

Routing Stability in Congested Networks: Experimentation and Analysis

SIGCOMM 2000 Presentation by Aman Shaikh

Authors

Aman Shaikh (UCSC, AT&T Research)

Lampros Kalampoukas (Xebeo Communications)

Rohit Dube (Xebeo Communications)

Anujan Varma (UCSC, TeraOptic Networks)

Aug 31, 2000

Outline

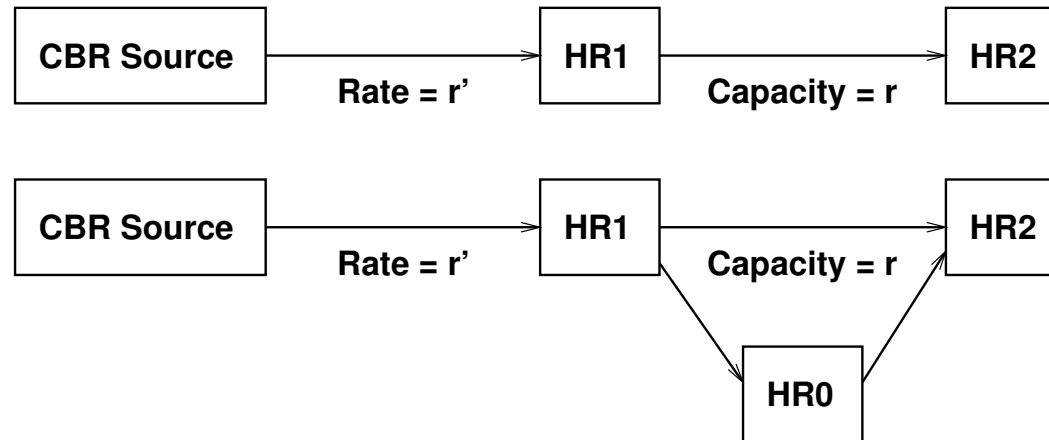
- **Introduction**
- **Methodology and network configuration**
- **Analytical models for OSPF and BGP**
- **Experimental results**
- **Conclusions**
- **Future work**

Introduction

- **Routing protocols exchange various “control messages” for disseminating routing information and determining liveness of peering sessions.**
- **Investigate effects of control message losses on stability of routing protocols**
 - **Focus on packet losses due to network congestion.**
- **Previous studies have reported on the correlation between BGP instability and network usage in the Internet.**
- **Goal of this work: study behavior and evaluate robustness of OSPF and BGP when routing messages are dropped possibly due to congestion.**
- **Use both experimentation and analytical modeling to gain insight to protocol dynamics.**

Network configuration

- Study protocol behavior in *2-node* and *3-node* configurations:



- Keep the link HR1- \rightarrow HR2 consistently overloaded by having a CBR traffic source send packets at a desired rate.
- Packets are dropped with a certain probability $p = \frac{r'-r}{r'}$ at HR1.
- Link overload factor f is defined as $f = \frac{r'-r}{r}$.

Methodology

- Depending on protocol, successive routing packet losses result in peering session failure.
 - Determine the effect of traffic overload on *Mean-Time-to-Flap* (or *U2D*) and *Mean-Time-to-Recover* (or *D2U*) for OSPF and BGP.
- In the *2-node* experiments, a static route pointing directly to HR2 is installed in HR1.
 - This allows studying the effect of traffic overload on *flap-recovery* (*D2U*) time.
- In general, the *3-node* topology represents a more realistic configuration.

