha_1 + \alpha_2)P^+ + \alpha_3 P^-} c_F \approx 2480 \text{ packet/s} \quad (x_3^+ = 19.84\,\text{Mb/s}). \tag{14}$$

By simulation, the equilibrium rates are $x_1 = 7125$, $x_2 = 4750$ and $x_3 = 2500$ packet/s ($x_1^+ = 57$, $x_2^+ = 38$, $x_3^+ = 20\,\text{Mb/s}$).

The throughput of flow 3 is far below the link capacity of 100 Mb/s, since its ACKs are limited by link l_F which is saturated. Hence Link l_B is not saturated, with the utilization $U_B = 0.198$, while link l_F obtains $U_F = 0.992$.

Example 3. *In this example, two FAST flows transmit data packets in the opposite directions. Flow 1 transmits data packets in the direction from* *node 3 to node 4, with $\alpha = 100$ packets. Flow 2 transmits data packets in the direction from node 4 to node 3, with $\alpha = 30$ packets. The bottleneck link is (one direction of) a full duplex link with capacity $c_F = 100$ Mb/s from node 1 to node 2, and $c_B = 10$ Mb/s from node 2 to node 1*[1]*. The round trip propagation delay is 20 ms.*

By Theorem 1, link l_B is the bottleneck. This gives

$$x_1 = \frac{\alpha_1}{\alpha_2 P^+ + \alpha_1 P^-} c_B \approx 3676 \text{ packet/s} \qquad (x_1^+ = 29.4\,\text{Mb/s}). \tag{15}$$

$$x_2 = \frac{\alpha_2}{\alpha_2 P^+ + \alpha_1 P^-} c_B \approx 1103 \text{ packet/s} \qquad (x_2^+ = 8.82\,\text{Mb/s}). \tag{16}$$

From simulation, the throughputs are $x_1 = 3533$ and $x_2 = 1075$ packet/s ($x_1^+ = 28.3$ and $x_2^+ = 8.6$ Mb/s). The bottleneck is link l_B, but $U_B = 0.88$; Compared to Example 2, the utilization of the saturated link in this example is small because ACKs take much of the capacity of the link. The utilization of link l_F is $U_F = 0.294$.

For two-way Tahoe flows, there is no condition in which both directions of the bottleneck link are fully utilized [10]. It is also the case for two-way FAST flows. However we can find the factors that can increase the utilization for FAST. We now consider two additional examples in which the ratio of the numbers of the numbers of flows is close to the threshold given by (17), at which point both links are saturated.

Example 4. *In this example, flow 3 is added to Example 3. Flow 3 transmits data as the same direction as flow 1, and uses $\alpha_3 = 200$ packets.*

In this example, $\sum_{s \in F^+} \alpha_s / \sum_{s \in F^-} \alpha_s = c_F / c_B$. At a first glance, the utilization of the bottleneck link may seem to be 1. However it is not the case because of the ACK packet flows.

By Theorem 1, the bottleneck is l_B. And the total throughput of l_F is

$$x_1 + x_3 = \frac{\alpha_1 + \alpha_3}{\alpha_2 P^+ + (\alpha_1 + \alpha_3) P^-} c_B \approx 8929 \text{ packet/s} \quad (x_1^+ + x_3^+ = 71.4 \text{ Mb/s}).$$

Then $x_1 \approx 23.8$ Mb/s, $x_3 \approx 47.6$ Mb/s since rates are in proportion to α. Similarly $x_2^+ = c_B - (x_1^+ + x_3^+) \frac{P^-}{P^+} = 7.14$ Mb/s.

The utilizations are $U_F = 0.714$, and $U_B = 0.714$. The additional flow 3 does not significantly reduce the utilization of link l_B (from 0.88 to 0.714),

[1] Note that if the link is symmetric, and there is only one flow in each direction, we can easily obtain the result that the flow with larger α will experience congestion.

but increases the utilization of link l_F (from 0.294 to 0.714), which makes the network more efficient. This verifies that increasing $\sum_{s \in F^+} \alpha_s$ results in a higher U_F and/or a lower U_B, all else being equal.

The simulation results are that flows 1, 2 and 3 achieve equilibrium throughput rates of 2663, 900 and 5300 packet/s (or 21.3, 7.2 and 42.4 Mb/s), respectively. The simulation results are a bit smaller than deduced above. The reason is that the parameters of the network are balanced, meaning that each link is almost saturated. The bottleneck may switch due to the discrete data packet number in simulation, which leads to the sending rate not being exactly the capacity. And then FAST will adjust its congestion window so that the bottleneck changes. When changing the scenario, the link utilization is smaller and the network efficiency is lower.

