Some Observations on Individual TCP Flows Behavior in Network Traffic Traces

Lei Qian
Brian E. Carpenter

Department of Computer Science

THE UNIVERSITY OF AUCKLAND
NEW ZEALAND
Te Whare Wānanga o Tāmaki Makaurau

October 2011

Disclaimer and acknowledgements

- This talk represents our views only. We have no mandate to speak for any of the bodies mentioned.

- Thanks for valuable comments to Nevil Brownlee, Dongjin Lee, Richard Nelson in particular.

- Thanks to the Department of Computer Science, The University of Auckland for supporting my travel.

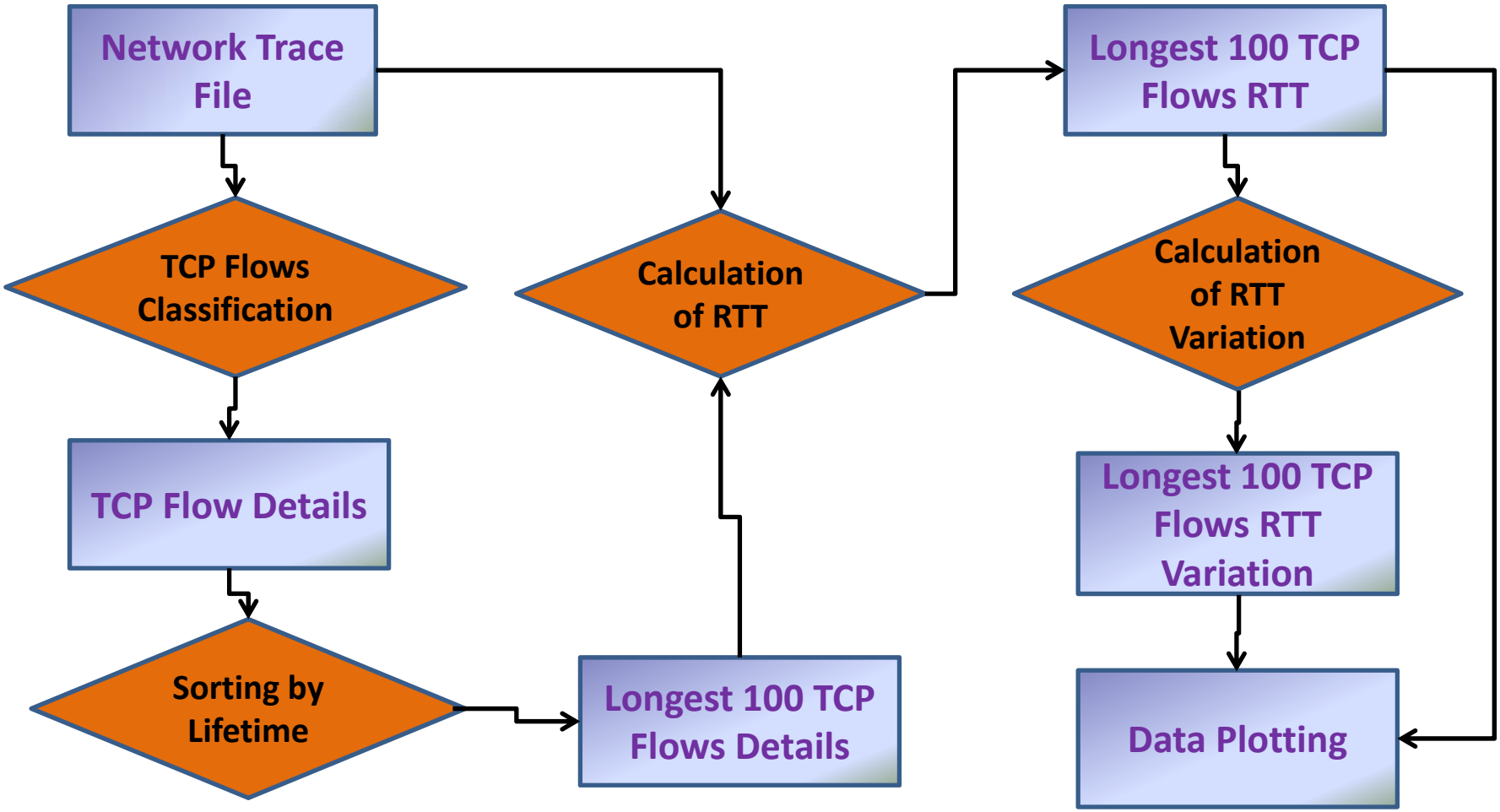
Motivations of the present work

- W. Leland et al stated that “Ethernet local are network (LAN) traffic is statistically self-similarity” in 1994.

- N. Wisipongphan et al stated that ”if one or more streams passing through the bottleneck is self-similar and the aggregate flow does not exceed the capacity, traffic observed at the bottle neck will also be self-similar” in 2003.

- **Our original motivation :** whether individual TCP flows in the trace files available to us have self-similar properties

Measurement Architectures



Contributions of the paper

- A modified algorithm for TCP flows classification and RTT computation. - precise, dynamic and flexible.

- Observed four interesting TCP Flows patterns existence in the network traffic traces.

  (Details will be provided later)

Algorithm used for TCP Flows Classification

- Typical Algorithm: end the flow with long period of Silence. (E.g. 30 Second).
- Modified Algorithm: Applied the Typical Algorithm iff hash table is almost full.
- Therefore, Long, Sparse TCP Flows are able to survive in the hash table.