Methodology (cont'd)

- **Evaluated robustness of OSPF and BGP for a variety of parameters.**
- **Overload factors used: 25%, 50%, 100%, 200%, 400%.**
- **Data packet sizes: 64 bytes, 256 bytes and 1500 bytes.**
- **Two different buffer sizes at HR1: 4 MB and 16 MB.**
- **2–node vs. 3–node setup.**
- **Collected between 10 and 16 samples for U2D as well as D2U for each configuration.**
- **Compute average values, and 95% and 99.5% confidence intervals for U2D (flap) and D2U (recovery) times.**
- **Compared experimental results with analytical results.**

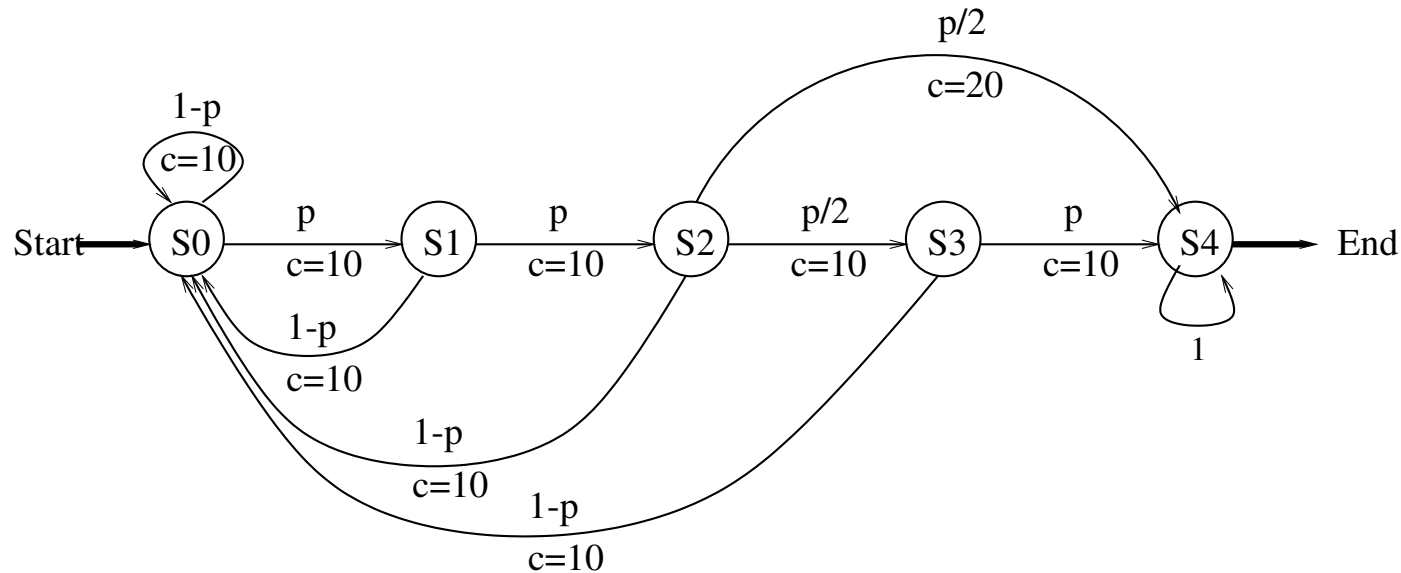
Analytical models

- Our analytical modeling of OSPF and BGP relies on the following two assumptions:
 1. The overload factor remains constant.
 2. Every packet has the same probability p of being dropped which depends on the overload factor. Decision of dropping a packet is made independently for every packet.
- Use of Markov chains to find the expected values of U2D and D2U for OSPF and BGP.

OSPF model

- HR1 sends a “hello” packet every “HelloInterval” to HR2.
- HR2 declares HR1 down if it does not receive a hello packet from HR1 for “RouterDeadInterval”.
- Everytime HR2 receives a hello packet from HR1, it resets “RouterDeadTimer” and schedules it to expire “RouterDeadInterval” later.
- On our routers, HelloInterval = 10 seconds
RouterDeadInterval = 40 seconds.
- $E[U2D]$ for OSPF = Expected time for 4 consecutive hello packets to be dropped at HR1.
- However, in reality, “HelloTimer” is jittered, following a uniform distribution over $(10 - \delta, 10 + \delta)$ interval.
- On our routers, $\delta = 1$ second.

