Do Current Fluid Approximation Models Capture TCP Instability?

Daniel Genin and Vladimir Marbukh, NIST
Email: dgenin@nist.gov, marbukh@nist.gov

***Abstract*—We compare unstable parameter regions of a fluid approximation model of TCP and the corresponding *ns2* simulations. We identify the parameter region corresponding to unstable equilibria of the fluid approximation model. Loss of stability by the equilibrium typically leads to appearance of periodic solutions. Simulations indicate that the aggregate TCP stream also has two distinct regimes which can be described as stochastic (stable) and oscillatory (unstable). The geometry of the unstable regime parameter region disagrees substantially with the unstable parameter region of the fluid approximation model. The reason for the disagreement appears to be that the oscillations in the fluid approximation model and in simulations have fundamentally different origins. We argue that a serious revision of the fluid approximation model is necessary if it is to accurately capture non-equilibrium TCP dynamics, such as periodic solutions.**

I. Introduction

Fluid approximation models of TCP have been studied extensively over the last 20 years. There are clear practical benefits to having good mathematical models of TCP. Even relatively simple networks with large numbers of sources can take a long time to simulate on reasonable time scales. Simulating multiple sets of networks conditions, involving combinations of varying buffer sizes, link bandwidths, and propagation delays, to find the optimal network parameters is prohibitively time consuming. Therefore, accurate computationally tractable network models would be a great boon to network engineers as well as computer scientists designing networking protocols. Fluid approximation models of the TCP network protocol, which is responsible for carrying the bulk of Internet traffic today, are prototypical examples of this line of research. In spite of the substantial literature on the equilibrium properties and stability of these models, however, there still appears to be little understanding of how accurate they are, particularly when it comes to stability, or even exactly what regime they model.

In this paper we compare one of the commonly used fluid approximation models against *ns2* simulation data with focus on the question of equilibrium stability. This fluid approximation model of TCP was originally introduced by Kelly et al. [2]. Since its introduction the model has been extensively used to model multiple TCP sources in networks with arbitrary topologies and heterogeneous delays (for example [8],[3],[5]). It is well known that the model has a unique equilibrium, which is easy enough to compute. Local and global stability of the equilibrium have also been extensively studied. Feedback propagation delay makes even linear stability analysis of the equilibrium a non-trivial task and as a result the linear stability condition for the equilibrium solution [8] has an implicit form with respect to the two model parameters — bandwidth-delay product and buffer size. This makes it difficult to determine exactly which network parameters correspond to stable equilibria.

In order to obtain a more explicit stability condition we rewrite the stability inequality of [8] in a way that makes it clear that it is the ratio of the equilibrium free buffer space to band-width delay product that determines the linear stability of the fluid model. Using the new form of the linear stability condition we prove that the equilibrium is linearly stable regardless of the bandwidth-delay product provided that the buffer is sufficiently small (less than 18 packets according to our computations). Conversely, the equilibrium is linear stable regardless of the buffer size provided the bandwidth-delay product is small (less than 3 packets). In general, the equilibrium of the fluid approximation model considered is linearly stable if the bandwidth-delay product is large compared to the buffer size.

With these results in hand we move on to *ns2* simulations. In order to compare theoretical predictions with simulation results it is important to understand which limiting regime the fluid approximation model represents. The consensus in the literature is that it models the mean aggregate rate in the limit of infinitely many flows with identical (or nearly identical) round-trip times [6]. Rigorous results in this direction, however, are scarce and make strong assumptions about flow statistics [7], [6], [1]. Taking the correctness of this view on faith, for the moment, we compare the results of *ns2* simulations with fluid approximation model predictions and find substantial disagreement in the geometry of the stable parameter region.

