MODIFYING THE TCP ACKNOWLEDGEMENT MECHANISM

EVALUATION AND APPLICATION TO WIRED AND WIRELESS NETWORKS

Andrés ARCIA-MORET

December 3, 2009

OUTLINE

1 INTRODUCTION
- The Transmission Control Protocol
- Adapting the Congestion Window

2 ACK CONGESTION CONTROL (ACK-CC)
- Motivation
- State of the Art
- The ACK-CC Mechanism
- ACK-CC Evaluation
- Conclusions on ACK-CC

3 ACK DIVISION
- Definition
- Divack's Evaluation
- Conclusions on ACK Division

4 CONTRIBUTIONS

5 CONCLUSIONS
- Future Work

OUTLINE

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

OUTLINE

MOTIVATION

- TCP is omnipresent: controls about 90% of the 200000 terabytes crossing the Internet every second.
- Nowadays TCP also transports on-demand and live streaming: up to 75% of this traffic.
- Many scenarios to be adapted to.
- Up to now there is no ideal ACK frequency when network conditions change.
- We were looking for an experimental answer to the problem on adapting the ACK frequency to network congestion conditions.

BACKGROUND

- October 1986 Internet traffic overran Network Capacity.
- Mid-1987 Van Jacobson saves the Internet designing congestion control algorithms: assuring fair share.
- Some relevant IETF standardization dates we'll be dealing with:
    - 1989 → Delayed ACKs.
    - 1996 → Selective ACKs (SACKs).
    - 1999 → Slow Start, Congestion Avoidance and Fast Retransmit.
    - 1999 → Appropriate Byte Counting (ABC).
    - 1999 → New-Reno.
- We're starting our way on the IEFT by an upcoming Informational RFC about ACK congestion control.

TCP CLOCKING EXPLAINED

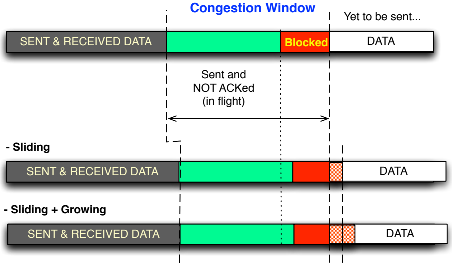


FIGURE: TCP window dynamic

- Sliding: packet conservation principle (flow control).
- Growing: adapting the transmission to network capacity (congestion control).
- Self-clocking Principle: ACKs clock out data packets to maintain the transmission rate and to discover network capacity.

TCP CONCEPTS AND MECHANISMS

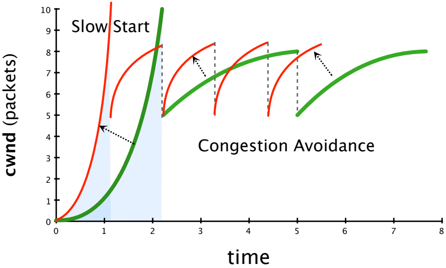


FIGURE: TCP ACK Clocking

- TCP *cwnd* depends on ACK frequency.
- Slow Start for capacity discovery.
- Congestion Avoidance to assure a fair share.
- TCP issues: burstiness and unfair share.

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

EARLY MODICATIONS TO THE ACK CLOCKING

- TCP was conceived to send 1 ACK per packet (RFC 793).
- Delayed ACKs appeared to:
  - Decrease ACK's excessive processing time.
  - Optimize interactive connections ($1/3$ of ACKs saving).
  - Improve performance in shared links (e.g., Ethernet, 802.11).

ACK CLOCKING MODIFICATIONS

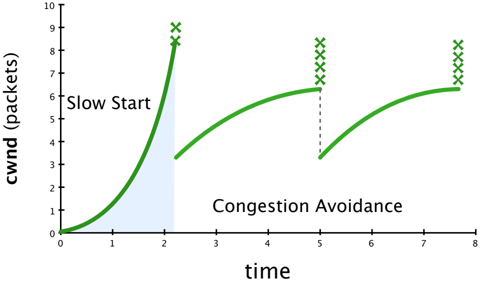


FIGURE: Slowed Down TCP window

When we slow down the ACK Rate we have:

- Reduced sending rate.
  - Slower discovery of network capacity.
  - Larger sustained rate.
- Increased burstiness.
  - Higher loss rate.