The switching process is obviously related to the round trip propagation time of the network. In the next example the round trip propagation time of the path is set shorter.

Example 5. *In this example, two FAST flows transmit data in opposite directions called flows 1 and 2, whose α parameters are 30 packets. The bottleneck link has the same capacities 10 Mb/s in two directions. The round trip propagation time is 2 ms.*

This is a balanced network. The bottleneck switches frequently between l_B and l_F due to the discrete data packet number. However the throughput of FAST flows in both scenarios are stable and approximately the same. The theoretical throughput results are $c/(1+b) = 9.62\,\mathrm{Mb/s}$ according to Theorem 1. However the simulation results are that both flow 1 and flow 2 achieve the throughput rate of 9.4 Mb/s, which is less than theoretical results. Some network efficiency is lost during the switching process of the two scenarios.

Compared with Example 4, it is obvious that large round-trip propagation delay will increase the switching time, and then decrease the throughput and link utilization. Therefore the small round-trip propagation delay will achieve higher efficiency in a balanced network.

5.2. Variable bit rate traffic impact

Variable bit rate (VBR) cross traffic will interfere with the network by competing for the same resources with FAST. Such interfering traffic may not easily be modeled. We will investigate the impact of VBR on FAST through simulation. As an example of typical VBR traffic, we use the Star Wars IV trace, offered by Fittex [25].

The Star Wars IV traffic are added to Example 3, transmitted on the path from node 4 to node 3, from 10 s to 80 s. We arbitrarily select the frames of the movie from 1431280 ms. In the 70 s simulation, 908 frames are played, the average frame size is 5032 bits. Then the average rate is $908 \times 5032/70 = 65$ kb/s. We consider an extreme situation that 20 users are watching the movie at the same time without multi-cast technology. To maximize variability, we consider the case that all streams are perfectly synchronized. Therefore the whole VBR traffic, 1.3 Mb/s, will have a greater influence on FAST flows than independent VBR traffic does. We plot the throughput of the two FAST flows and the frame size of Star Wars IV in Fig. 4.

Obviously, the different and unexpected frame size of Star Wars IV results in the negative pulses of both FAST flows. The average throughput rates of flows 1 and 2 are 3613.96 and 1058.39 packet/s, respectively. The throughput rate of flow 2 with VBR is slightly lower than without VBR, while the throughput rate of flow 1 is made larger.

We then let 200 Star Wars IV traffic be transmitted on the path from node 3 to node 4, from 10 s to 80 s, in Example 3. The same negative pulses of both flows as in Fig. 4 appear. The average throughput rates of flows 1 and 2 are 3523.73, 1064.99 packet/s, respectively.

Next, we let Star Wars IV traffic to be transmitted on the path from two directions in Example 3. The negative pulses of both flows appear too. The average throughput rates of flows 1 and 2 are 3602.57, 1053.76 packet/s, respectively.

From the above results, we can conclude that a small amount of VBR traffic will lead to minimal variability of the rate of FAST. And because the bottleneck in Example 3 is l_B (the path from node 4 to node 3 is congested), VBR traffic through this path will influence FAST more seriously than VBR traffic through the noncongested path from node 3 to node 4.

5.3. Summary

Here, we list the main parameters of FAST flows and the network of the five examples in Table 1, their throughput and link utilization results in Table 2, and the percentage difference defined as $|(analysis - simulation)/simulation|$ in Table 3.

In all examples, FAST flows are stable whether the bottleneck link is l_F or l_B. The simulation throughput of FAST flows in the first three examples are approximately consistent with the theoretical results. The simulation link utilizations in the first three examples are also approximately consistent with