OSPF model: U2D (flap) time



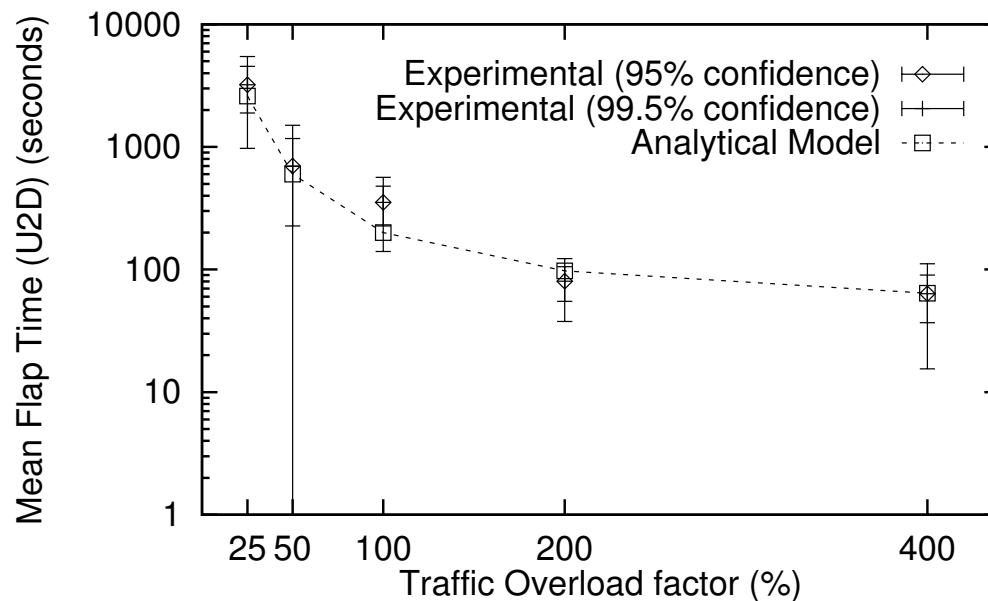
- $E[U2D]$ = Expected time for system to move from start state (S_0) to end state (S_4).
- State transition from $S_2 \rightarrow S_4$ models the effect of the “HelloTimer” jitter:
 - 50% of the time OSPF will flap only after 3 “hellos” fail to reach HR2.

OSPF model: U2D (flap) time (cont'd)

- $E[U2D]$ for OSPF =

$$\left(\frac{20}{p^4 + p^3} + \frac{20}{p^3 + p^2} + \frac{10p + 20}{p^2 + p} + \frac{10}{1 + p} \right)$$

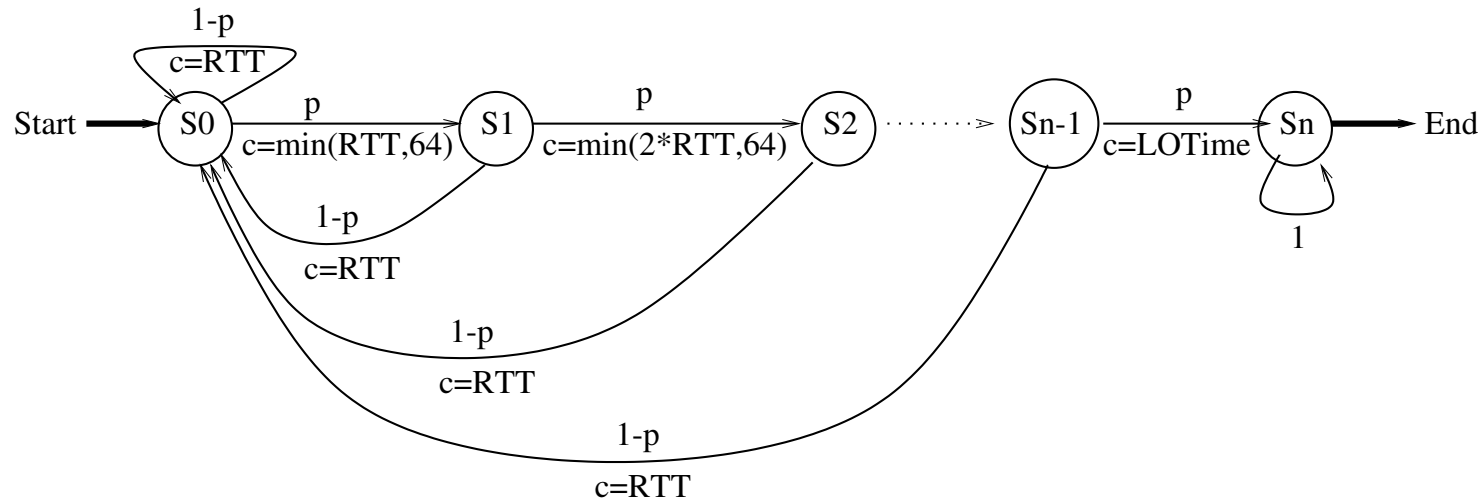
- **Experimental Results: packet size = 64 bytes, buffer size = 4 MB.**



