Impact of Network Buffers on TCP Start-Up

Chadi BARAKAT
INRIA Sophia Antipolis - FRANCE

Chadi.Barakat@sophia.inria.fr
http://www.inria.fr/mistral/personnel/Chadi.Barakat

IEEE Middle East Workshop on Networking
Thursday, November 4 1999

Organization of the talk

– Introdution
  * Overview of TCP.
  * Why do network buffers impact TCP performance?
  * A factor ignored in research works on improving TCP Start-Up.
– Our mathematical model to study this impact.
– Analytical and simulation results (`ns`).
– Conclusions and future work.

Overview of TCP

– A reliable window-based ACK-clocked transport protocol.

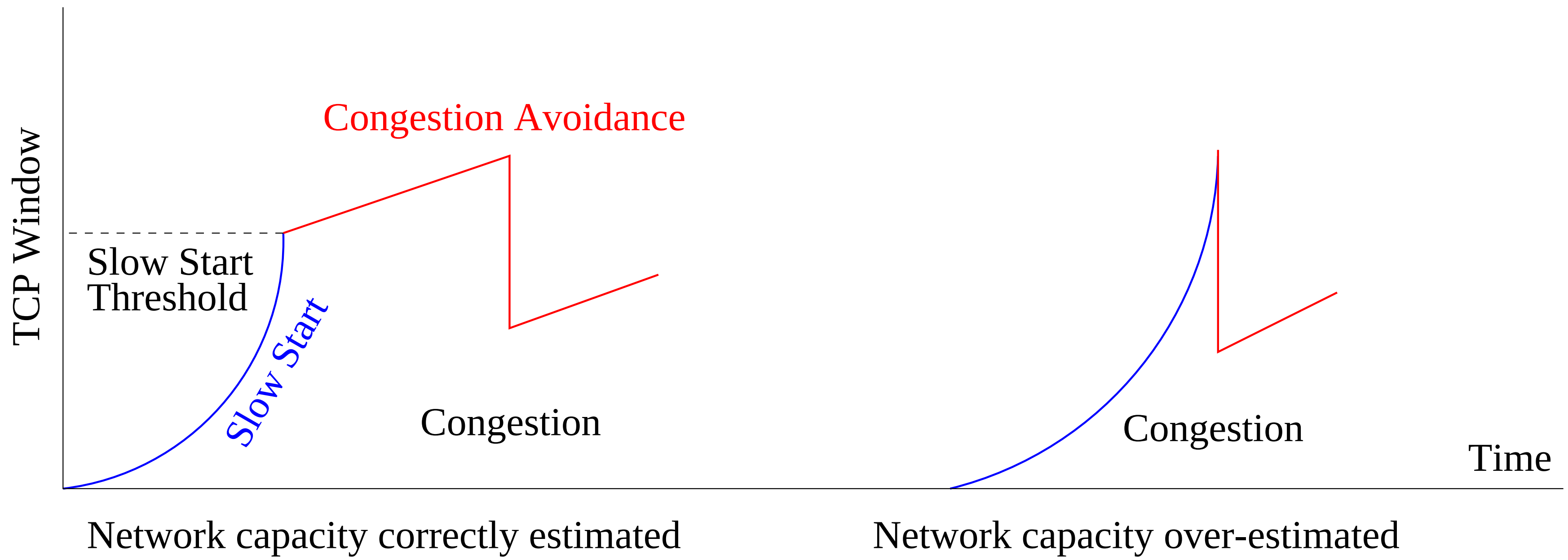
– First, a *Slow Start* phase to increase quickly the congestion window until an estimate of the network capacity (Slow Start threshold).

– Second, *Congestion Avoidance* to probe slowly the network for any extra bandwidth.

– A packet loss means a congestion and results in window reduction.

– A loss during Slow Start means that the network capacity has been overestimated. Slow Start serves then as a quick means to obtain a more accurate estimate.

Overview of TCP



Main problem of TCP Start-Up: Slow Start

– Objective of Slow Start: Fill quickly the network pipe.

– In the presence of the ACK clock, a fast window increase during Slow Start leads to burstiness, to an overload on network buffers, to losses and to an underestimation of the network capacity.

– And a slow window increase leads to poor performance due the low bandwidth utilization during Slow Start.

– The rate chosen by Jacobson is an increase by one packet for every ACK.

– But, who says that this rate is enough for all buffer sizes?

Other problems of Slow Start

- The window increase rate is inversely proportional to the round-trip time.
  Poor performance is noticed in satellite networks and on transoceanic paths.
  The *intuitive* solution is to accelerate the window growth.
  But, accelerating the window growth (e.g. Byte Counting) without accounting for buffers may deteriorate the performance instead of improving it.

- A congestion of the network during Slow Start has been shown to cause many losses and a long Timeout ($\simeq 750$ms).
  The appropriate solution is to set correctly the Slow Start threshold.
  A value equal to the *bandwidth-delay product* has been proposed.
  But again, setting it without accounting for buffers may not avoid losses.