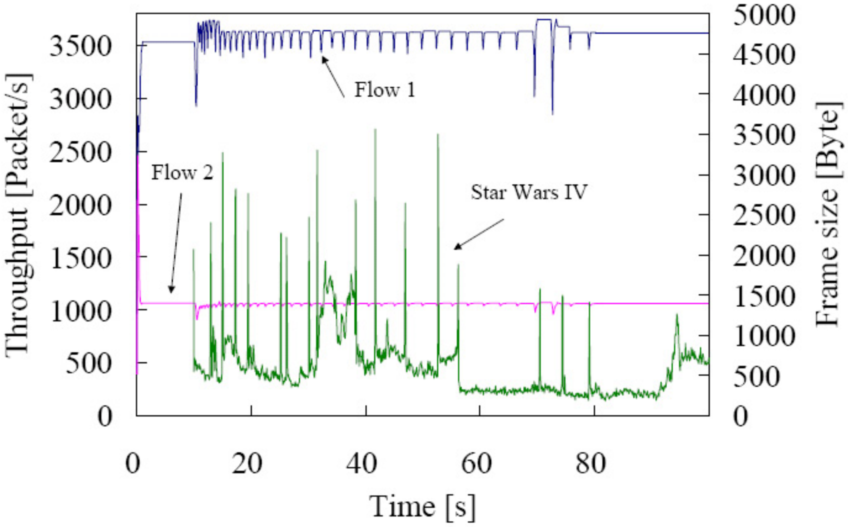

Figure 4: Throughput rates of flows 1 and 2 with Star Wars IV

Table 1: Setting of examples

Example	c_F[Mb/s]	c_B[Mb/s]	d [ms]	α	stable	Bottleneck
1	100	1	20	(100,200)	Yes	l_B
2	100	100	20	(300,200,100)	Yes	l_F
3	100	10	20	(100,30)	Yes	l_B
4	100	10	20	(100,30,200)	Yes	$l_F - l_B$
5	10	10	2	(30,30)	Yes	$l_F - l_B$

the theoretical results, while the sum of link utilization of the two directions is smaller.

The sum of link utilization of the two directions in the fourth example are larger due to their more balanced settings. More balanced setting makes the network switch its bottleneck frequently, which leads to the simulation throughput results of the fourth example being slightly smaller than the theoretical results. If the switch time is small, as the propagation delay in the fifth example is 2 ms (much smaller than 20 ms in the fourth example), the throughput of FAST flows increase, and the simulation results of throughput and links utilization are closer to those of theoretical results.

Table 2: Comparisons of analytical and simulation results

Example	Throughput [Mb/s] (Simulation)	Throughput [Mb/s] (Analysis)	Utilization (Simulation)	Utilization (Analysis)
1	(8.2, 16.8)	(8.3, 16.7)	(25%, 0)	(25%, 0)
2	(57, 38, 20)	(59.5, 39.7, 19.84)	(95%, 20%)	(99.2%, 19.8%)
3	(28.3, 8.6)	(29.4, 8.8)	(28.3%, 86%)	(29.4%, 88%)
4	(21.3, 7.2, 42.4)	(23.8, 7.14, 47.6)	(63.7%, 72%)	(71.4%, 71.4%)
5	(9.4, 9.4)	(9.62, 9.62)	(94%, 94%)	(96.2%, 96.2%)

Table 3: Percentage difference between analytical and simulation results

Example	Throughput	Utilization
1	(1.22%,0.6%)	(0, -)
2	(4.39%,4.47%,0.8%)	(4.42%, 1%)
3	(3.89%, 2.32%)	(3.89%, 2.32%)
4	(11.74%, 0.83%, 12.26%)	(12.09%, 0.83%)
5	(2.34%, 2.34%)	(2.34%, 2.34%)

6. Arrival and Departure of Flows

So far, the number of FAST flows in the network has been considered fixed. However, in practice flows come and go as file transfers complete. This section will consider the dynamics of a system in which flows arrive in each direction as independent Poisson processes of rates λ_F and λ_B, and send exponential amounts of data with equal means $1/\mu$ [bits]. All flows have equal α_s, P_s^+ and $P_s^- = bP_s^+$. The aim is to determine the fraction of time a dumbbell network will spend with link l_F as the bottleneck

It will be assumed that there is a timescale separation between that of TCP dynamics and that of flow arrivals and departures, so that the rates reach the equilibrium specified by (5) or (6) as soon as the number of flows change. The system can then be represented by a two dimensional Markov chain, with state $(n, m) = (|F^+|, |F^-|)$. The service rates depend on which link is the bottleneck, and hence on the state. The form of the service rates depend on whether the (n, m) is above or below a dividing line with slope θ; that is, on whether or not $n \geq \theta m$. When $n = \theta m$, the load is balanced, so that both links are fully saturated.

Consider first the degenerate cases of (7) in which the coefficients of $|F^+|$ and $|F^-|$ have the same sign. If both are positive, link l_F is always the bottleneck and so $\theta = 0$; if both are negative, link l_B is always the bottleneck and so $\theta = \infty$. In these cases, the fraction of time that each link is a bottleneck does not depend on any of the above assumptions.

