Understanding Slow BGP Routing Table Transfers

Zied Ben-Houidi[*†]
Mickael Meulle[*], Renata Teixeira[†‡]

**France Telecom R&D Orange Labs*
†UPMC- Sorbonne Universités, LIP6
‡Centre National de la Recherche Scientifique (CNRS)

Slow table transfers can delay BGP convergence

- BGP convergence involves three main steps
  - Event detection
  - Route propagation
  - Routing table update

- Propagation time can be significant
  - When too many routes change (widely suspected to be true but no hard evidence to back it up)

- Table transfer = change of a large number of routes
  - BGP sessions failures/resets
  - Intra-domain events that cause BGP changes

Our contributions

- Analysis of table transfers in a VPN provider
  - YES! They are slower than they should be.
  - They are slow because of gaps
- Controlled experiments with different router models
  - Gaps happen in all routers
  - Gaps are due to timer-driven implementation
- Can we make them faster?
  - Yes
  - We explore the limits
    - Sender induced
    - Receiver induced

What are we measuring?

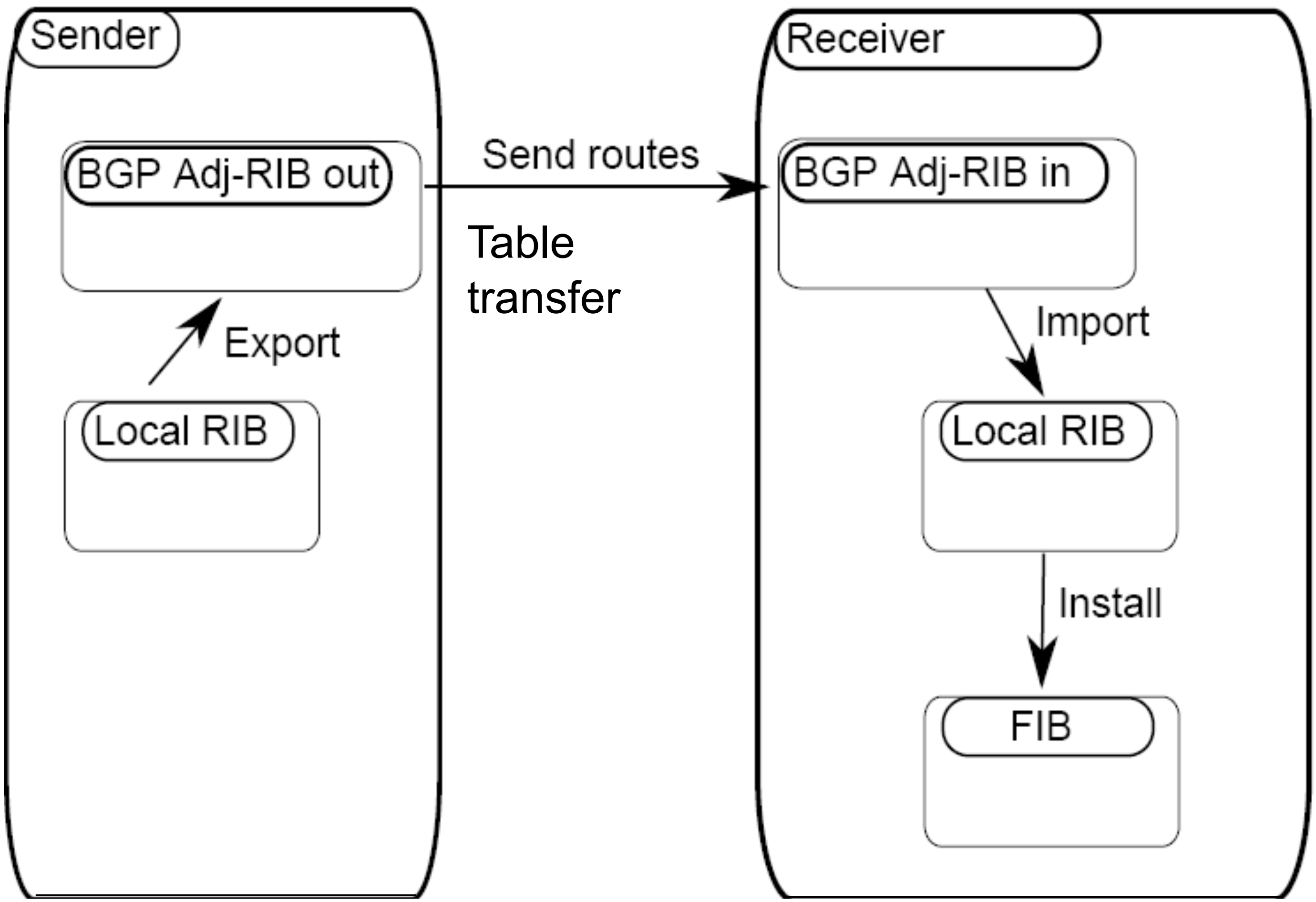
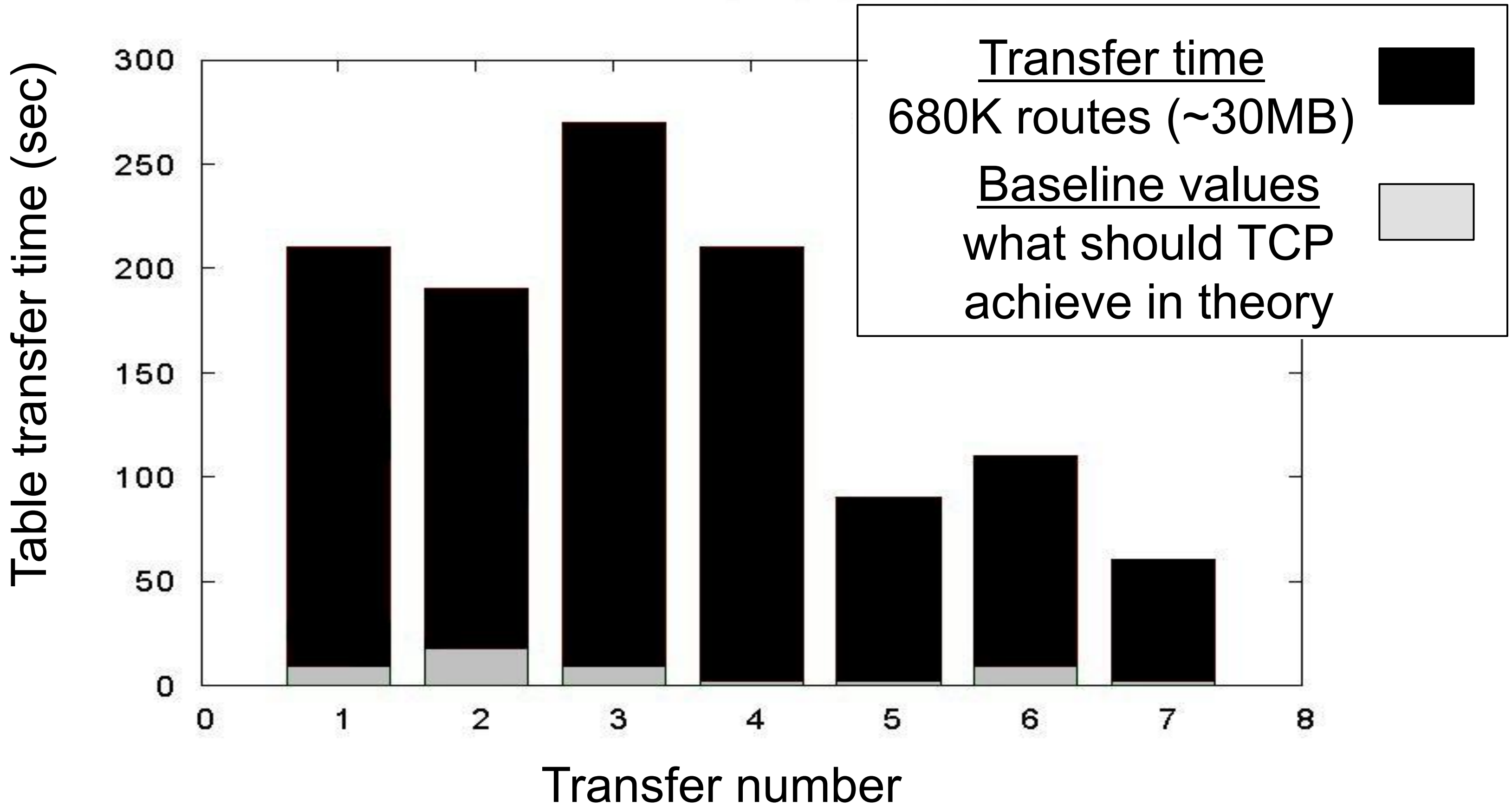