BGP model: U2D (flap) time

- **BGP uses TCP as an underlying reliable transport layer.**
- **When modeling the U2D time, we need only to model the successful transmission of a *single* BGP “keepalive” message.**
 - **TCP sessions is already established.**
 - **TCP enforces in-order packet delivery: keepalives generated by BGP will be transmitted in the order they were submitted to the TCP session.**
 - **Similarly, receiving TCP sessions will pass keepalive messages to corresponding BGP session in a sequence preserving order.**
- **The behavior depends on TCP retransmissions and RTT estimations.**
 - **Hard to get a general closed-form analytical expression that is configuration independent.**

BGP model: U2D (flap) time (cont'd)

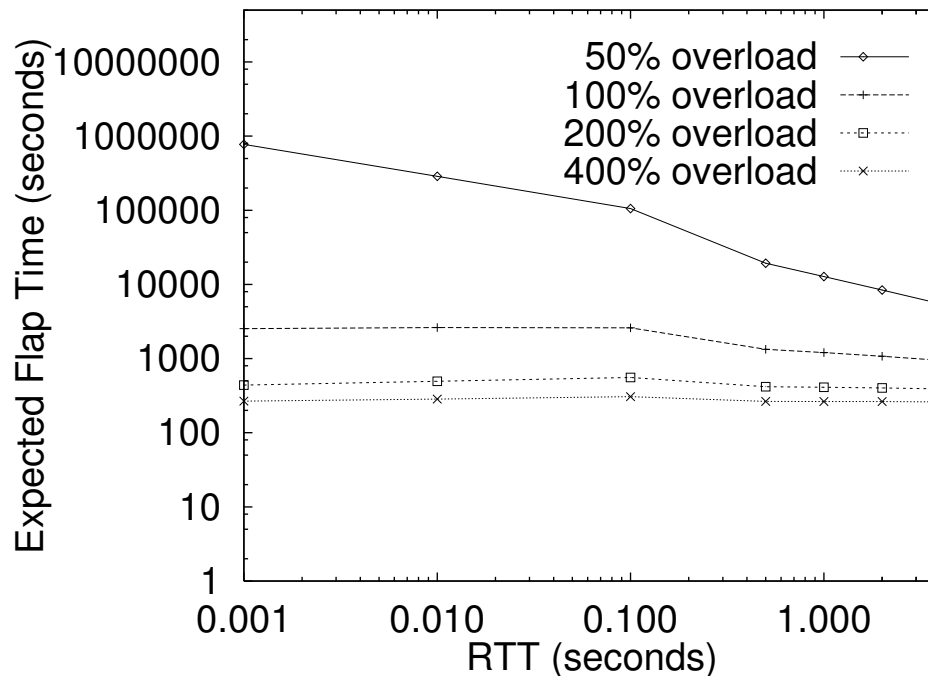


- $E[U2D]$ for BGP with RTT of 1 second and HoldTime of 180 seconds =

$$\frac{1}{p^8} (1 + p + 2p^2 + 4p^3 + 8p^4 + 16p^5 + 32p^6 + 64p^7 + 52p^8)$$

BGP model: U2D (flap) time (cont'd)

- Effect of RTT on $E[U2D]$ of BGP ...