Simulations were run with 1000 flows transiting a single router, with round trip times uniformly distributed within $\pm 5\%$ interval about the mean. Buffer size and propagation delay were varied to determine the affect of these parameters on throughput and packet loss. To summarize, the simulation data indicate that buffer size more than bandwidth-delay product determines the regularity and amplitude of oscillations in transmission rate. Qualitatively the picture is somewhat similar to the results on the fluid approximation model, in that there appears to be a critical buffer size below which fluctuations in transmission rate are largely random and uncorrelated and above which the fluctuations are nearly perfectly periodic. This critical buffer size depends albeit weakly on the bandwidth-delay product and likely on the number of flows, though we did not test the dependence on the later. Assuming that the oscillations in transmission rate observed in simulations are the result of the limit cycle appearing in the fluid approximation model there is a significant disagreement in the parameter values where the two appear. The greatest surprise, however, stemming from the simulation data was non-monotonicity of packet loss with respect to buffer size. This in our view is the clearest indication yet of the presence of multiple dynamic regimes, already observed by [4], and of the limitations of the considered fluid approximation model.

The above discrepancies raise the question of what is it that the model is missing and whether there may be some parameter domain where the model is valid.

The answer to the first question appears to be loss synchronization. As observed in [4], the instability in TCP congestion control manifests itself as synchronization between flows, which in turn, causes periodic oscillations in transmission rate. Loss synchronization means that a large proportion of flows experiences nearly simultaneous packet loss in a single buffer busy period. It is caused by the interaction between flows at the router yet no fluid approximation model takes this inter-flow interaction explicitly into account. The standard assumption in all fluid approximation convergence results is that losses for different flows occur independently and the arrival stream at the buffer is a sum of Poisson processes. Which in turn allows us to conjecture an answer the second question. The conditions of loss independence and Poisson arrivals are very nearly satisfied when buffer size is below the critical value at which synchronization induced oscillations set in.

The rest of the paper is structured as follows. In Section II we lay out the TCP fluid model in the detail necessary for the subsequent analysis. Section III contains theoretical results derived from the fluid approximation model, followed by simulation results in Section IV. In Section V we summarize our findings and discuss directions for future research.

II. TCP Fluid Approximation Model

TCP uses a congestion window to control the maximum number of unacknowledged packets in transit. Precise dynamics of the congestion window size are notoriously hard to capture. For this reason a fluid approximation mirroring the main features of TCP congestion control — additive-increase multiplicative-decrease (AIMD) — is commonly studied instead. The fluid approximation aims to model the mean transmission rate under the assumption that the number of packets transmitted in a round-trip is large. The relationship between transmission rate x and window size W is approximated by $x(t) = W(t)/T$, where T is the round-trip link delay and $W(t)$ is the congestion window size. The basic equation describing the mean transmission rate $x(t)$ is then

$$\dot{x}(t) = \frac{x(t-T)(1-p(t-T))}{T^2 x(t)} - \frac{1}{2}x(t-T)p(t-T)x(t), \quad (1)$$

where $p(t) \in [0,1]$ is the fraction of the packets lost which is interpreted as an indicator of congestion [8].

In correspondence with TCP-Reno the first term in (1) gives the additive increase of $1/W$ per acknowledged packet, while the second gives the multiplicative decrease by $1/2$ for every unacknowledged packet. To complete the model $p(t)$ must be given a specific form in terms of $x(t)$, which we postpone for the moment. In equilibrium the two competing processes of AIMD are balanced and $\dot{x}(t) = 0$. Solving (1) with the left-hand side set to zero and $x(t) = \hat{x}$ we find that the equilibrium transmission rate is given by

$$\hat{x} = \frac{1}{T}\sqrt{\frac{2(1-\hat{p})}{\hat{p}}}, \quad (2)$$

where $\hat{p} = p(\hat{x})$ is the equilibrium packet loss probability.

