draft-alvestrand-rmcat-congestion-02

Stefan Holmer (holmer@google.com), Google
Luca De Cicco (l.decicco@poliba.it), Politecnico di Bari
Saverio Mascolo (mascolo@poliba.it), Politecnico di Bari

Known issues

- Poor self-fairness.
  - Due to having an adaptive jitter model.
- Starved by loss-based flows.
  - Delay-based CC sees congestion all the time.
  - No transition between delay-based and loss-based.

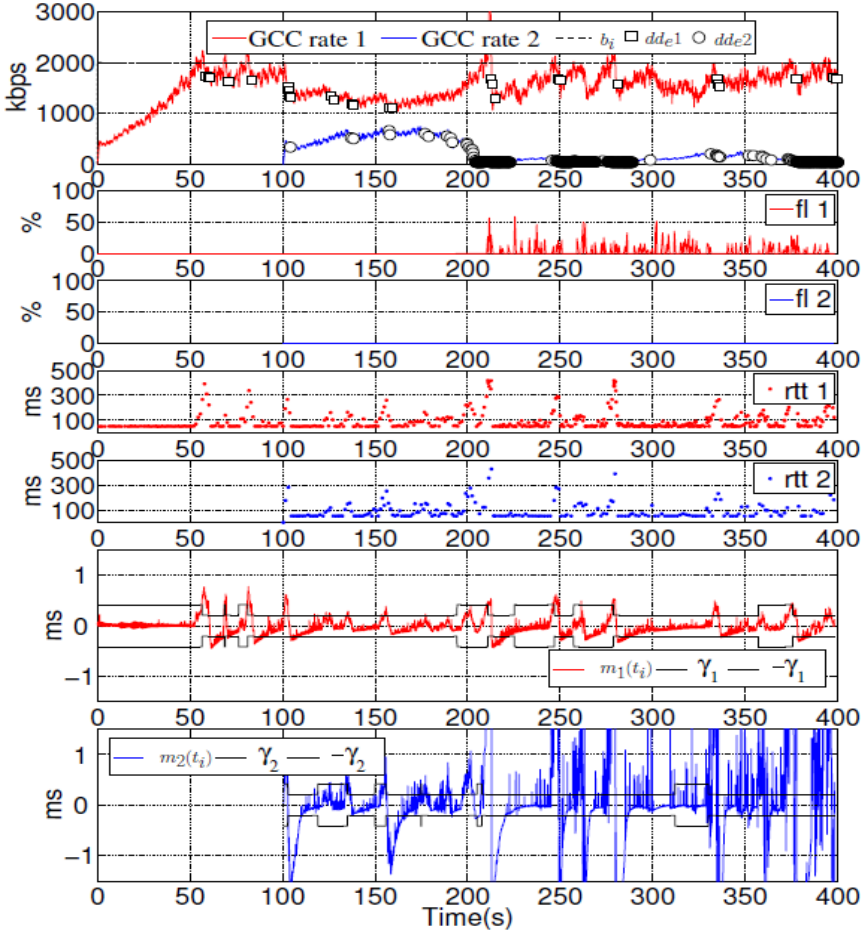
Solution?