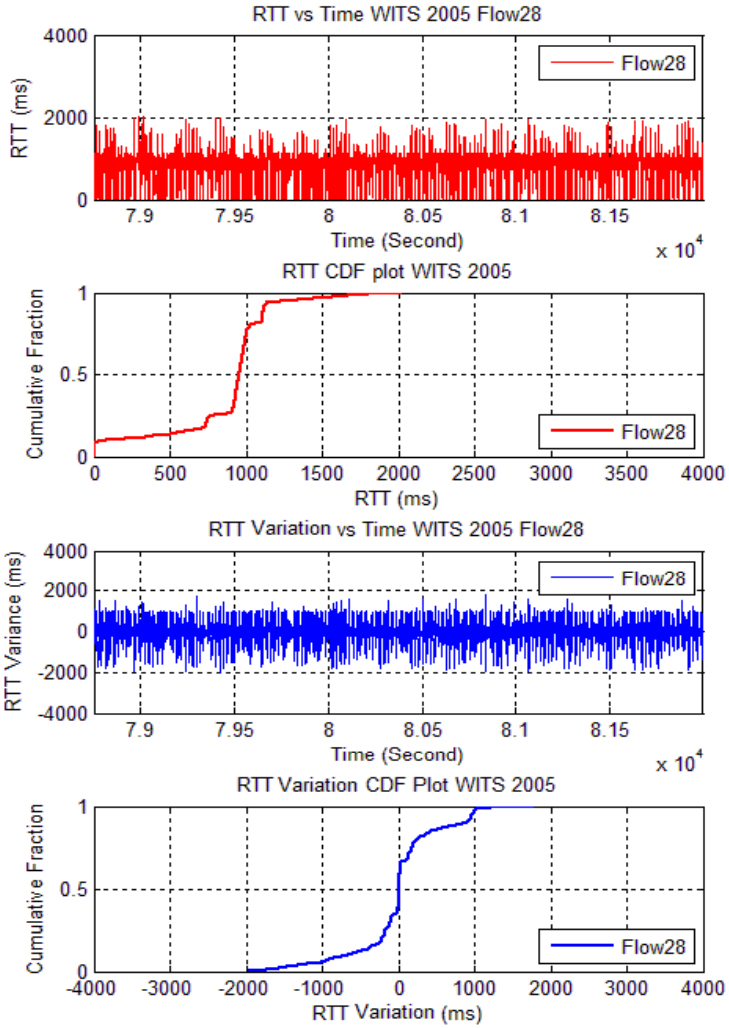
Algorithm used for RTT Computation

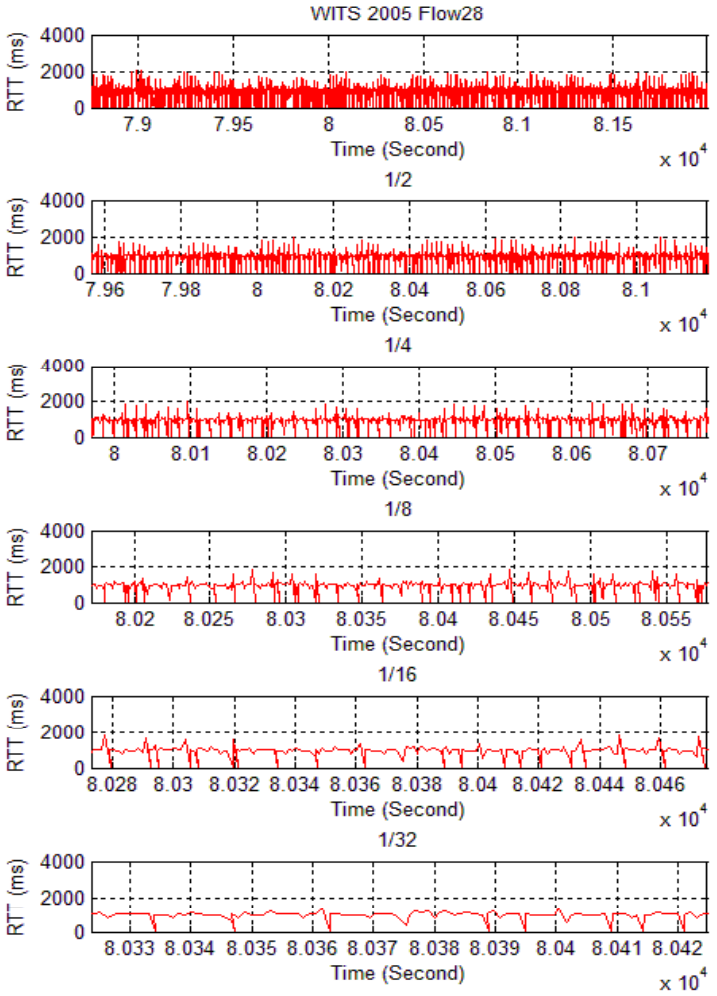
- Typical Algorithm: Treat a packet as lost reply packet after long period. (E.g. 5 Second)
- Modified Algorithm: Applied Typical Algorithm iff hash table is almost full.
- Therefore, Long RTTs are able to be detected.