To complete the model we take $p(t) = p(x(t)) = (1 - x/c)(x/c)^B/(1-(x/c)^{B+1})$, which is the probability that an M/M/1/B queue with capacity c is full. Since theoretical results obtained below are to be compared against simulation data, we remark at this point that the particular loss model does not correspond to simulated environment in that the inter-arrival and, certainly, service times in simulations are not necessarily exponentially distributed. We leave the question of accuracy of the loss model for future research. The equilibrium solution now is determined by the equation

$$(1 + \hat{x}^2 T^2/2)^{-1} = p(\hat{x}).$$

We rewrite this in terms of the dimensionless quantities of load, $\rho = x/c$, and bandwidth-delay product, $L = cT$,

$$(1 + \hat{\rho}^2 L^2/2)^{-1} = p(\hat{\rho}), \quad (3)$$

where, abusing notation we now take $p(\rho) = \rho^B / \sum_{k=1}^{B} \rho^k$ as it will be through out the rest of the paper. It is likely that closed algebraic form for $\hat{\rho}$ in terms of L and B does not exist. Nevertheless, using standard methods of control theory it is possible to show [8] that the equilibrium solution is stable in linear approximation if

$$\hat{p}'/\hat{p} < \pi L/2 \quad (4)$$

where $\hat{p}' = dp/d\rho(\hat{\rho})$ and $\hat{p} = p(\hat{\rho})$. (Our model corresponds to the one considered in 5.1 of [8] with $\kappa = 1/T^2$ and $\beta = T^2/2$.) The left-hand side of (4) is a complicated function of L and B, which obscures the exact relationship between bandwidth-delay product L, buffer size B and stability of the equilibrium solution of (1).

III. Theoretical results

We begin by plotting the stability inequality (4) in the L-B plane. By a plot of the inequality we mean the plot of the set where equality holds, and which is thus the boundary of the inequality solution set. Solving for L from (3) gives $L = 1/\hat{\rho}\sqrt{2(1-\hat{p})/\hat{p}}$. Substituting this into (4) and moving all the terms depending on ρ to the left-hand side gives

$$\hat{\rho}(\hat{p}'/\hat{p})\sqrt{\hat{p}/(1-\hat{p})} < \pi/\sqrt{2}. \quad (5)$$

Plotting (5) with inequality replaced by equality in the L-B plane gives a plot as in Fig. 1. The figure suggests the following observations about the linear stability of the equilibrium solution.

1) The equilibrium solution is linearly stable for all L provided B is sufficiently small.
2) For any fixed B the equilibrium solution is linearly stable provided L is sufficiently large.

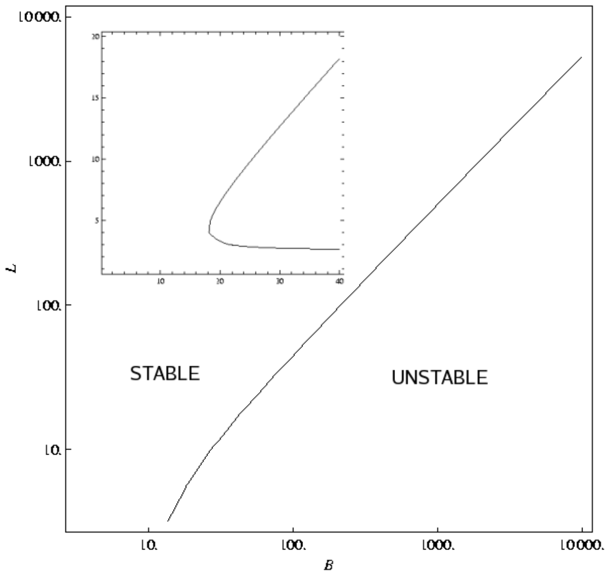

Fig. 1. Plot of (5) with inequality replaced by equality. The inset shows a blow up of the curve near the origin.

The scale of the figure hides another part of the stability region — a strip of roughly constant width that runs along the B axis. Part of this region is visible in the inset of Fig. 1. This last observation is in agreement with the common knowledge that equilibrium solution is stable for links with small enough bandwidth-delay product [8].

Next we support each of the above observations with analytic computations in the process deriving an interesting modification of the stability inequality. We begin by rewriting the linear stability condition (4) in a more useful and informative form. Derivative of p with respect to ρ can be rewritten as

$$p'(\rho) = \frac{p(\rho)}{\rho}\left(B - \frac{\sum_{k=0}^{B} k\rho^k}{\sum_{k=0}^{B} \rho^k}\right) = \frac{p(\rho)}{\rho}(B - Q)$$

where Q is the expected queue length at the buffer when the load is ρ. Substituting this into (4) we get

$$B - \hat{Q} < (\pi/2)\hat{\rho}L = (\pi/2)\hat{W} \tag{6}$$

where $\hat{W}$ and $\hat{Q}$ are the equilibrium congestion window size and buffer queue size respectively.