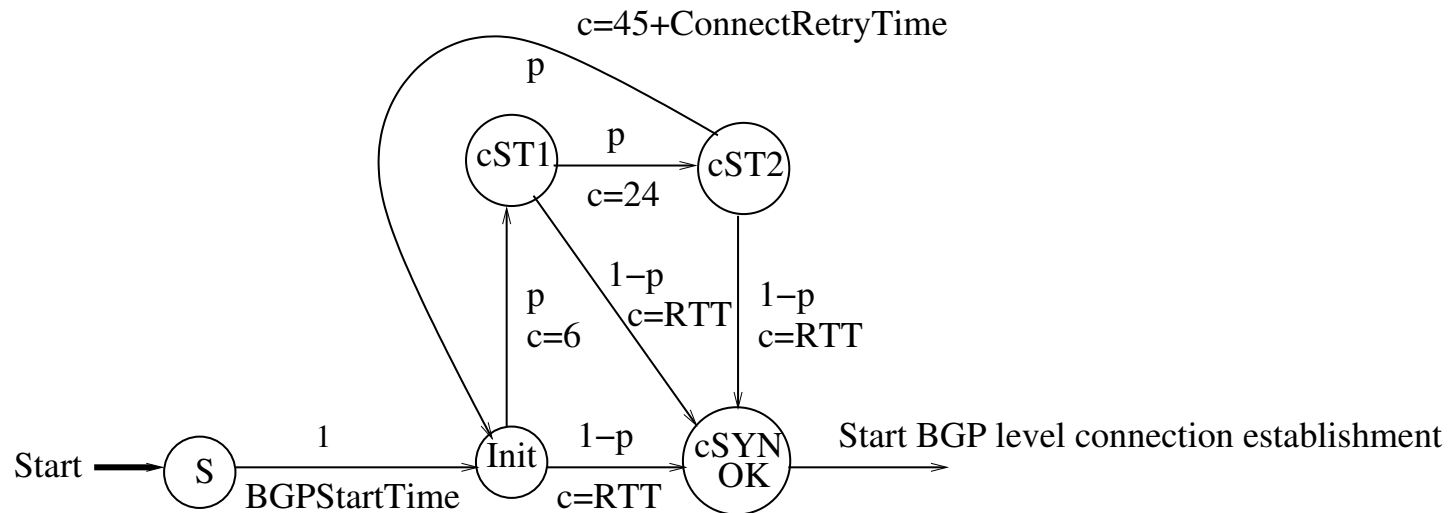
- As RTT increases, less retransmission opportunities for the “keepalive” message.
- Less retransmission opportunities “keepalive” gets, the easier it is for the BGP peering to go down.

BGP model: D2U (recovery) time

- **BGP session establishment has two parts:**
 1. **TCP connection establishment: 3-way handshake.**
 2. **BGP connection establishment.**
- **Two possibilities for TCP connection establishment:**
 1. **HR1 is client; HR2 is server. Congestion in client->server direction.**
 2. **HR2 is client; HR1 is server. Congestion in server->client direction.**
- **Hence BGP session establishment is bi-directional in nature.**
- **We treat them separately.**

BGP D2U model: server→client

• Modeling 3-way handshake of TCP ...