Chadi BARAKAT
INRIA Sophia Antipolis - FRANCE

Objectives of our study

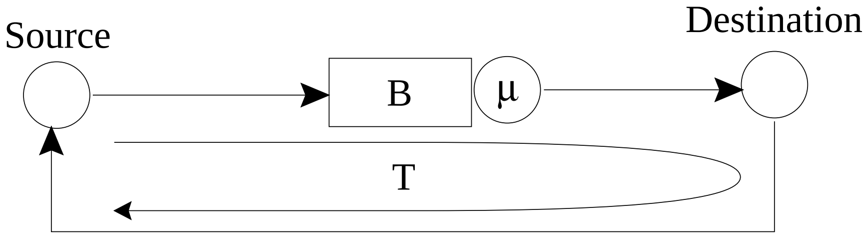
– An improvement of the Slow Start phase is now more required than before given the increase in link speeds and round-trip times.

– But, network buffers may be unable to absorb an increase in burstiness (limitation on buffer size, need for a small queueing time).

– We try to answer two main questions; given a certain bottleneck bandwidth and a certain buffer size in the network:

  * How fast to increase the window during Slow Start?
  * How to set the Slow Start threshold in order to avoid losses?

The model



The key point is to calculate the window at which B overflows during Slow Start.

Denote this window by W_B and denote the Slow Start threshold by W_{th}.

– The window growth rate during Slow Start can be increased until W_B becomes less than the pipe size.

– The condition to avoid losses during Slow Start is $W_{th} < W_B$.

The model

Let $W(n)$ be the window size in packets at the end of round-trip n.

Define a factor d that models the window growth rate during Slow Start:

$$W(n+1) = W(n) + W(n)/d = \alpha W(n)$$

d can be the result of the destination acknowledging every d packets or the source increasing its window by $1/d$ packets for every ACK.

Theorem 1 *If Slow Start does not end before the occurrence of losses, B will overflow at a window*

$$W_B = B + \min(\mu T, Bd)$$

Corollaries

– A buffer larger than $\mu T/d$ is required to absorb the burstiness of Slow Start.

– A buffer smaller than $\mu T/d$ is not able to support Slow Start until the filling up of the pipe.

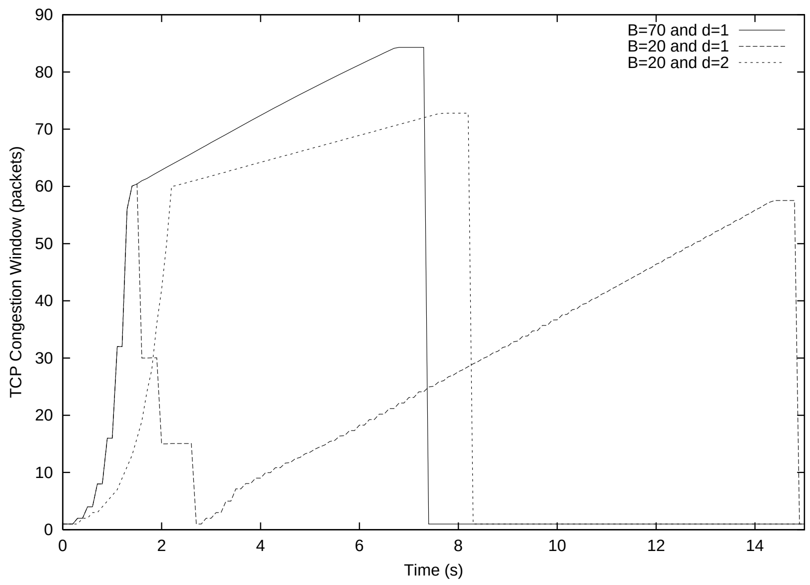
– The best performance is obtained when W_{th} is set to just less than W_B.

– For a $B < \mu T/d$, the required W_{th} is a function of B and d and independent of the bottleneck bandwidth.

– For a W_{th} equal to the bandwidth-delay product to work, B must be larger than $\mu T/(d+1)$.

A simulation example

$W_{th} = 50$ packets, BDP$= 70$ packets, packet size$= 512$ Bytes, file size$= 100$KB



How to increase the window?

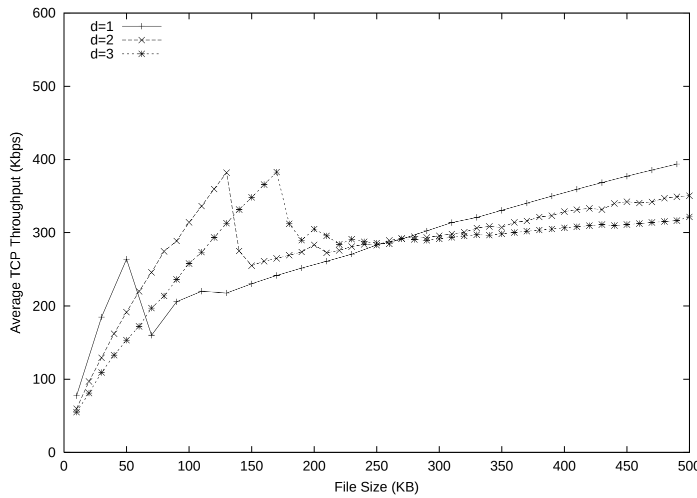
- While $B > \mu T/d$, any increase in the window growth rate improves the performance since it reduces the duration of Slow Start while not changing the overflow window.