Stability inequality (6) shows that linear stability of the equilibrium solution is determined by the amount of the free buffer space relative to the window size in equilibrium. The reason for this appears to be that increasing buffer size drives the equilibrium load closer to 1 and simultaneously increases sensitivity of the system near $\rho = 1$ by increasing p'/p. Notice that p'/p has a singularity at $\rho = 1$ in the limit of infinite buffer size.

A. Stability for small buffer size

Estimates required to obtain a strong lower bound on the buffer size at which instability may appear are too long for the current exposition. Instead we restrict our attention to the case $B = 1$ and show that in this case the equilibrium is linearly stable regardless of the bandwidth-delay product L. If $B = 1$ then $p(\rho) = \rho/(1+\rho)$ and $p'(\rho) = 1/(1+\rho)^2$. Substituting this into (3) gives $\hat{\rho} = 2^{1/3}/L^{2/3}$. Now stability inequality (4) can be rewritten solely in terms of L as

$$L^{1/3}/(2^{2/3} + (2L^2)^{1/3}) < \pi/2$$

The maximum of the left-hand side is attained at $L = \sqrt{2}$ and so is $1/2\sqrt{2}$ which is less than $\pi/2$. Hence the equilibrium solution is linearly stable for any bandwidth-delay product L.

A much better estimate can be obtained by expanding the left-hand side of the stability inequality (5) in a Taylor series about $\rho = 1$ but the error estimates are quite cumbersome. This more precise estimate gives $B^* = 18$ as the critical buffer size at which instability may appear. The tip of the "nose" in the inset of Fig. 4 corresponds to B^*.

B. Stability for large bandwidth-delay

Theorem 1. *The equilibrium solution is stable in linear approximation for any fixed B if L is sufficiently large.*

Proof: First, observe that $\hat{p} \to 0$ as $L \to \infty$. This can be seen by considering solution $\hat{\rho}$ of the equilibrium equation (3) as the intersection of the graphs of the right and left-hand sides with respect to ρ. The right-hand side is a monotonically increasing function of ρ and approaches 0 as ρ tends to 0. The left-hand side is a monotonically decreasing function of ρ which converges to zero point-wise as L tends to infinity. Since the right-hand side does not depend on L it follows that the intersection point of the two graphs converges to 0 as L tends to infinity.

Next, we evaluate the right-hand side of (6) as L tends to infinity. Multiplying both sides of equilibrium equation (2) by T gives $\hat{W} = \sqrt{2}\sqrt{(1-\hat{p})/\hat{p}}$. Since $\hat{p} \to 0$ as $L \to \infty$, $\hat{W}$ diverges as L increases. The left-hand side, on the other hand, is bounded above by B. Hence the equilibrium solution will be linearly stable for all sufficiently large L. ■

To test whether instability is really intermittent in L we solved the delay differential equations (1) using NDelayDSolve Mathematica package. The stability criterion, after all, is only a sufficient stability condition and does not say anything about existence of unstable equilibrium solutions. Fig. 2 shows plots of numerical solutions to (1) for $B = 25$ and increasing values of L. The low and the high values of L (plots (a) and (c)) converge to steady state equilibria, while the plot for the intermediate value of L (plot (b)) settles down to oscillations about the equilibrium, indicating that the equilibrium solution is not stable.