ACK CLOCKING MODIFICATIONS

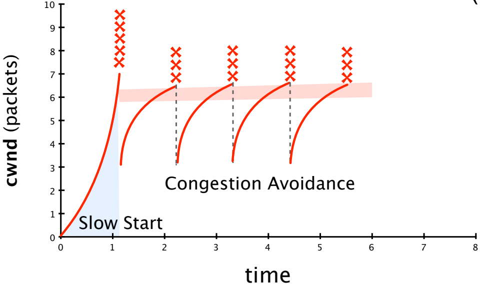


FIGURE: Fast TCP window

When we accelerate the ACK rate (if we count packets):

- Slow Start one ACK covering X packets induces burst of $X + 1$ packets.
  - A high ACK frequency → sender too aggressive.
- In Congestion Avoidance burstiness is the same as legacy 1 ACK per packet.
  - Induces much more than 1 loss per RTT.

WHAT CAN WE DO AT THE SENDER?

Counting bytes (ABC) or packets (typical implementation)?

- Byte-counting → allows uniform increase but highly bursty sender.
- Packet-counting → simplifications in implementation & widely deployed but allows ACK division.

OUR RESEARCH PROPOSAL

MODIFICATIONS ON THE ACK CLOCKING

In this thesis we propose and study methods and algorithms to understand the impact of varying the ACK sending frequency to adapt the transmission to different network configurations.

The next slides will deal with:

- There is no standard solution for ACK Congestion Control (e.g., case of asymmetric networks and wireless access).
  - We propose a end-to-end sender controlled solution to manage the ACK congestion in the constrained path.
- There is no clear answer for ACK division as a mechanism for accelerating transfers and a beleif that divacks are harmful altogether.
  - We evaluate the impact of ACK division technique in a congested network.

OUTLINE

WHEN TO CONTROL ACK FREQUENCY

- The problem
  - In asymmetric networks the uplink bandwidth is constrained.
  - The problem is exacerbated by long delay (as in satellite links).
- Design premises for a solution:
  - An end-to-end solution.
  - ACKs are not harmful altogether: they are helpful at the beginning of the transfer.
  - An optional mechanism to congestion control.

OUTLINE

1 INTRODUCTION
- The Transmission Control Protocol
- Adapting the Congestion Window

2 ACK CONGESTION CONTROL (ACK-CC)
- Motivation
- **State of the Art**
- The ACK-CC Mechanism
- ACK-CC Evaluation
- Conclusions on ACK-CC

3 ACK DIVISION
- Definition
- Divack's Evaluation
- Conclusions on ACK Division

4 CONTRIBUTIONS

5 CONCLUSIONS
- Future Work

EXISTING SOLUTIONS

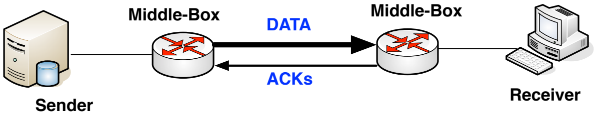


FIGURE: Reference Topology

SENDER-RECEIVER APPROACH (E2E) ACK congestion control with an ECN mark and a variable Delayed ACK factor.

RECEIVER ONLY (E2E) Estimation the cwnd from the receiver or an estimation of the congestion to calculate the appropriate ACK sending rate.

MIDDLE-BOX APPROACH ACK filtering or header compression to decongestion reverse path, ACK reconstruction to accelerate window increase.

BARAKAT & ALTMAN'S DELAYED FILTERING

Characteristics of Delayed Filtering:

- Middle-box approach for asymmetric satellite networks.
- Per ACK-flow congestion control in the reverse path.
- Uses a EWMA filter to measure of ACK-delay.
- Maintain a minimum number of ACKs per queue.
- Report important gains in web-like traffic.