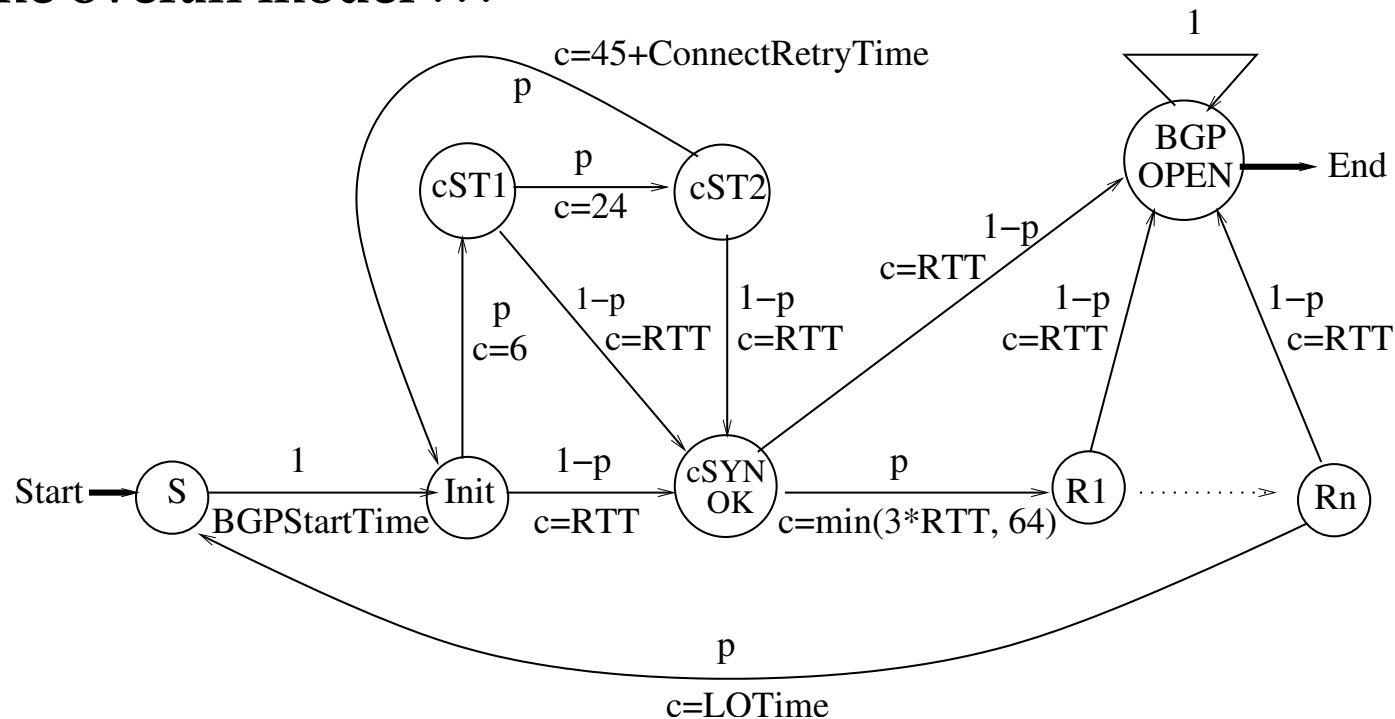


Init HR2's SYN reaches HR1

cSYN OK HR1's SYN ACK has reached HR2 and HR2's ACK comes back

BGP D2U model: server→client (cont'd)

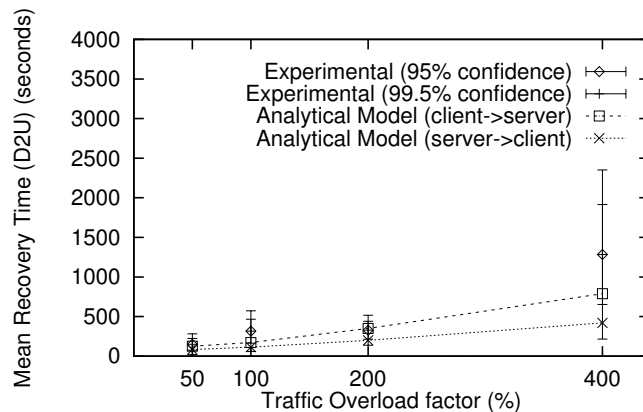
- The overall model ...