Main findings on TCP Flows Behaviours

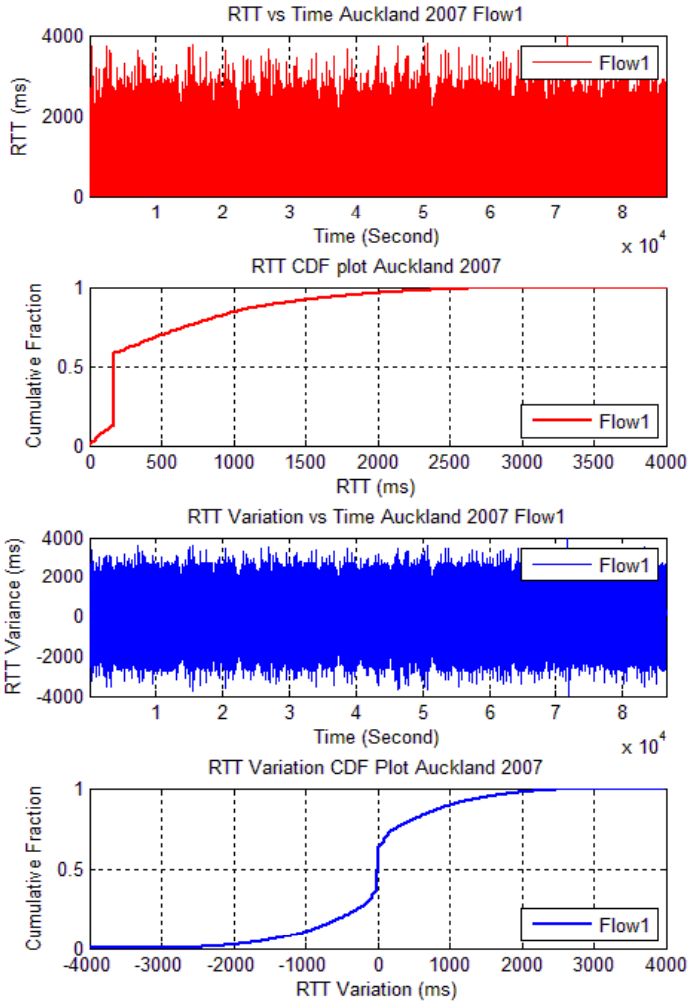
- TCP flows with regular patterns of RTT distribution.
- TCP flows may apparently have self-similar RTT distributions.
- Most long-active but sparse TCP flows use port 80 with a relatively short RTT.
- Flows with extremely high RTT values, but stable end-to-end behavior, that do not necessarily reflect unstable network condition.

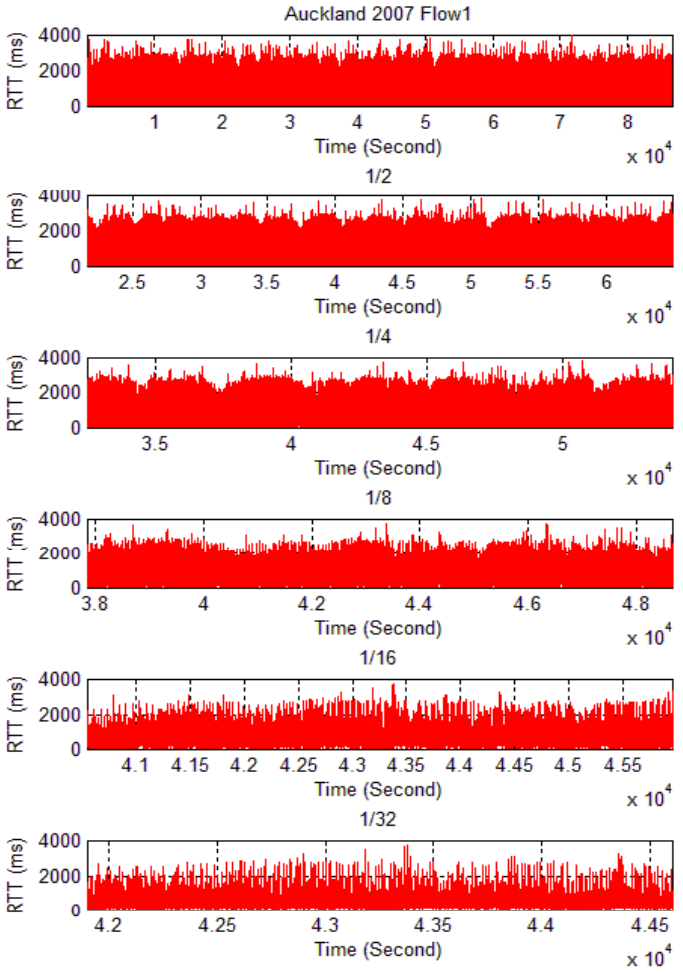
TCP Flows With Regular Patterns of RTT/RTTV Distribution