IV. Simulation Results

Supposing that the fluid approximation models transmission rate in the limit of infinite number of identical flows, we compare the above theoretical results with *ns2* simulation data. Experimenting with different numbers of flows we converged on 1000 as a number that is large enough for aggregate behavior to emerge but at the same time small enough that

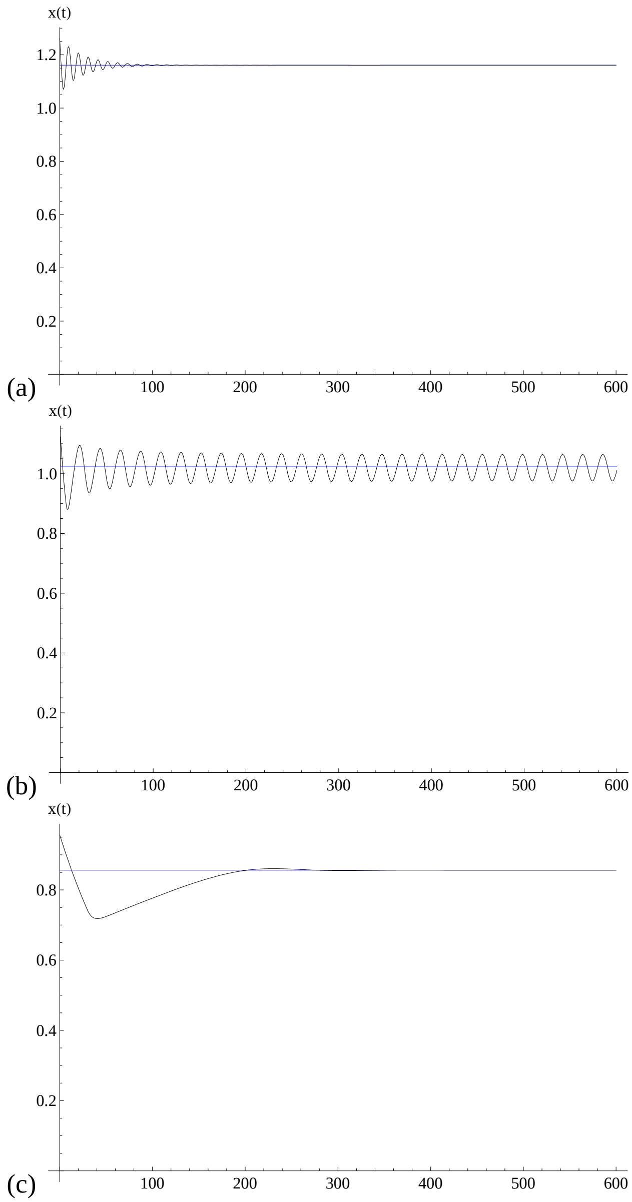

Fig. 2. Numerical solutions of (1) with $B = 25$ initial condition $\rho(t) = \hat{\rho} + .1$ on $[-T, 0]$ and (a)$L = 3$ (b)$L = 6$, (c)$L = 30$. The equilibrium solution is shown in blue.

a simulation can be run in reasonable time (on average a few hours for 100s of simulated time).

The network topology used in simulations was is in Figure IV. The propagation delays of the access links are randomly uniformly distributed in the range between 0 and about 10% of the propagation delay on the main link $n1$ to $n0$. This relatively small random component is added to the main propagation delay on the $n1$-$n0$ link to avoid periodicity that develops when packet transmission times lock into a fixed pattern. Additional randomization is introduced by starting flows at random times uniformly distributed within the first second of the simulation. The router capacity was set to 1Gb/s so that a typical congestion window size was large enough to

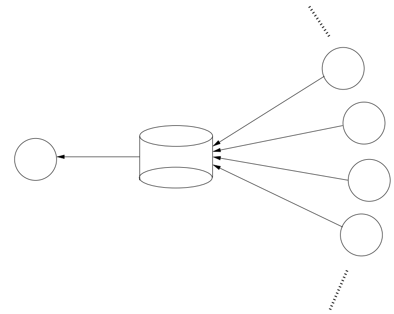

Fig. 3. Network layout.

make the fluid approximation reasonable. This left precisely two parameters — the propagation delay and buffer size of the $n1$-$n0$ link, which correspond to T and B parameters of the fluid model 1 respectively. Each simulation ran for 100s of simulated time but the first 10s were then discarded to avoid contamination of data with transient effects.

Figures 4 summarize the simulation data. The plots show throughput, goodput and packet loss as functions of buffer size B for values of the propagation delay T ranging between 100ms and 500ms in 100ms increments.