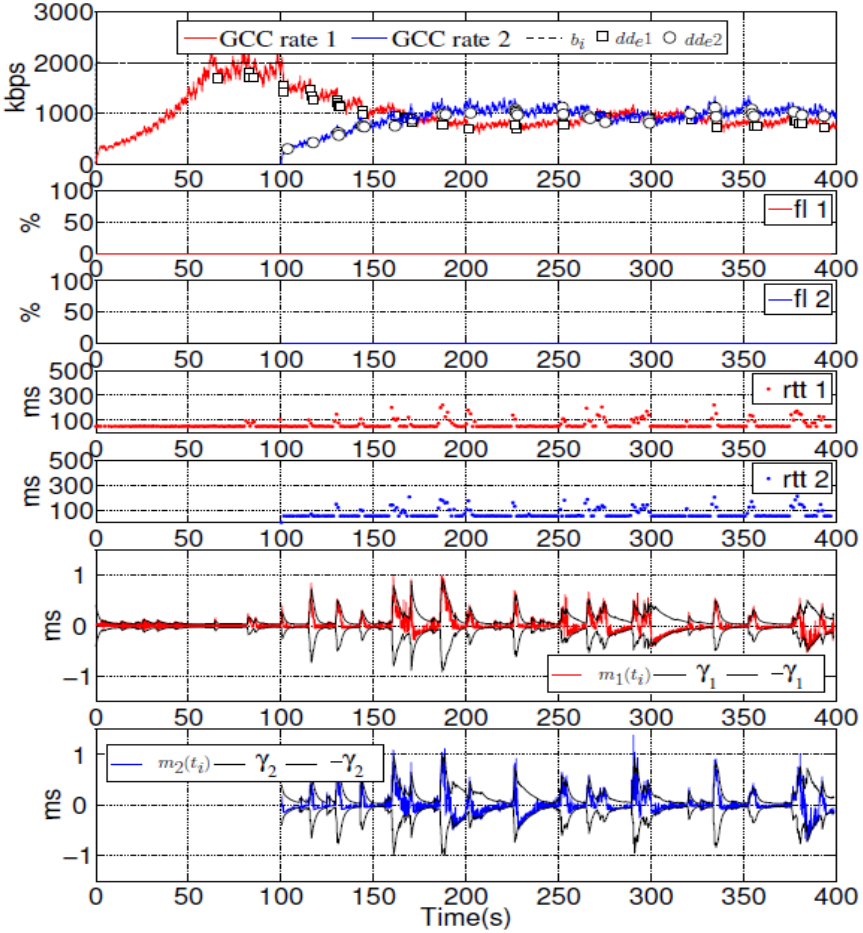
- More jitter, more filtering, less sensitive.
  - Takes more bandwidth, due to static threshold.

- Make the delay threshold (gamma) adaptive.
  - Increase the threshold when detecting often.
  - Reduce when not detecting.

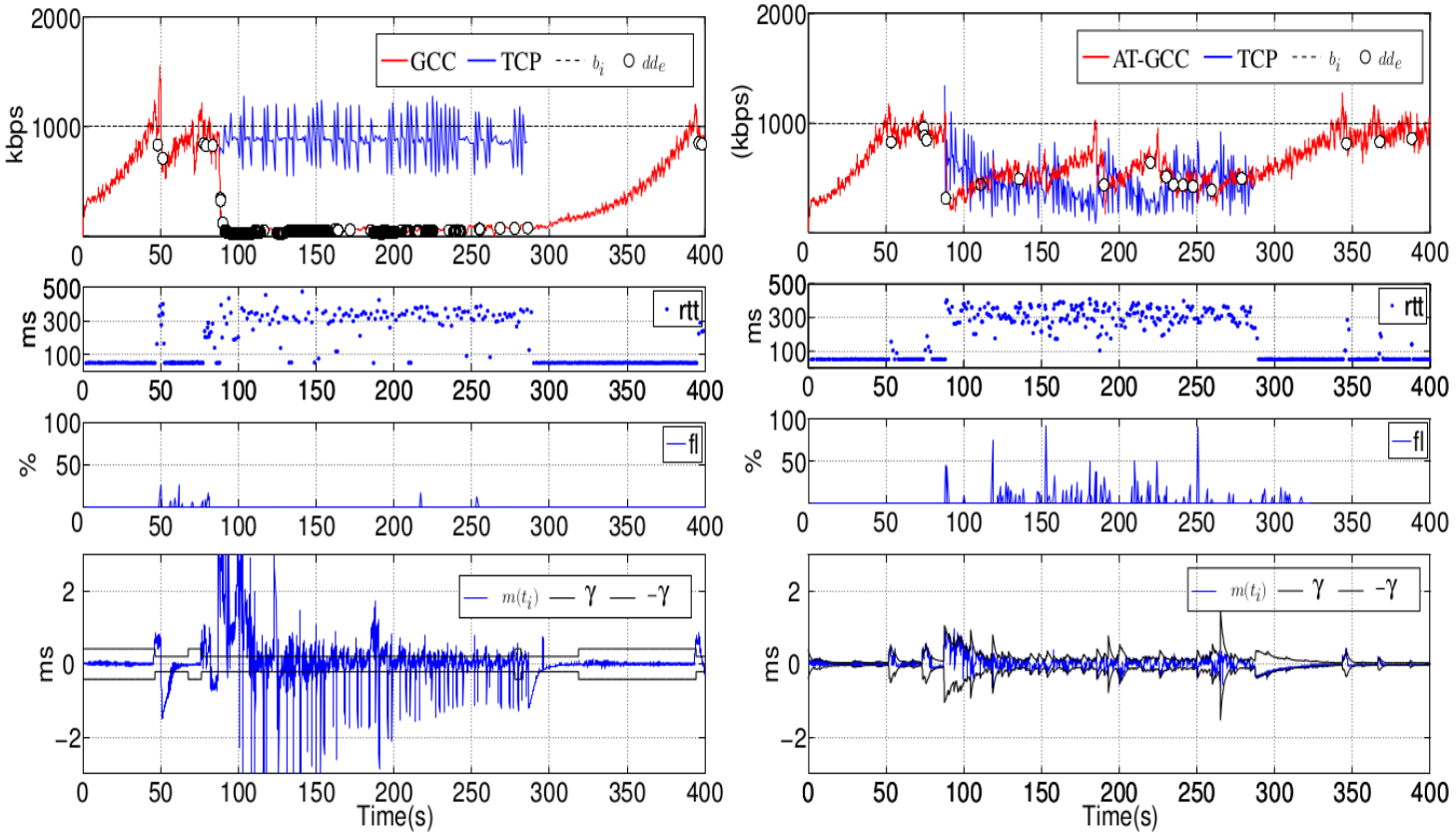
$$\gamma(t_i) = \gamma(t_{i-1}) + \Delta T \cdot \begin{cases} k_d(|m(t_i)| - \gamma(t_{i-1})) & |m(t_i)| < \gamma(t_{i-1}) \\ k_u(|m(t_i)| - \gamma(t_{i-1})) & \text{otherwise} \end{cases}$$