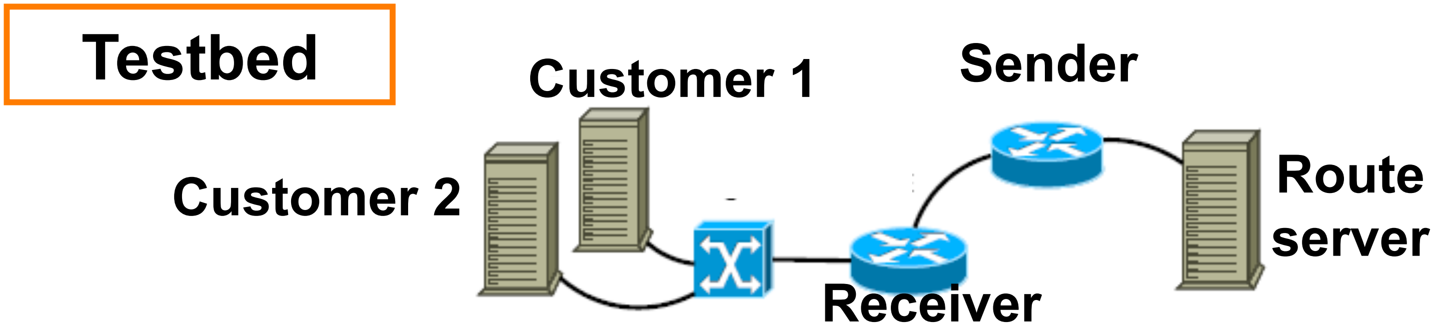
Table transfers are slow



Slow table transfers because of gaps

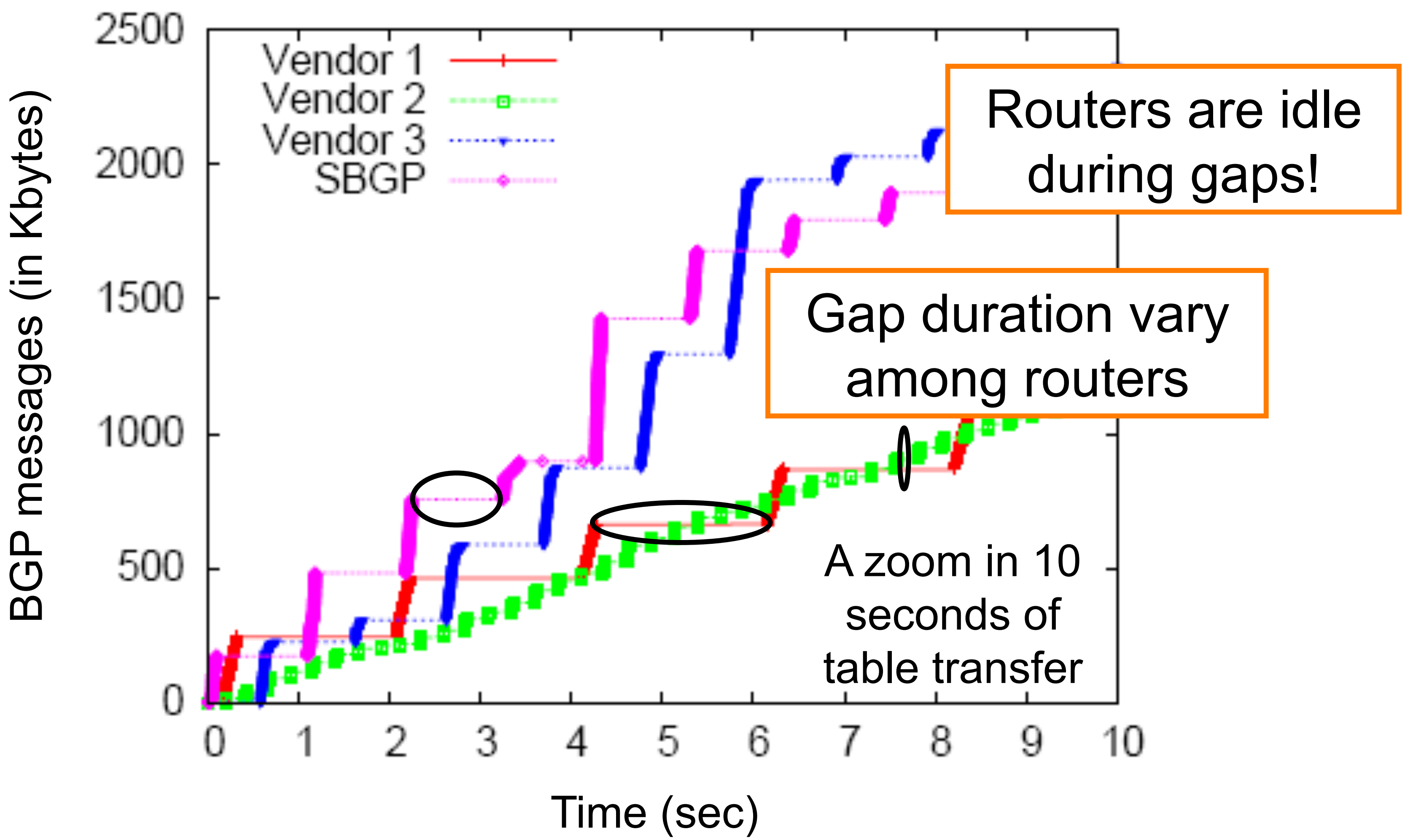
- Tap connection between two routers
- Gaps = no routes are exchanged
  - Tend to be regular (duration/frequency)
  - Account for most of the transfer time
- Causes of gaps not documented
  - Problem: no control of operational routers

Controlled experiments



- Induce table transfers
  - Reset session between Sender and Receiver
  - Test 6 Senders and 2 Receivers
    - Carrier-class routers from 3 different vendors
- Tap connection between Sender and Receiver
- Monitor routers' CPU load

Gaps arise in all tested routers



Gaps are caused by a timer-driven implementation

- Analysis of a software BGP speaker
  - Timer rate-limits sending of routes
    - Expires each second

- Discussions with two router vendors
  - Both confirm our analysis
  - Vendor 1: Unintentional
  - Vendor 2: Rate-limiting to control router load

Reducing table transfer time

- Increase the sending rate
  - Remove the timer: send routes after TCP ack
    - Transfer time ~ Baseline values
    - Event driven SBGP (fast SBGP)
    - Vendor 1's new software
  - Send more routes between two gaps
- Reduce the table size

The timer-driven implementation of Vendor 1

- One process formats BGP routes and put them in a queue
- Another process drains this queue to send routes (using TCP)
- If, for some reason, the queue is full, the process goes into two seconds sleeping mode which causes the two seconds gaps.