The remainder of the section will consider the usual non-degenerate case that $b < \min(c_F, c_B)/\max(c_F, c_B) \leq 1$; the anomalous case when $b > 1$ (ACKs are larger than data packets) is analogous. In this case, by (7), the slope of the dividing line is

$$\theta = \frac{c_F - bc_B}{c_B - bc_F}. \tag{17}$$

The service rate of each flow is then $x_s P^+ \mu$ where, by (7), x_s is given by (5) if $n \geq m\theta$ and (6) otherwise. Specifically, the transition rates are:

transition	**rate**
$(n, m) \to (n+1, m)$	λ_F
$(n, m) \to (n, m+1)$	λ_B
$(n, m) \to (n-1, m)$	$n\mu C(n, m)$
$(n, m) \to (n, m-1)$	$m\mu C(n, m)$

where

$$C(n, m) = \begin{cases} \frac{c_F}{n+bm} & \text{if } n \geq \theta m \text{ and } n > 0 \\ \frac{c_B}{bn+m} & \text{if } n < \theta m \\ 0 & \text{otherwise.} \end{cases}$$

Like in a standard two-class processor sharing queue with equal service time distribution for each class, at each (n, m), the service rate for each class is proportional to the number of jobs of that class.[2] However, (when $\theta \in (0, \infty)$) the total service rate is not constant, but instead depends on the ratio n/m. The maximum rate of $\mu(c_F + c_B)/(1 + b)$ occurs at the dividing line $n = m\theta$, and there are local minima of μc_F when $m = 0$ and μc_B when $n = 0$.

This Markov chain may be stable (ergodic), or unstable in both dimensions simultaneously. When acknowledgements are ignored, the system is unstable in the dimension of link l if the total offered load on link l exceeds its capacity: $\sum_{\text{flows } s \text{ using link } l} \lambda_s/\mu \geq c_l$ [26, 27]. For the bidirectional case studied here, the number of flows using link l_F will be unstable

[2] This may be seen as "fair". Alternatively, it may be considered unfair, since the flows whose ACKs are bottlenecked cause much less of the congestion but still only get the same rate. This could be addressed using size-based priority scheduling on the link, although that would cause other complications.

if $(\lambda_F + b\lambda_B) \geq c_F$, and similarly the number using l_F will be unstable if $(\lambda_B + b\lambda_F) \geq c_B$. Notice that in either case, both n and m are unbounded, since all flows use both links.

Although a necessary condition for stability is that $\sum \lambda_s/\mu < c_l$ on each link, this is not sufficient, as shown by several examples in [27]. However, for unidirectional flows, it is sufficient in many cases: when rates are "α-fair", it is sufficient even if file sizes are not exponential [28], or timescale separation is not assumed [29]. When acknowledgements are considered, it remains an open question whether the simple condition of stability of each link individually is sufficient for network stability.

When the Markov chain is unstable, the flows enter a fluid regime, in which the ratio $n/m = \lambda_F/\lambda_B$ with probability 1. Hence, if $\lambda_F/\lambda_B > \theta$ then l_F is almost always the bottleneck, and if $\lambda_F/\lambda_B < \theta$ then l_B is almost always the bottleneck. When $\lambda_F/\lambda_B < \theta$, the fraction of time each link is the bottleneck is not easy to determine, but both links are "almost" the bottleneck all of the time: for any $\epsilon > 0$ and $l \in \{l_F, l_B\}$ the saturation is $y_l/c_l \geq 1 - \epsilon$ with probability 1.

The fraction of time each link is a bottleneck depends on the total load, as well as the amount of traffic on each path. Figure 5 shows this for a single bidirectional link, with $\lambda_F (= \lambda_B)$ change from 0 to $1/(1+b)$, $1/\mu = 1\,\mathrm{Mb}$, $c_F = 1\,\mathrm{Mb/s}$, $c_B = 2\,\mathrm{Mb/s}$, and $b = 0.04$. At low load, the system is often empty, but when it is not empty, each link is quite likely to be the bottleneck. At higher load, the law of large numbers causes the ratio n/m to become more constant, and so the bottleneck does not shift so often.

7. Conclusions

This paper has provided a study of bi-directional FAST flows. We have derived the throughput of two-way FAST and link utilization in a network consisting of a single bidirectional link. Either direction of this link may be the bottleneck, giving rise to two distinct but symmetric scenarios.