Two GCC Flows

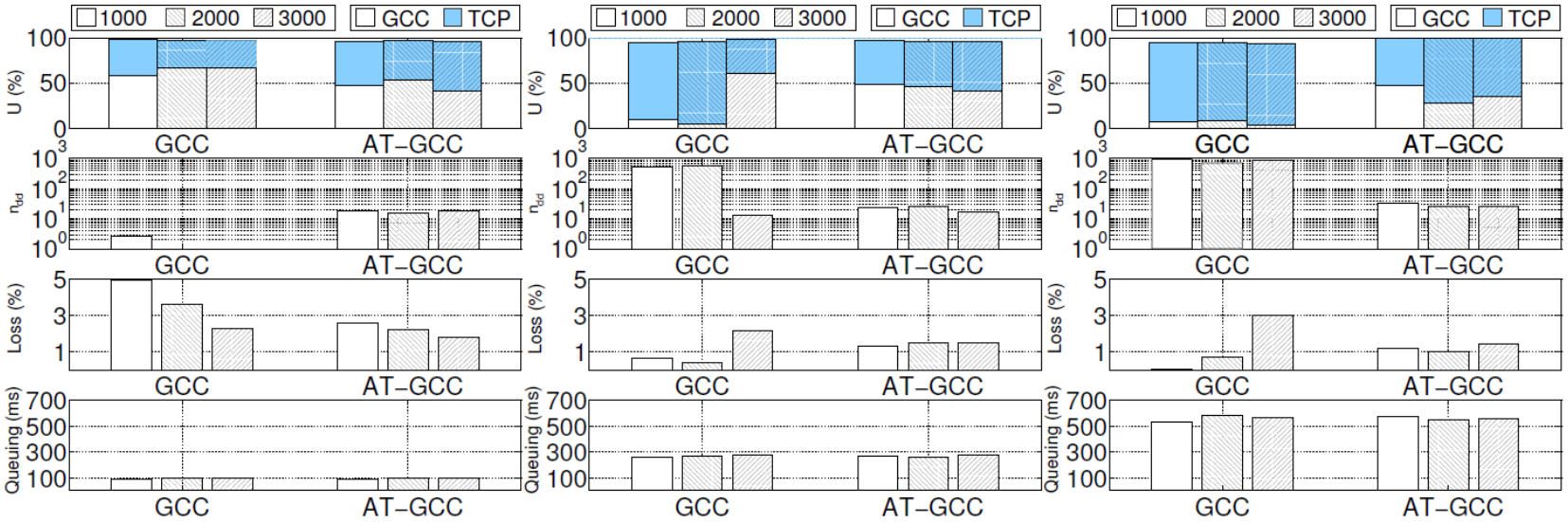




TCP vs GCC



TCP vs GCC



(a) $q_i = 150$ms (b) $q_i = 350$ms (c) $q_i = 700$ms

Conclusions

- An adaptive-threshold mechanism has been proposed.

- Evaluation in Chromium showed:
  - TCP starvation is avoided.
  - Self-fairness is improved.
  - In the case of a single flow queuing delay and losses are reduced.