Towards an event-driven implementation

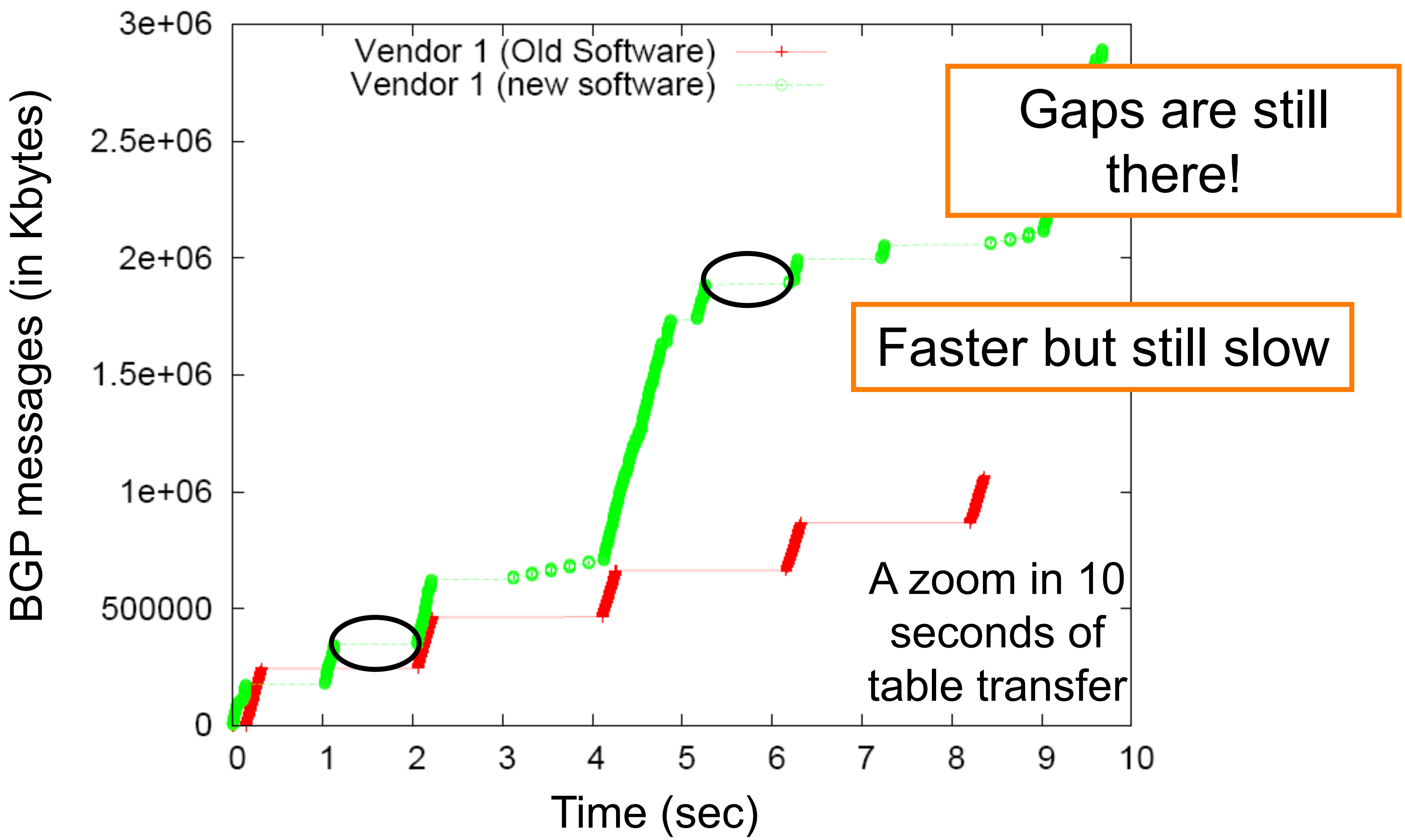
- We modified SBGP
  – Remove the timer, send after TCP ACK

  => Event driven

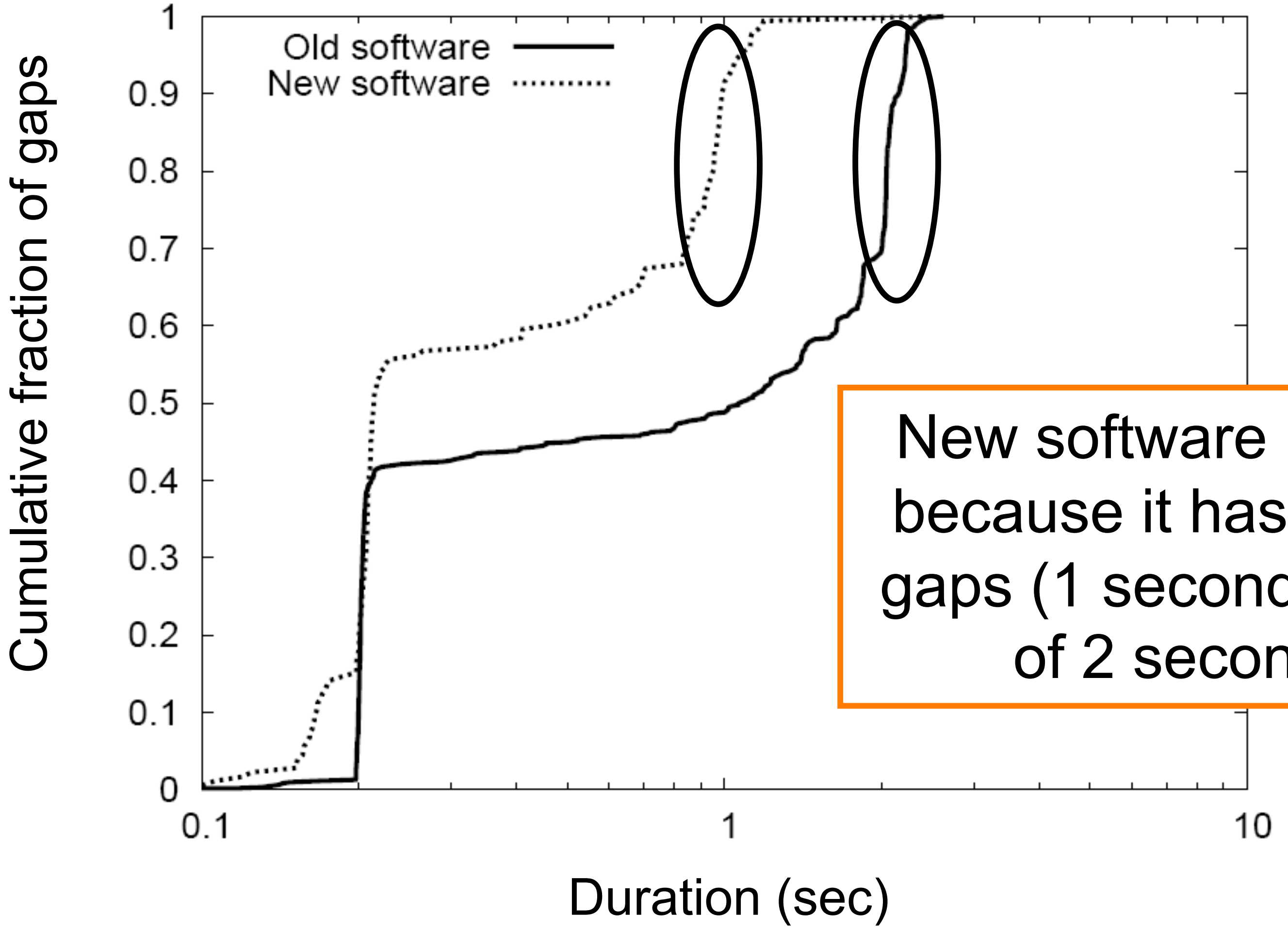
- Vendor 1's fix:
  – Sleeping process "wakes up" after TCP ACK