The packet loss plot immediately stands out because of the counter-intuitive non-monotonic shape of the loss curves. Standard queuing theory intuition would suggest that the packet loss should decrease exponentially with increasing buffer size, and this is indeed what happens when buffer sizes are relatively small. TCP congestion control mechanism, however, adjusts the transmission rate based on packet loss feedback which throws a wrench into the gears of the . For each fixed propagation delay the decreasing packet loss trend reverses at some critical buffer size $B_c(T)$ and the packet loss increases over a range of buffer sizes before resuming its downward trend. What is the cause of this odd behavior? Looking more closely at the data, by examining the time series of the mean aggregate transmission rate (measured as the packet arrival rate at the router) for buffer sizes less than and greater than $B_c(T)$ (Figure 5) it is easy to see the likely proximate cause of the reversal. The obvious difference between the two time series is the presence of periodic oscillations for buffer sizes greater than $B_c(T)$.

The reason for this dichotomy is that the system operates in two fundamentally different regimes depending on whether $B < B_c$ or $B > B_c$.When $B < B_c$ the buffer is not big enough to accommodate the bursts arising on the short time scale from "piling up" of packets due to the differences in the round-trip times and back to back packets resulting from congestion window increases. This means that losses occur before the aggregate source rate saturates the link, i.e. reaches the capacity of the router. A large proportion of flows is, thus, unlikely to suffer simultaneous losses because the buffer can begin to clear immediately following the burst. The loss events remain largely independent within and across flows on the long time scale.

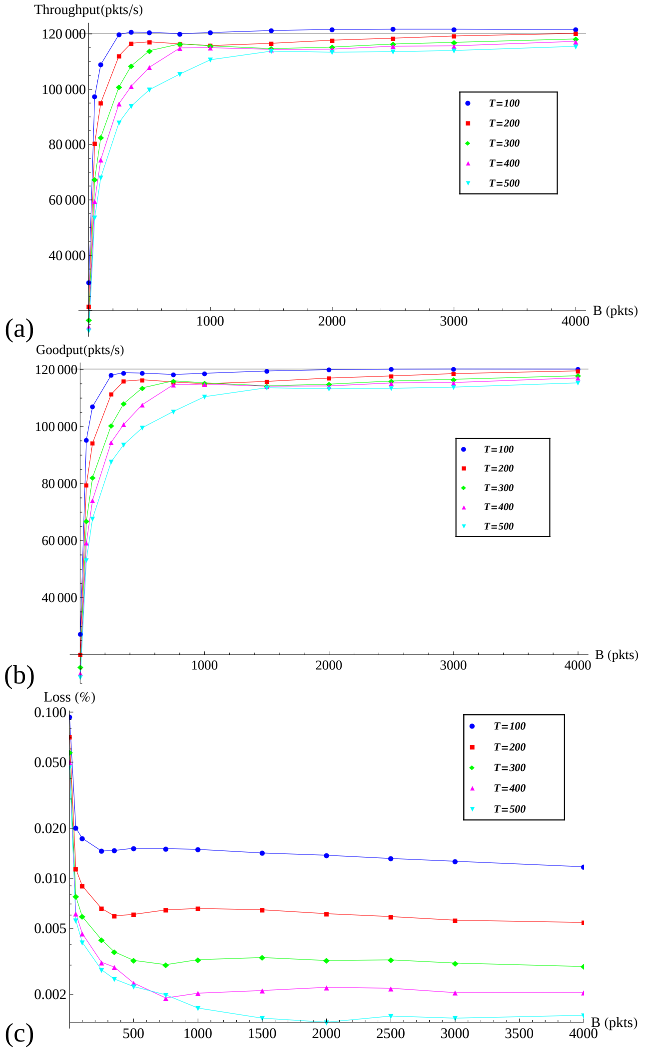

Fig. 4. Aggregate throughput(a), goodput(b), and fraction of lost packets(c) vs buffer size B and propagation delay T.