The results have been validated by simulations in several example cases. Further examples demonstrated that balanced parameters leads to higher network efficiency. We also argued that reverse queueing need not fundamentally change the stability issues associated with FAST in simple topologies.

Finally, we have demonstrated that the fraction of time for a link being bottlenecked depends on the total load, as well as the amount of traffic on each path.

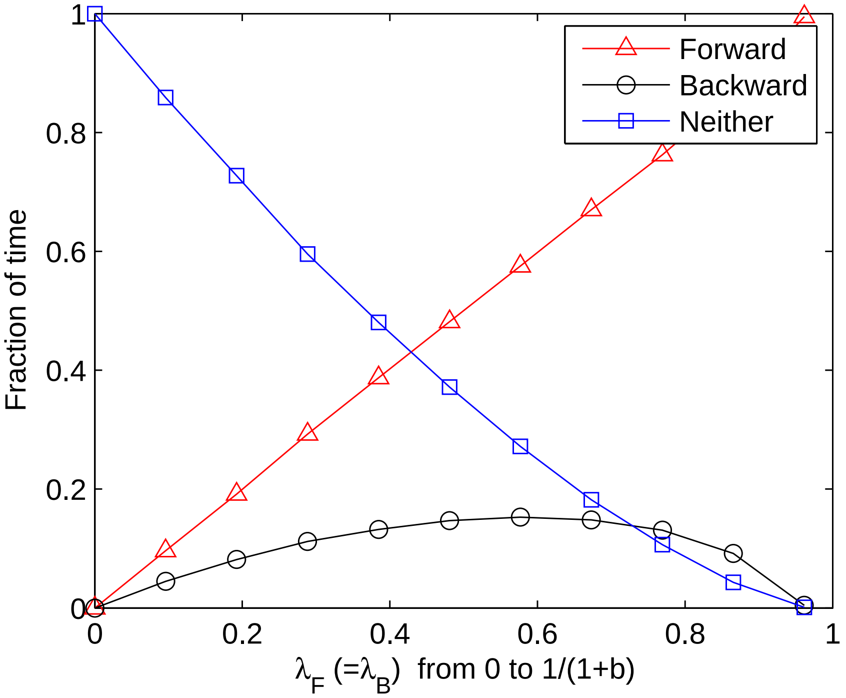

Figure 5: Fraction of time l_F is bottlenecked, l_B is bottlenecked, and neither is bottlenecked (i.e., the system is empty).

8. Acknowledgements

The work described in this paper was supported by a grant from City University of Hong Kong (Project No. 9380044), the National Natural Science Foundation of China (no.61070197), the self-determined research funds of CCNU from the college's basic research and operation of MOE in China, and the Australian Research Council (ARC grants DP0985322 and FT0991594). Part of the work described in this paper was done during the one-year visit (2009-2010) of F. Ge at City University of Hong Kong which was supported by project no. 9380044.

References

[1] S. H. Low, L. Peterson, L. Wang, Understanding Vegas: A duality model, Journal of ACM 49 (2) (2002) 207–235.

[2] D. X. Wei, C. Jin, S. H. Low, S. Hegde, FAST TCP: Motivation, architecture, algorithms, performance, IEEE/ACM Trans. Netw. 14 (6) (2006) 1246–1259.

[3] K. Tan, J. Song, Q. Zhang, M. Sridharan, A compound TCP approach for high-speed and long distance networks, in: Proc. IEEE INFOCOM 2006, Barcelona, Spain, 2006, pp. 1–12.

[4] C. Jin, D. Wei, S. H. Low, J. Bunn, H. D. Choe, J. C. Doylle, H. Newman, S. Ravot, S. Singh, F. Paganini, G. Buhrmaster, L. Cottrell, O. Martin, W. Feng, FAST TCP: from theory to experiments, IEEE Netw. 19 (1) (2005) 4–11.

[5] J. Wang, D. X. Wei, S. H. Low, Modelling and stability of FAST TCP, in: Proc. IEEE INFOCOM 2005, Vol. 2, Miami, FL, 2005, pp. 938–948.

[6] A. Tang, L. L. H. Andrew, K. Jacobsson, K. H. Johansson, S. H. Low, H. Hjalmarsson, Queue dynamics with window flow control, IEEE/ACM Trans. Netw. 18 (5) (2010) 1422–1435.

[7] K. Jacobsson, L. L. H. Andrew, A. Tang, S. H. Low, H. Hjalmarsson, An improved link model for window flow control and its application to FAST TCP, IEEE Trans. Autom. Control 54 (3) (2009) 551–564.