  => Event (TCP ACK) driven.

Evaluation of vendor 1's new software

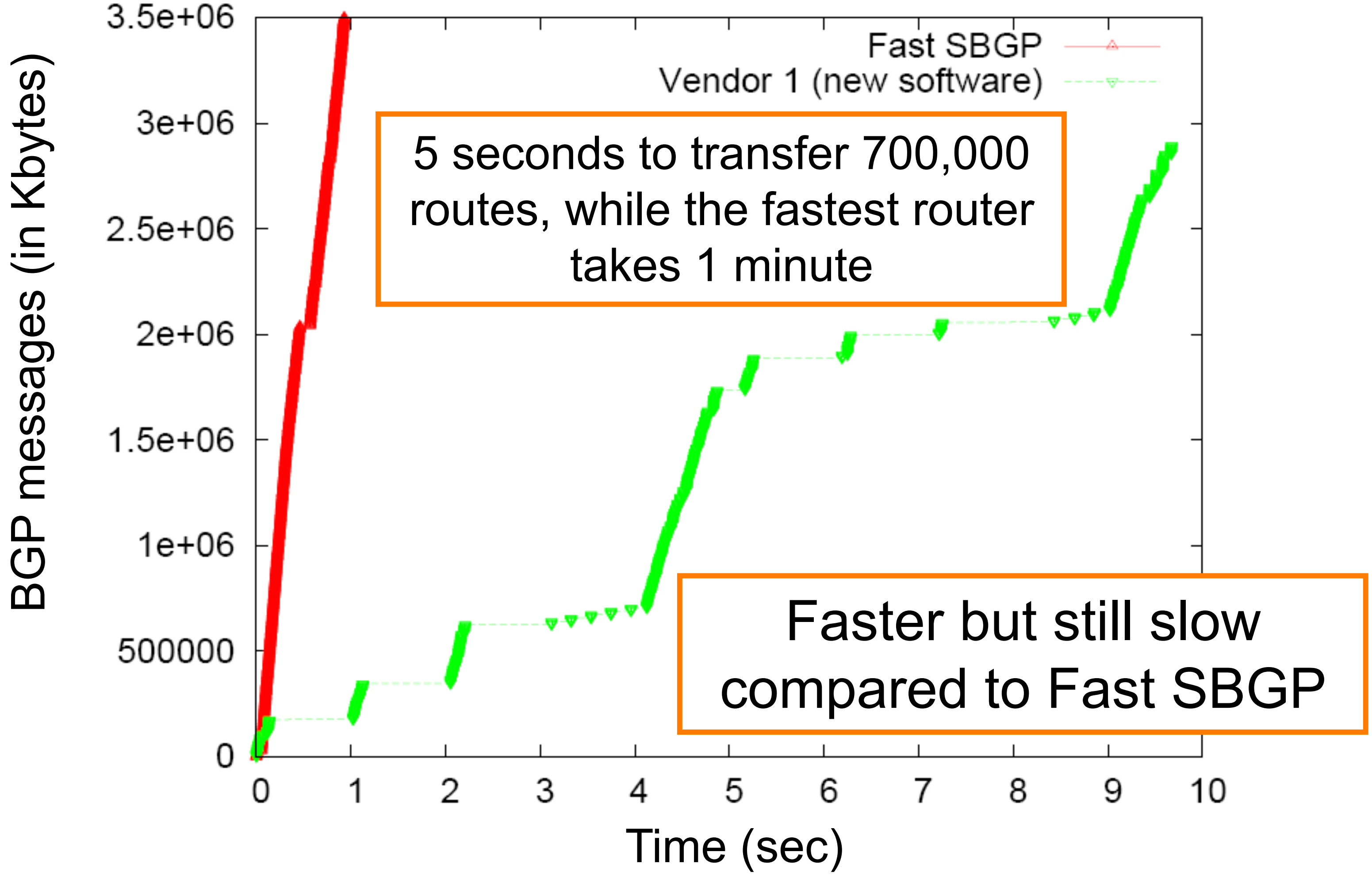


Duration of gaps in vendor 1's software