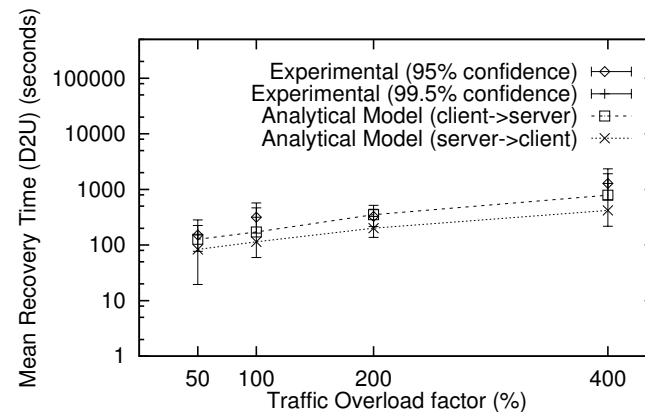
- $E[D2U]$ = Expected time it takes for system to reach BGP OPEN state starting from S.
- $E[D2U]$ depends on BGPStartTime, ConnectRetryTime, Hold-Time and RTT.

BGP model: D2U (recovery) time (cont'd.)

- For our router, BGPStartTime = 60 seconds, ConnectRetryTime = 120 seconds, HoldTime = 180 seconds.
- $RTT = 4 * (1 - p)$ (routers configured to use Drop-from-Front policy).
- Experimental Results: packet size = 256 bytes, buffer size = 4 MB.



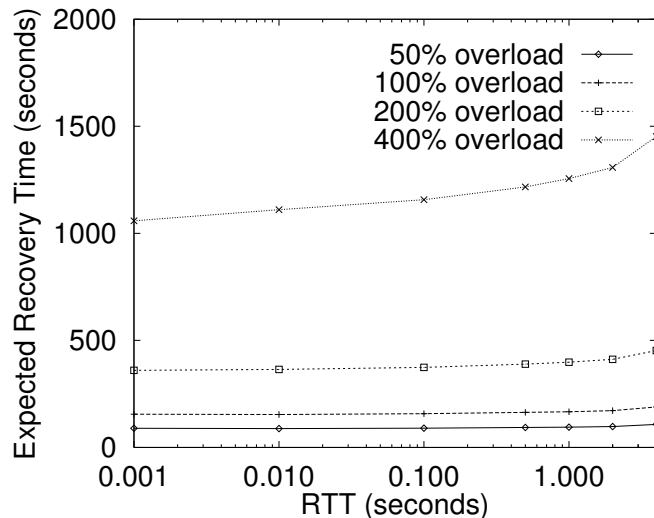
(a) linear scale



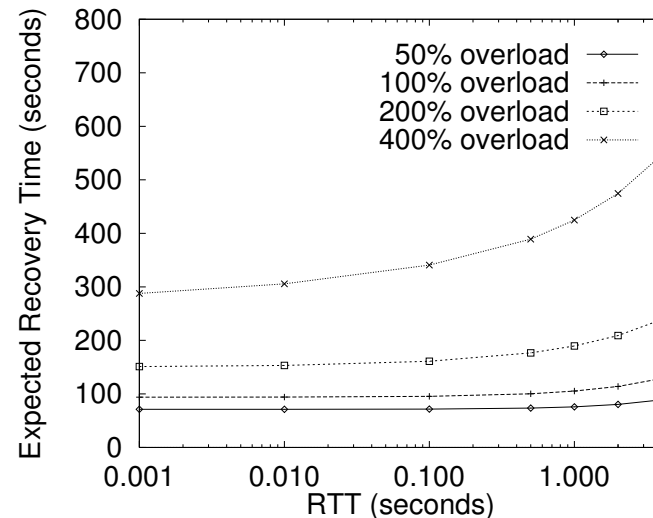
(b) log scale

BGP model: D2U (recovery) time (cont'd.)