OUR PROPOSAL

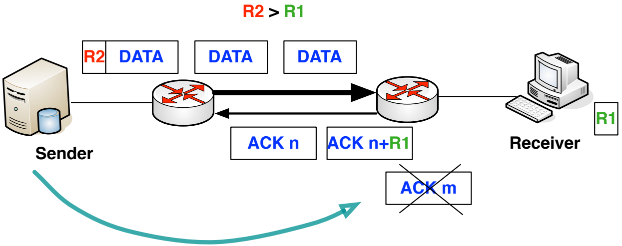


FIGURE: Sender Oriented ACK-CC

A SENDER-RECEIVER E2E APPROACH

We wanted to push the complexity to the extremes of the connexion. And we propose an ACK congestion control mechanism at the sender that controls the receiver after a continuous inspection of the ACK flow.

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

ACK CONGESTION CONTROL PRINCIPLE

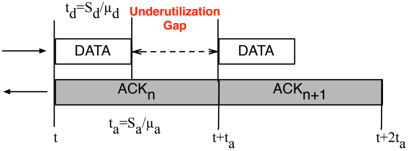


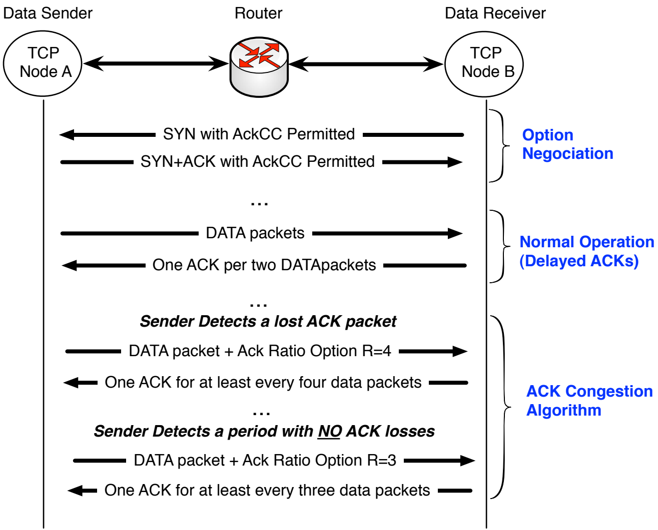
FIGURE: ACK Congestion Problem

- ACK congestion induces gaps of underutilization.
- ACK-CC closes those gaps by liberating more data packets.
- Clocking is more aggressive with fewer ACKs (just a form of aggressiveness).

SPECIFICATIONS

We propose an end-to-end mechanism that have:

- A couple of options for agreement and updating the ACK Ratio.
- A sender controlled mechanism.
- The ACK rate is roughly TCP-friendly.
- ACK sequence inspection for detecting ACK congestion.
- Sender adjust the ratio once per RTT.



SYN with AckCC Permitted

SYN+ACK with AckCC Permitted

Option Negociation

...

DATA packets

One ACK per two DATApackets

Normal Operation (Delayed ACKs)

...

Sender Detects a lost ACK packet

DATA packet + Ack Ratio Option R=4

One ACK for at least every four data packets

...

Sender Detects a period with NO ACK losses

DATA packet + Ack Ratio Option R=3

One ACK for at least every three data packets

ACK Congestion Algorithm

FIGURE: ACK-CC dynamics

ACK RATIO ADJUSTMENTS

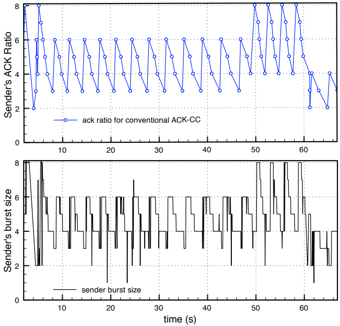


FIGURE: ACK Ratio and burst dynamics

- ACK Ratio decreases until $R = 2$ if no congestion detected.
- Oscillations in ACK Ratio control the persistent congestion.
- When ACK Ratio is allowed to decrease until $R = 2$ we call it: conventional ACK-CC.