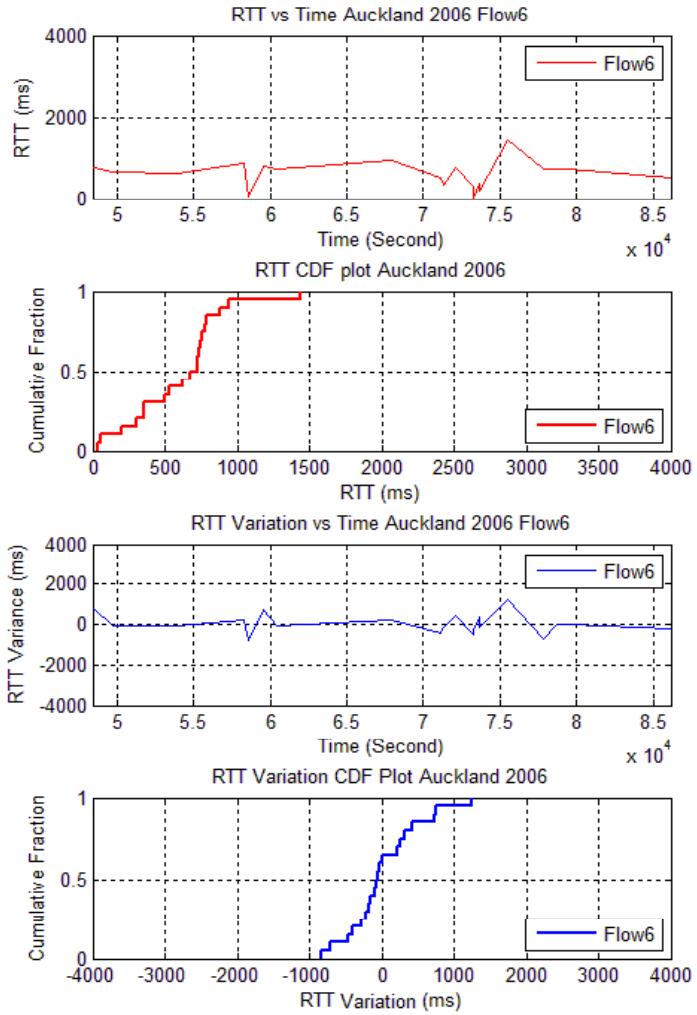


TCP Flows With Self-similar RTT/RTTV Distributions

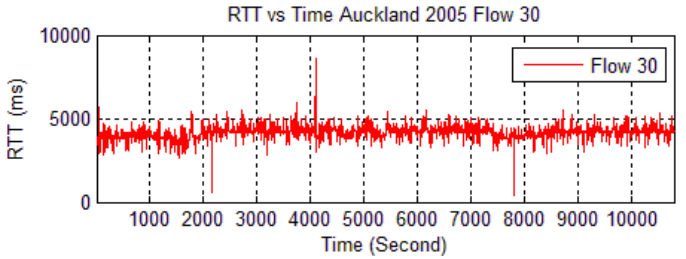


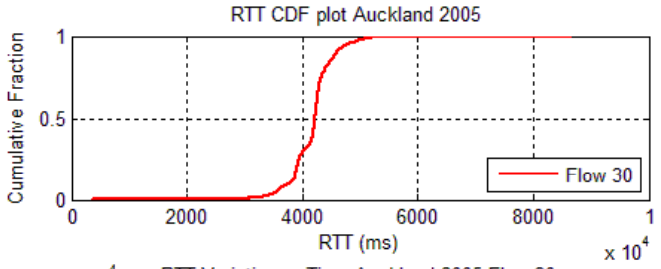


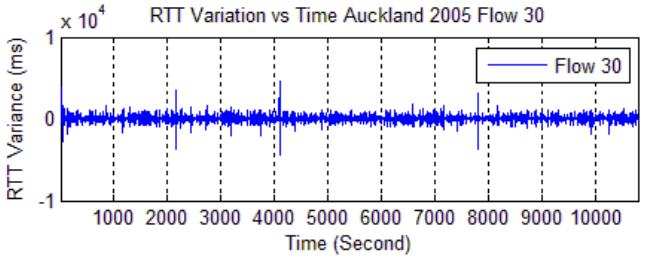
Long-active but Sparse TCP Flows RTT/RTTV Distribution