[8] L. Tan, C. Yuan, M. Zukerman, FAST TCP: fairness and queuing issues, IEEE Commun. Lett. 9 (8) (2005) 762–764.

[9] L. Kalampoukas, A. Varma, K. K. Ramakrishnan, Two-way TCP traffic over rate controlled channels: effects and analysis, IEEE/ACM Trans. Netw. 6 (6) (1998) 729–743.

[10] L. X. Zhang, S. Shenker, D. D. Clark, Observations on the dynamics of a congestion control algorithm: the effects of two-way traffic, SIGCOMM Comput. Commun. Rev. 21 (4) (1991) 133–147.

[11] H. Bullot, R. L. Cottrell, R. Hughes-Jones, Evaluation of advanced TCP stacks on fast long-distance production networks, Journal of Grid Computing 1 (2003) 345–359.

[12] H. Balakrishnan, V. N. Padmanabhan, How network asymmetry affects TCP, IEEE Commun. Mag. 39 (4) (2001) 60–67.

[13] K. T. Truong, S. Weber, R. W. Heath, Transmission capacity of Two-Way communication in wireless Ad Hoc networks, in: Proc. IEEE ICC 2009, Dresden, Germany, 2009, pp. 1–5.

[14] E. Lopez-Aguilera, M. Heusse, Y. Grunenberger, F. Rousseau, A. Duda, J. Casademont, An asymmetric access point for solving the unfairness problem in WLANs, IEEE Trans. Mobile Comput. 7 (10) (2008) 1213–1227.

[15] T. V. Lakshman, U. Madhow, B. Suter, TCP/IP performance with random loss and bidirectional congestion, IEEE/ACM Trans. Netw. 8 (5) (2000) 541–555.

[16] J. S. Sun, S. Chan, K. T. Ko, G. R. Chen, M. Zukerman, Instability effects of two-way traffic in a TCP/AQM system, Comput. Commun. 30 (10) (2007) 2172–2179.

[17] C. P. Fu, S. C. Liew, A remedy for performance degradation of TCP Vegas in asymmetric networks, IEEE Commun. Lett. 7 (1) (2003) 42–44.

[18] F. Ge, L. Tan, Improving FAST TCP performance in asymmetric networks, in: Wireless Broadband and Ultra Wideband Communications, 2007. AusWireless 2007. The 2nd International Conference on, Sydney, NSW, Australia, 2007, pp. 56–56.

[19] F. Ge, L. Tan, M. Zukerman, Throughput of FAST TCP in asymmetric networks, IEEE Commun. Lett. 12 (2) (2008) 158–160.

[20] F. P. Kelly, A. K. Maulloo, D. K. H. Tan, Rate control for communication networks: shadow prices, proportional fairness and stability, Journal of the Operational Research Society 49 (9) (1998) 237–252.

[21] S. H. Low, A duality model of TCP and queue management algorithms, IEEE/ACM Trans. Netw. 11 (4) (2003) 525–536.

[22] J. Wang, D. X. Wei, J.-Y. Choi, S. H. Low, Modelling and stability of Fast TCP, in: P. Agrawal, P. J. Fleming, L. Zhang, D. M. Andrews, G. Yin (Eds.), Wireless Communications, Vol. 143 of The IMA Volumes in Mathematics and its Applications, Springer New York, 2007, pp. 331–356.

[23] The network simulator: ns2 [online] (2005). http://www.isi.edu/nanam/ns/.

[24] T. Cui, L. L. H. Andrew. FAST TCP simulator module for ns-2 [online] (2005). version 1.1.

[25] F. H. P. Fitzek, M. Reisslein, MPEG-4 and H.263 video traces for network performance evaluation, IEEE Netw. 15 (2001) 40–54.

[26] L. Massoulié, J. W. Roberts, Bandwidth sharing and admission control for elastic traffic, Telecommunication Systems 15 (2000) 185–201.

[27] T. Bonald, L. Massoulié, Impact of fairness on internet performance, in: Proc. ACM Sigmetrics 2001, Cambridge, MA, 2001, pp. 82–91.

[28] F. Paganini, A. Tang, A. Ferragut, L. L. H. Andrew, Network stability under alpha fair bandwidth allocation with general file size distribution, IEEE Trans. Autom. Control (submitted).

[29] X. Lin, N. Shroff, R. Srikant, On the connection-level stability of congestion-controlled communication networks, IEEE Trans. Inf. Theory 54 (5) (2008) 2317–2338.