EFFECTS OF ACK-CC

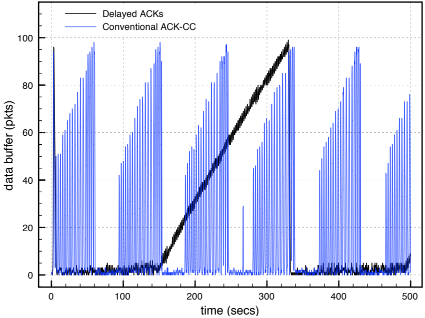


FIGURE: Sender's buffer dynamics.

- Oscillations in the ACK Ratio induce higher throughput at the sender.
- Delayed ACKs spends long time to accelerate the transmission.
- However, the underutilization can still be improved...

ACK RATIO IMPROVEMENT

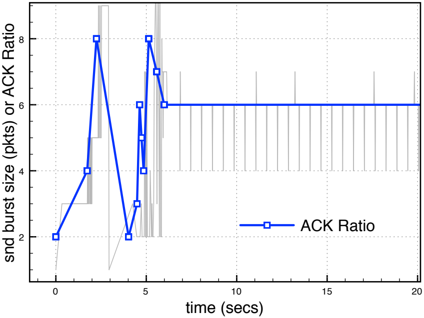


FIGURE: Limited ACK Ratio and burst dynamics

- ACK-CC tested in an asymmetric network.
- ACK Ratio increases multiplicatively, but decreases until a greater lower limit.
- Underutilization is further improved, because we adjust to a higher minimal $R \rightarrow$ gentle or limited ACK-CC.
- Just a case to illustrate the "ideal" correction.

EFFECT OF ACK-CC IMPROVEMENTS

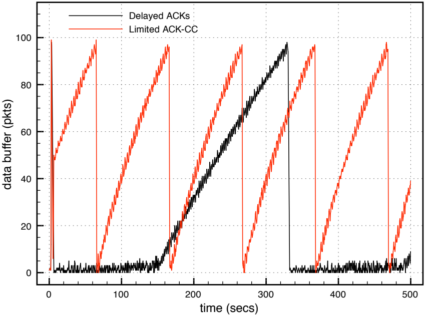


FIGURE: Improved sender's buffer dynamics.

- Less oscillations in the ACK Ratio induces even higher throughput at the sender.
- The sender becomes burstier...
- Creates more space in the reverse buffer.

OUTLINE

SIMULATION SETUP

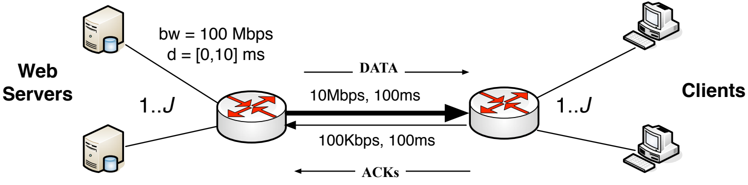
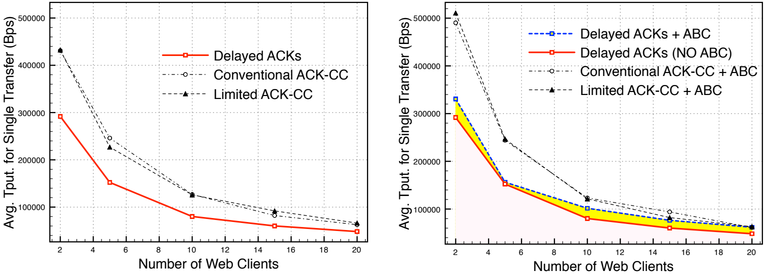


FIGURE: *Simulation Network Topology*

- Fast link for data packets, Slow link for ACKs as in satellite asymmetric topology.
- Asymmetry ratio $K = 2.66$.
- J different TCP flows traversing the bottleneck.