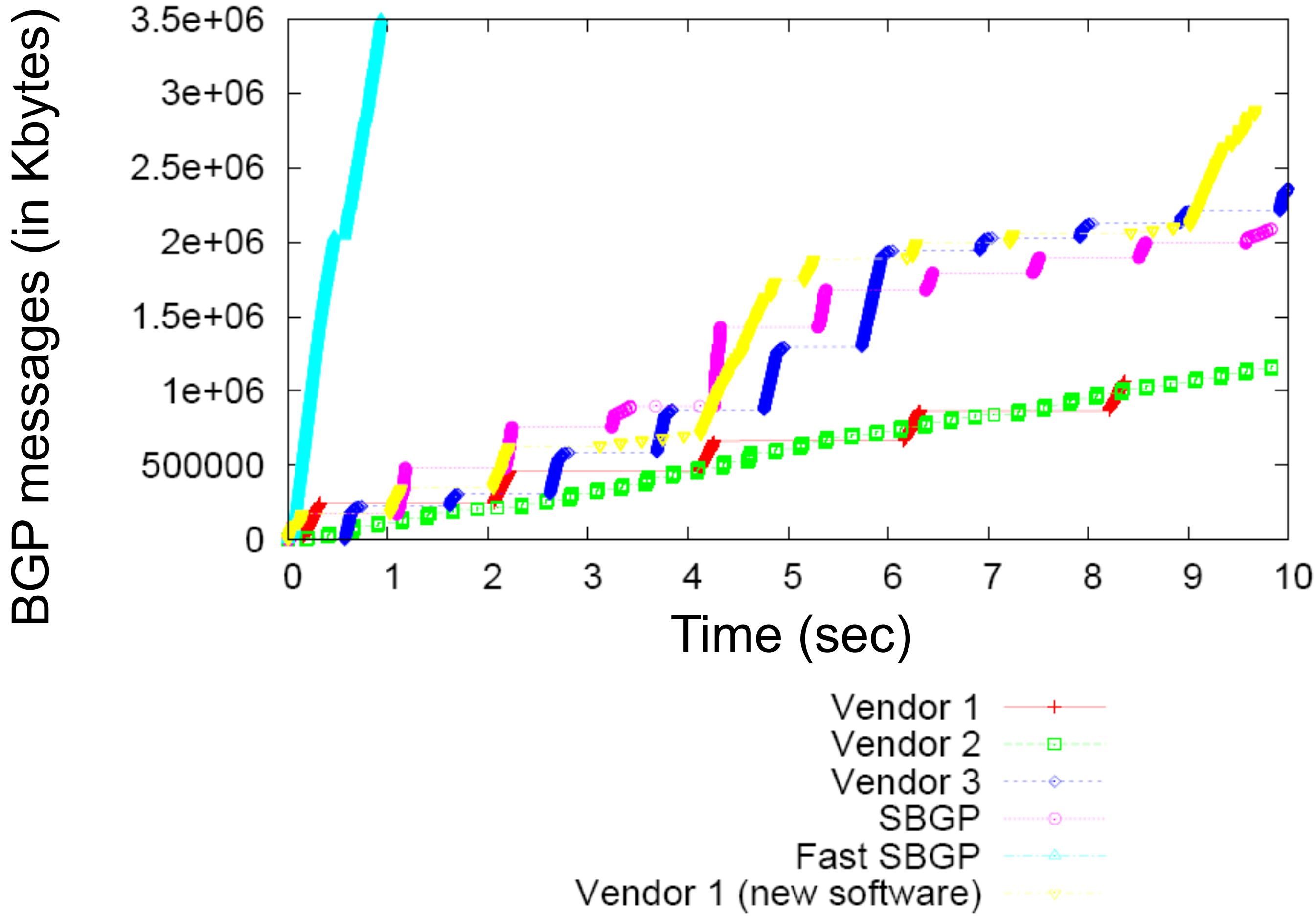


New software is faster because it has smaller gaps (1 second instead of 2 seconds)