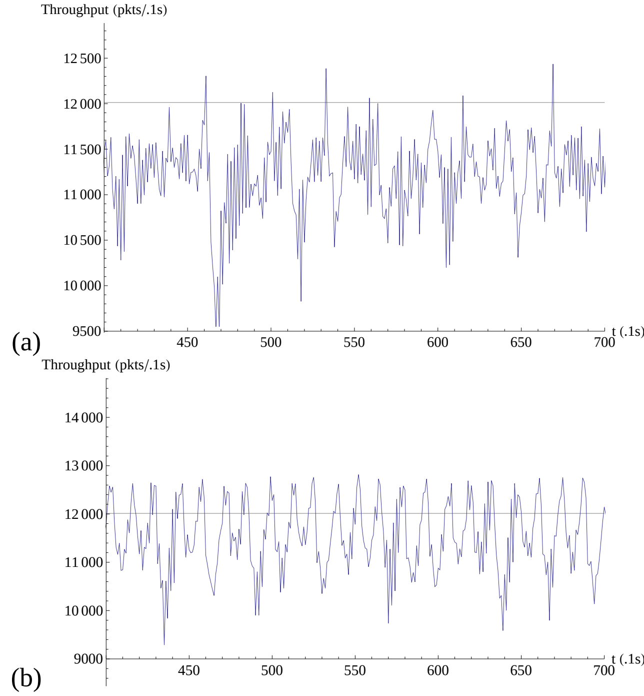

Fig. 5. Aggregate throughput over time for T=200ms and B=250pkts (a) and B=1000pkts (b). The horizontal line indicates the router capacity.

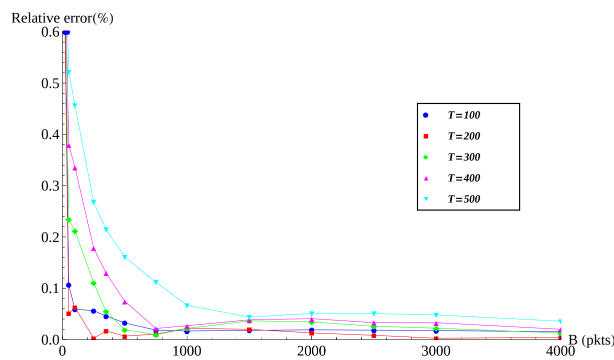

Fig. 6. Relative error in the fluid approximation equilibrium throughput.

If $B > B_c$ the buffer is big enough to absorb short time scale bursts and the sources can saturate the link. This increases the length of buffer busy periods because the buffer usually will not be able to start clearing until all the packets that have entered the link following the saturation point have left the system by either being processed or dropped. This number, of course, can not be greater than the bandwidth-delay product plus buffer size and so the buffer busy period cannot exceed $T + B/c$. The longer the busy period the more flows are likely to lose packets during it. When a large number of flows experience simultaneous losses a drastic drop in the aggregate transmission rate occurs as the flows cut their congestion windows in sync. Since the aggregate congestion window cannot exceed the bandwidth-delay product by more than B, which in our simulations is small compared with the bandwidth-delay product, after the decrease the aggregate congestion window will be significantly smaller than the bandwidth delay product and the aggregate transmission rate significantly smaller than router capacity. With this the buffer will finally begin to clear and then the cycle repeats.

Increasing the buffer size increases the duration of the above cycle because larger buffers take longer time to fill. Since each flow on average increases its congestion window by 1 packet per round trip time the buffer takes roughly TB/N seconds to fill, where N is the number of flows (1000 in our case). So the period between consecutive buffer overflows is proportional to the buffer size. At the same time the duration of the busy period remains relatively constant, at least if $B/c << 1$. Therefore, as the number of packets sent in a cycle grows compared with the number of packets dropped in during the busy period the fraction of lost packets starts to decline again in agreement with graphs in Figure 4(c).