ACK-CC FOR WEB-TRANSFERS (1)



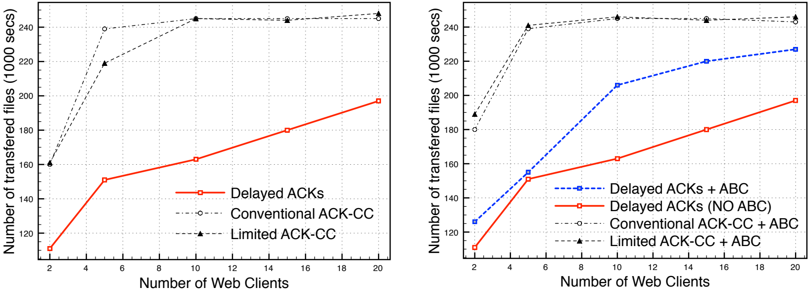
a) Average throughput per file (NO ABC)

b) Average throughput per file (**ABC**)

FIGURE: Avg. Throughput for Short Files Transfers

- ACK-CC throughput improvement.
- ABC partially compensates the throughput degradation.

ACK-CC FOR WEB-TRANSFERS (2)



a) Total transfered files file (NO ABC)

b) Total transfered files file (**ABC**)

FIGURE: Total transferred files

- The total transferred files increases.
- ABC partially compensates the throughput degradation.

ACK-CC CREATES SPACE FOR WEB-TRANSFERS

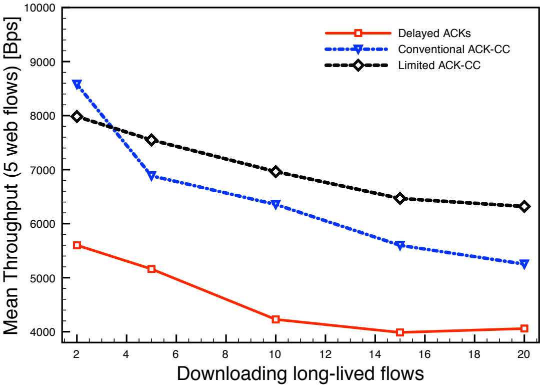
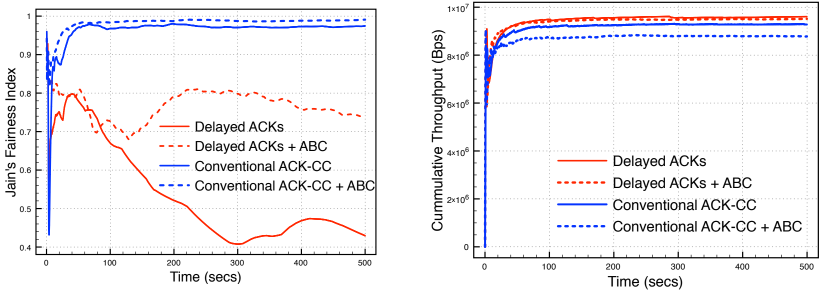


FIGURE: Average throughput per uploading web file.

- Limited ACK-CC creates more space in the reverse buffer.
- Does not considerably affect the downloading traffic.

ACK-CC for Long-lived Transfers



a) Jain's Fairness Index

b) Cummulative Throughput

Figure: 10 downloading flows in an asymmetric network

ACK-CC FOR LONG-LIVED TRANSFERS

For Long-lived transfers using ACK-CC:

- Same simulation set-up as web-like transfers.
- Long transfers occurs mostly in congestion avoidance.
- ACK-CC shows better performance at a minimal cost: 3.3% of aggregated throughput.
- Corrects severe degradation of fairness index.
- ABC compensates somehow the fairness degradation.

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

CONCLUSIONS ON ACK-CC

Through the experimental evaluation we have seen that:

- ACK-CC reduces considerably the delay induced by ACKs; by injecting enough ACKs at the beginning of the connection and reducing the ACK rate when the connexion is warmed-up.
- ACK-CC fixes-up a deformed ACK clocking by excessively delayed ACKs.
- ACK-CC improves the performance in both phases: slow start and congestion avoidance.

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

ACK DIVISION: THE OTHER SIDE OF THE COIN

How do we accelerate data transmission?