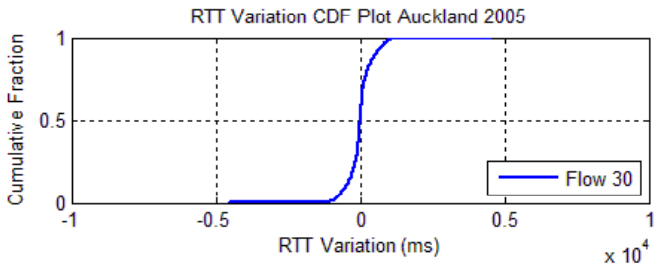


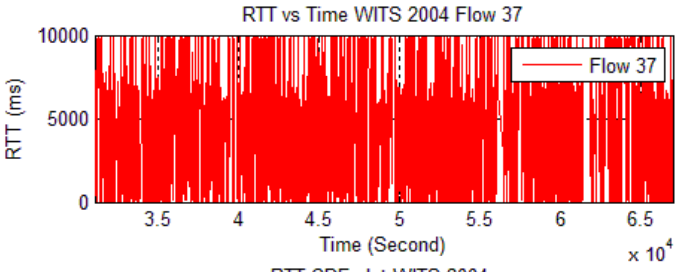
TCP Flows with Long RTT Value

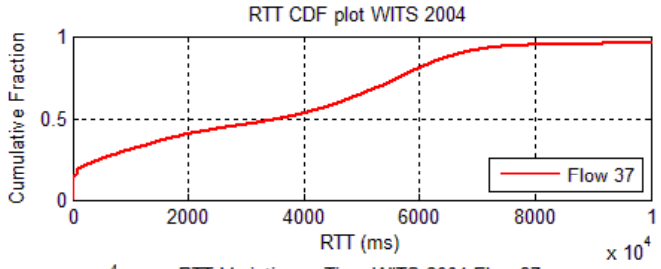


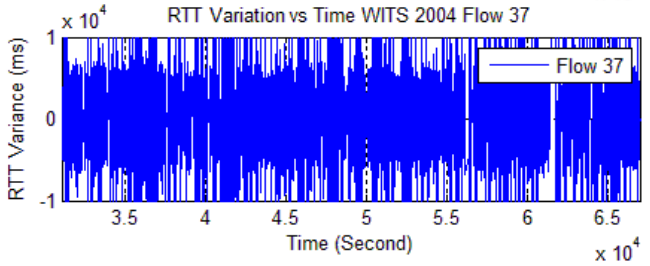


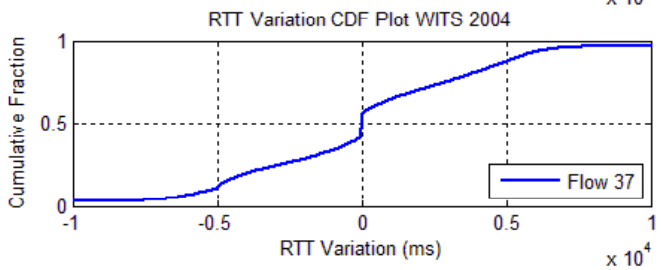












What might be Interesting for Future Work?

- A More Stable technique for long duration, low volume flows derived.
- Build of a Statistical Model to detect Self-Similarity would be interesting.
- Different selection criterion of TCP flows might have different result.

Questions ?

Thank you ☺!