- Once B gets below $\mu T/d$, such increase results in a buffer overflow before filling the pipe, thus in an underestimation of the available resources.
  However, the overall performance of the transfer depends on how d is implemented. Two cases to be considered:
  * Receiver-controlled d.
  * Sender-controlled d.

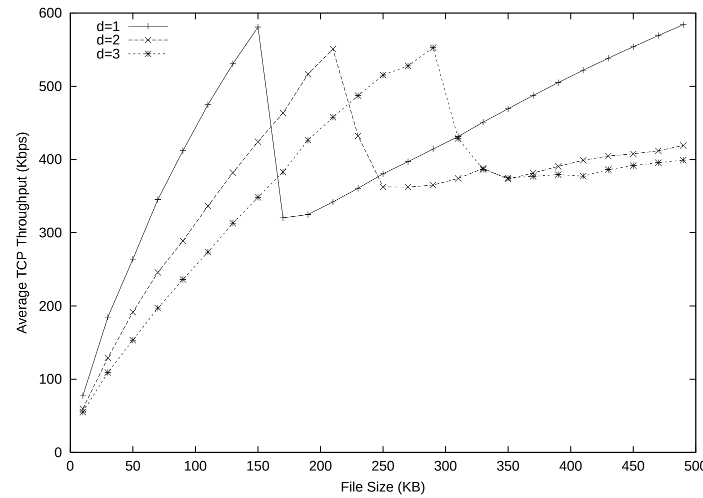
Receiver-controlled d

Affects both Slow Start and Congestion Avoidance.

The gain in performance when reducing the burstiness in case of small buffers is compensated by the slow window growth during Congestion Avoidance.



$B = 20$ packets



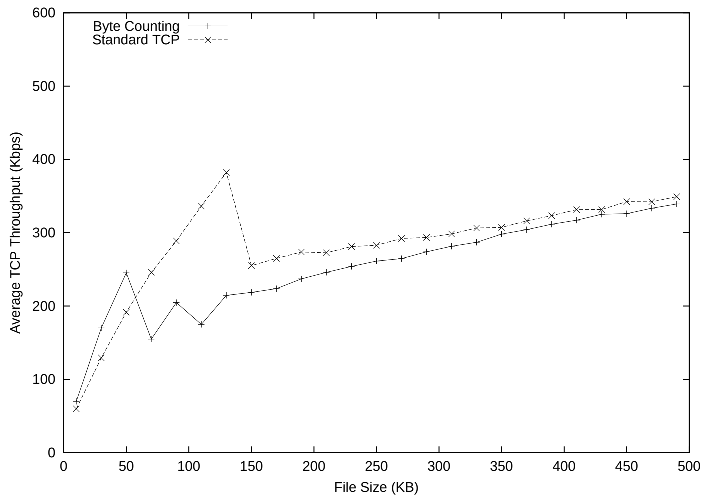
$B = 70$ packets

Impact of Network Buffers on TCP Start-Up

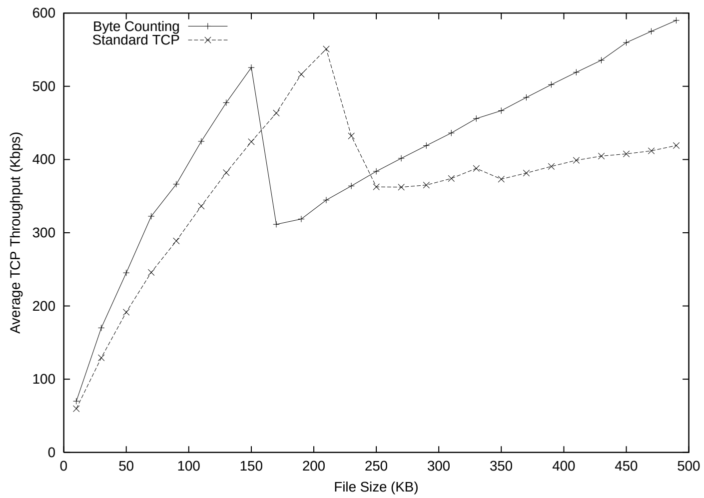
Chadi BARAKAT
INRIA Sophia Antipolis - FRANCE

Sender-controlled d

The sender is able to change the window increase policy during Slow Start without affecting Congestion Avoidance. This preserves the gain in performance obtained when reducing the burstiness in case of small buffers.



$B = 20$ packets



$B = 70$ packets

Conclusions and Future Work

– Network buffers may be unable to support Slow Start burstiness until filling the pipe. A *slower* Slow Start improves the performance in this case.

– The optimum Slow Start phase:

  * Start with the smallest possible d (the burstiest version of Slow Start).
  * Increase d when W_B is reached until filling the pipe.

– Network buffers must be considered while setting the Slow Start threshold.

– **To come:** Validation of the analysis in a more general scenario. Implementation of solutions that preserve the ACK clock and of other solutions that space packets in order to reduce Slow Start burstiness.