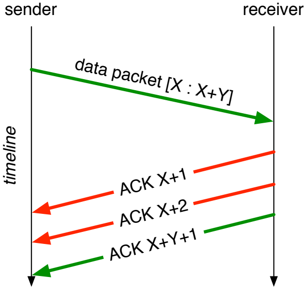


FIGURE: ACK division (divack) technique

ACK DIVISION: THE OTHER SIDE OF THE COIN

- ACK division (divack) it’s been determined as a potential threat for stealing bandwidth (Savage et al.).
- divacks has also been considered in wireless environments to recover from random losses.

AND SO WE DO ...

... Consider divacks as a complement to ACK congestion control, so we evaluate and assess its performance on a congested link.

THE MILESTONE

To what extent ACK division represents a problem when facing congestion?

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

SIMULATION EVALUATION

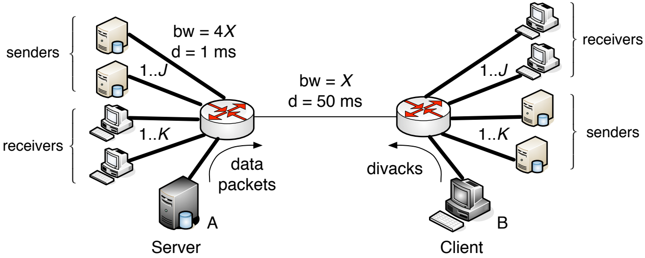


FIGURE: Total transferred files

- Interaction with congestion in different directions and loads.
- Long warm-up period to assure faire share.
- Identical RTTs to isolate the effect of divack flow.
- One divack flow with different flow sizes.

DIVACK'S ALGORITHMS

FIGURE: Policies tested for divack's flows.

Set	Policy	ACK division activated when the sender is in	Number of divacks sent when ACK division is activated	ACK sending mechanism when ACK division is deactivated
I	*divss1* *divca1* *divssca1*	SS CA SS+CA	m for every in-order data packet ($r = m$)	One ACK per in-order data packet ($r = 1$)
II	*divss2* *divca2* *divssca2*	SS CA SS+CA	m every other in-order data packet ($r = m/2$)	One ACK every other in-order data packet ($r = 0.5$)

DIVACK'S RISKS OR BENEFITS?

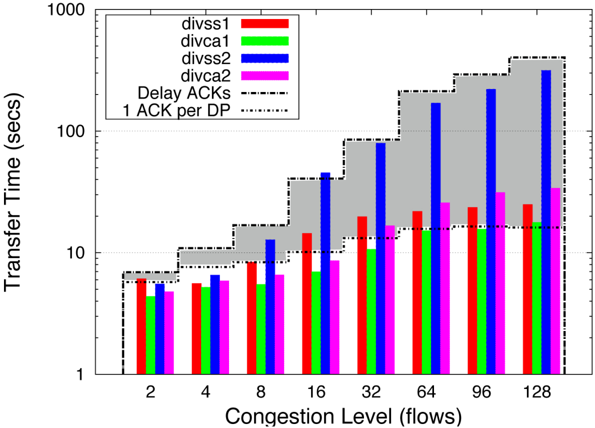
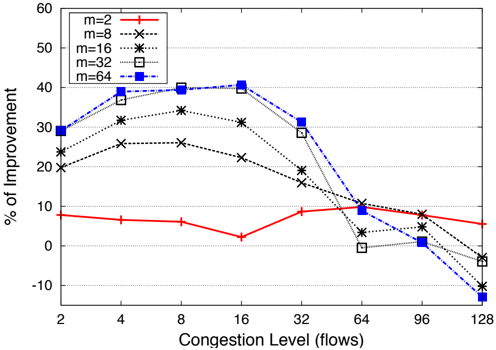


FIGURE: Large files (1.5 MB), without ABC, one-way background traffic.

- Gray zone ⇒ divacks does not improve *standard* performance.

Divack's risks or benefits?



Improvements during congestion avoidance.
policy: *divca1*, reference: 1-apdp, background: one-way traffic.