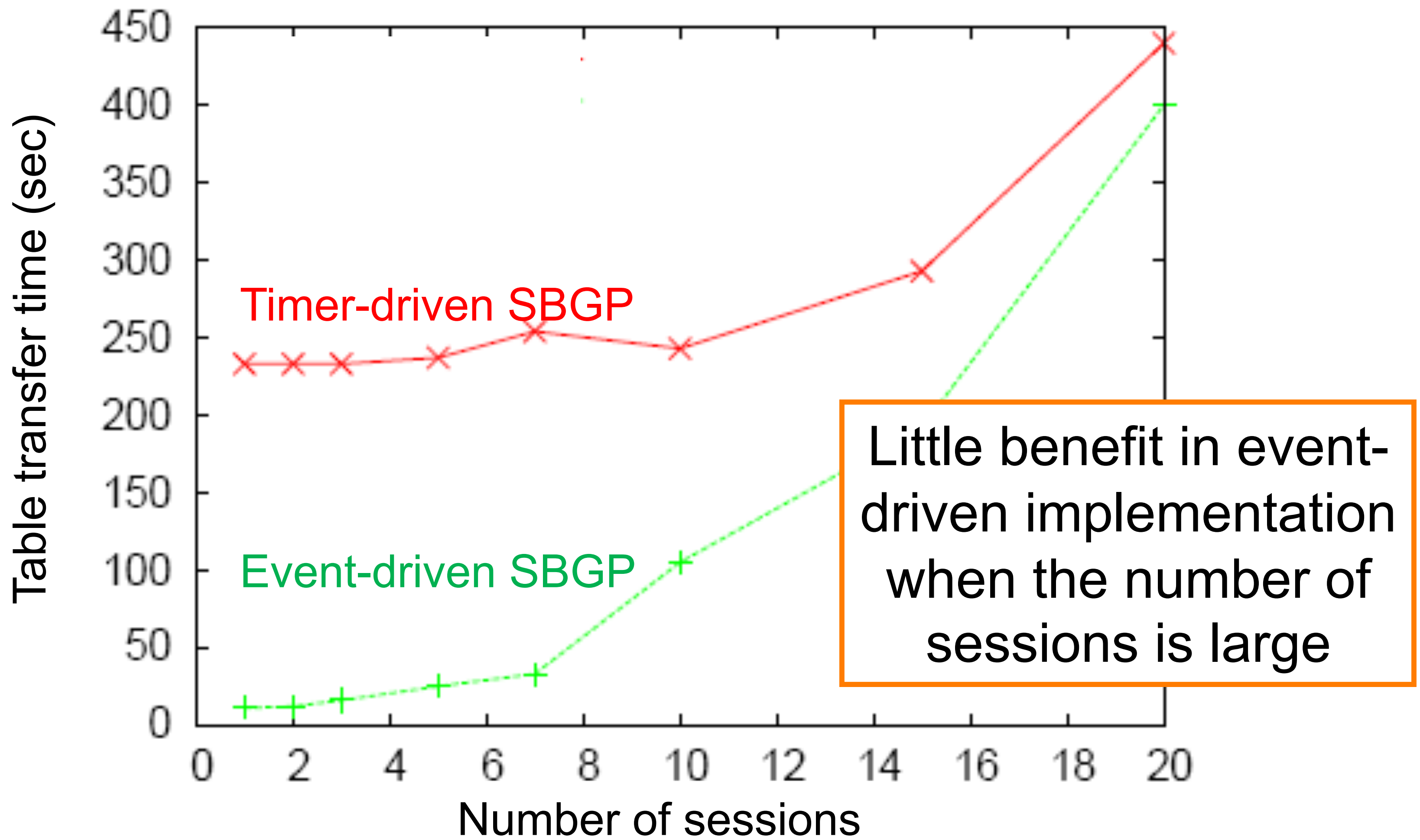
Vendor 1's new software Vs Fast SBGP



A comparison of all tested implementations



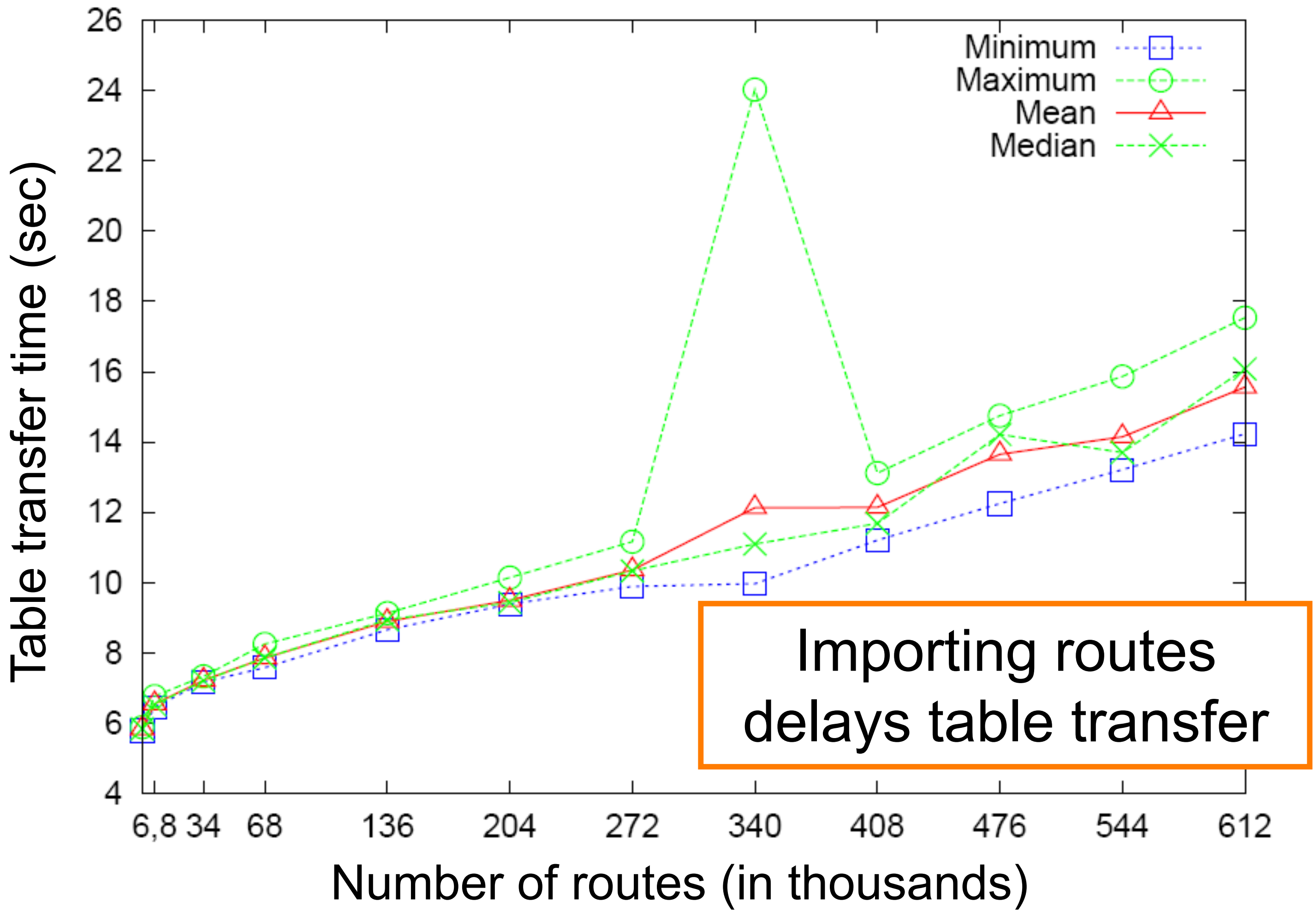
The number of sessions is a limit



The number of imported routes is a limit

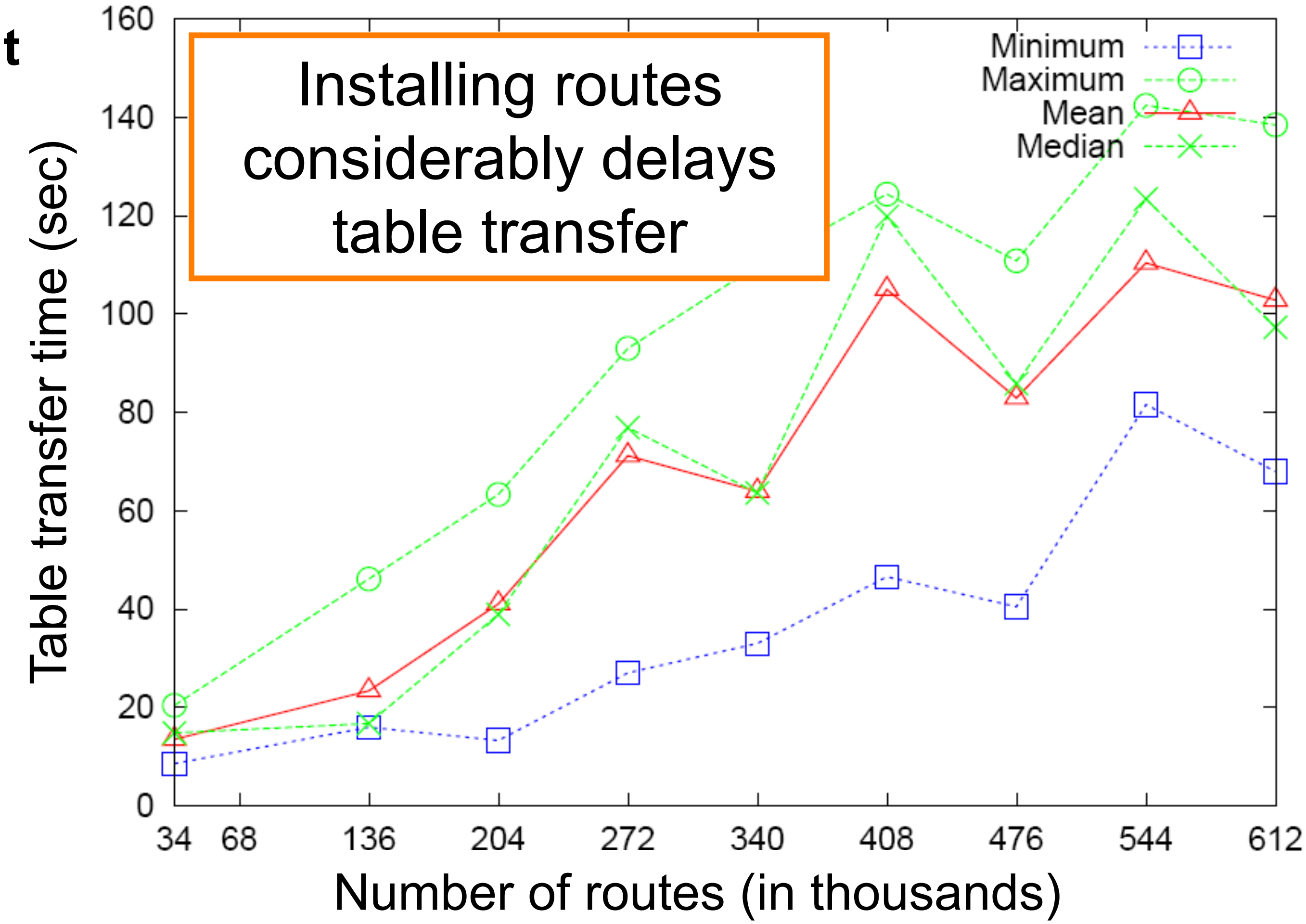
- Fix # of routes **sent**.

- Vary # of routes **imported**



The number of installed routes is a limit

- Fix # of routes **sent**
  - Vary # of routes **installed**



Conclusion

- Table transfers are slow
  – Gaps caused by timer-driven implementation

- Faster table transfers
  – Possible: Event-driven implementation
  – There are limits
    – Routers with multiple BGP sessions
    – Importing/installing routes

- Router load vs. fast table transfers
  – Best configuration depends on the network
  – Today's hard-coded solution is too limiting

More information

- Zied Ben Houidi, Mickael Meulle and Renata Teixeira, "Understanding Slow BGP Routing Table Transfers", in Proc. of ACM Internet Measurement Conference (IMC), November 2009.