Comparing these results with fluid approximation model

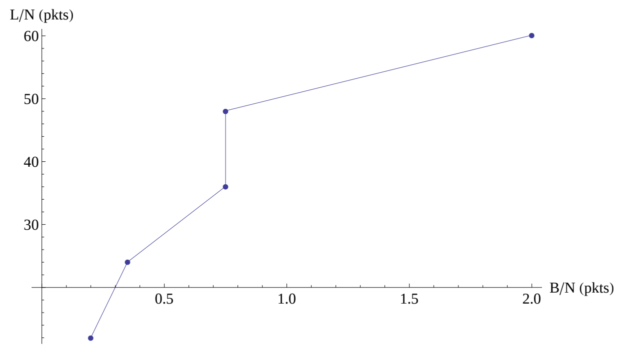

Fig. 7. Critical buffer curve.

predictions we see that wherethe equilibrium throughput is concerned the model 1 does relatively well when the bandwidth-delay product is small and the buffer size is large (Figure 6). This is somewhat surprising because the model clearly does not account for synchronization effects and, conversely, the lack of correlation between flows for small buffers seems to be exactly what the assumptions of the model require. We will discuss the likely reasons for this discrepancy and propose some modifications to improve the fidelity of the model briefly, but first we turn our attention to the main objective of this paper — instability.

If we take the $B < B_c$ regime to be stable and $B > B_c$ to be unstable, then the plot of $B_c(T)$ shows, that as regards instability, the agreement between theory (Figure 1) and experiment (Figure IV) is very far from perfect indeed. Although, there is some general qualitative agreement between the two graphs the degree of quantitative disagreement raises the question whether this agreement is anything more than pure coincidence. It becomes doubly unlikely when one considers that the parameter region where the fluid approximation makes the most accurate equilibrium predictions is actually the one where it is least likely to be dynamically correct because it does not take into account the effects of loss synchronization. And this the key point: the oscillations in fluid approximation model and the oscillations observed in *ns2* simulations appear to have different origins. The former caused by the interplay of delay and buffer size the later by loss synchronization.

V. Conclusion

We deduced the parameter region corresponding to linearly unstable equilibria of the fluid approximation model. Interpreting the fluid approximation as the infinite-flow limit we compare theoretical predictions with simulation results for a range of parameter values. Simulation results indicate that TCP congestion control has two distinct operating regimes whose choice is determined by the buffer size. Since one of the regimes exhibits periodic oscillations while the other does not, it is reasonable to consider them as unstable and stable respectively. The critical buffer size separating the two operating regimes varies with bandwidth-delay product but not in the way predicted by the fluid approximation model, which makes predictions of instability derived by linear analysis from fluid approximation model questionable. The main cause of oscillations in the aggregate TCP transmission rate in simulations is loss synchronization between flows which is not modeled by the fluid approximation model we consider or any other fluid approximation model, as far as we know. This suggests that if the fluid approximation is to be used for more than computing the equilibrium throughput this collective behavior must be included in the model.

References

[1] F. Baccelli, D. R. McDonald, and J. Reynier. A mean-field model for multiple tcp connections through a buffer implementing red. *Perform. Eval.*, 49(1-4):77–97, 2002.

[2] Tan DKH Kelly FP, Maulloo AK. Rate control for communication networks: shadow prices, proportional fairness and stability. *J. Oper. Res. Soc.*, 49:237–252, 1998.

[3] R. Srikant L. Ying, G. Dullerud. Global stability of internet congestion controllers with heterogeneous delays. *Transactions on Networking*, 14(3):579–591, 2006.

[4] G. Raina and D.J.Wischik. Buffer sizes for large multiplexers: Tcp queueing theory and instability analysis. *Next Generation Internet Networks*, pages 173–180, 2005.

[5] J. Wang S. Low, F. Paganini and J. Doyle. Linear stability of tcp/red and scalable control. *Computer Networks Journal*, 43(5):633–647, 2003.

[6] R.Srikant S. Shakkottai. How good are deterministic fluid models of internet congestion control. *INFOCOMM*, 2002.

[7] R.Srikant S.Deb, S.Shakkotai. Stability and convergence of tcp-like congestion controllers in a many-flows regime. *INFOCOMM*, 2003.

[8] Srikant. *Mathematics of Internet Congestion Control*. Willey, 2001.