Figure: Improvements in throughput for some divack's frequencies.

COMPENSATING EXCESSIVE WINDOW INCREASING

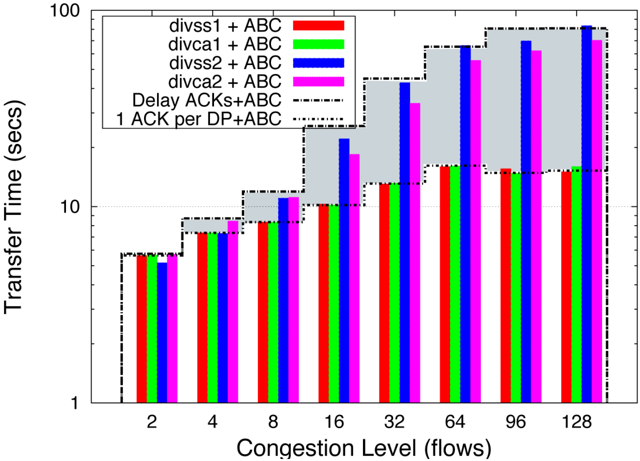


FIGURE: Transfer times for large files (1.5 MB) using ABC, one-way background traffic

OUTLINE

1. INTRODUCTION
   - The Transmission Control Protocol
   - Adapting the Congestion Window
2. ACK CONGESTION CONTROL (ACK-CC)
   - Motivation
   - State of the Art
   - The ACK-CC Mechanism
   - ACK-CC Evaluation
   - Conclusions on ACK-CC
3. ACK DIVISION
   - Definition
   - Divack's Evaluation
   - Conclusions on ACK Division
4. CONTRIBUTIONS
5. CONCLUSIONS
   - Future Work

CONCLUSIONS ON ACK DIVISION

- Divacks accelerates the ACK clocking improving the performance if a transfer concentrates mostly on one of the TCP phases.
- To make divacks attractive, there should be big buffers at the bottleneck (to deal with burstiness) and the receiver should be aware of the current sender's phase.
- Divacks does not represent a threat when congestion increases.
- Divacks helps in wireless access networks but can be a harm in wired networks.
- ABC regulates effectively the excessive number of ACKs.

CONTRIBUTIONS

We have worked out the following contribution in this thesis:

- Exhaustive definition (RFC 5960, waiting in editor’s queue) and testing of a mechanism for ACK Congestion Control (under submission).
- Investigation of the impact of ACK division to accelerate data transmission (CoNEXT 08).
- Testing and analysis of ABC to compensate ACK division (under submission).
- Study and enhancement of TCP performance over 802.16 networks (WCNC 09, LCN 09).

GENERAL CONCLUSIONS

- We've compiled evidence that shows that network impairments can be overcome by adapting the ACK frequency at the extremes.
  - ACK-CC deals with constrained return path both in SS and CA.
  - ACK-CC also allows the increase of performance of competing traffic on the reverse path.
  - ACK division may be used for both purposes, to accelerate transfers and to rapidly ramp-up of the congestion window.
  - There is a threshold for the number of divacks per data packet that limits the harm to competing flows in the divacks's path.
- However, variations in the ACK frequency increase the loss ratio (ACK-CC by increased burstiness, divacks by delayed reaction to a losses).

OUTLINE

ON ACK-CC...

- Embed a pacing mechanism into TCP to deal with burstiness.
- Measure the aggregated congestion and loss rates of ACKs in real networks.
- Implement ACK-CC in a kernel and test it in a test-bed.
- Use SACKs to recover faster the ACK-CC's induced losses.
- Extend the mechanism to MAC layers as WiFi in which *contention* rather than congestion impairs the performance of TCP.

ON DIVACKS...

- Use paced-divacks in the very early stages of slow start to ramp-up cwnd after handovers.
- Evaluate divacks for several users performing long and short transfers.
- Since divacks decrease the burstiness during the SS when using ABC, find a way to demotivate further the use of divacks.

ACKNOWLEDGMENTS

Thank you for your attention!

Questions?