- Effect of RTT on $E[D2U]$ of BGP ...



(a) client->server



(b) server->client

- As RTT increases, less retransmission opportunities for the “open” message.
- Less retransmission opportunities “open” gets, the harder it is for the BGP peering to come up.

Conclusions

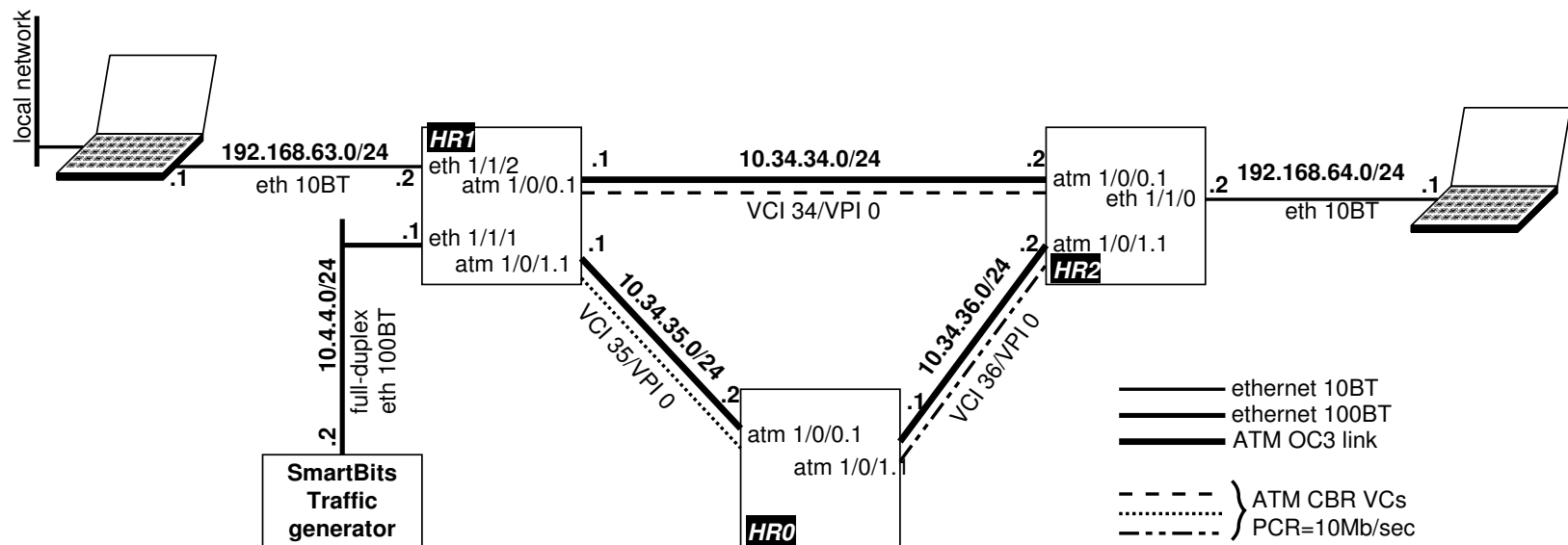
- **Developed detailed analytical models to quantify the behavior of OSPF and BGP in congested networks.**
- **OSPF's behavior depends only on traffic overload factor and is insensitive to packet size, buffer size or packet dropping policy.**
- **Analyzing BGP's behavior is complex because of the use of TCP as the underlying transport protocol.**
- **BGP's behavior depends on overload factor as well as the RTT.**
- **BGP's resilience to congestion decreases as RTT increases.**
- **Experimental results match the analytical results well.**
- **This work shows the need for isolating routing messages from data traffic by using a combination of scheduling and buffering mechanisms both for inter and intra-box communication paths.**

Future work

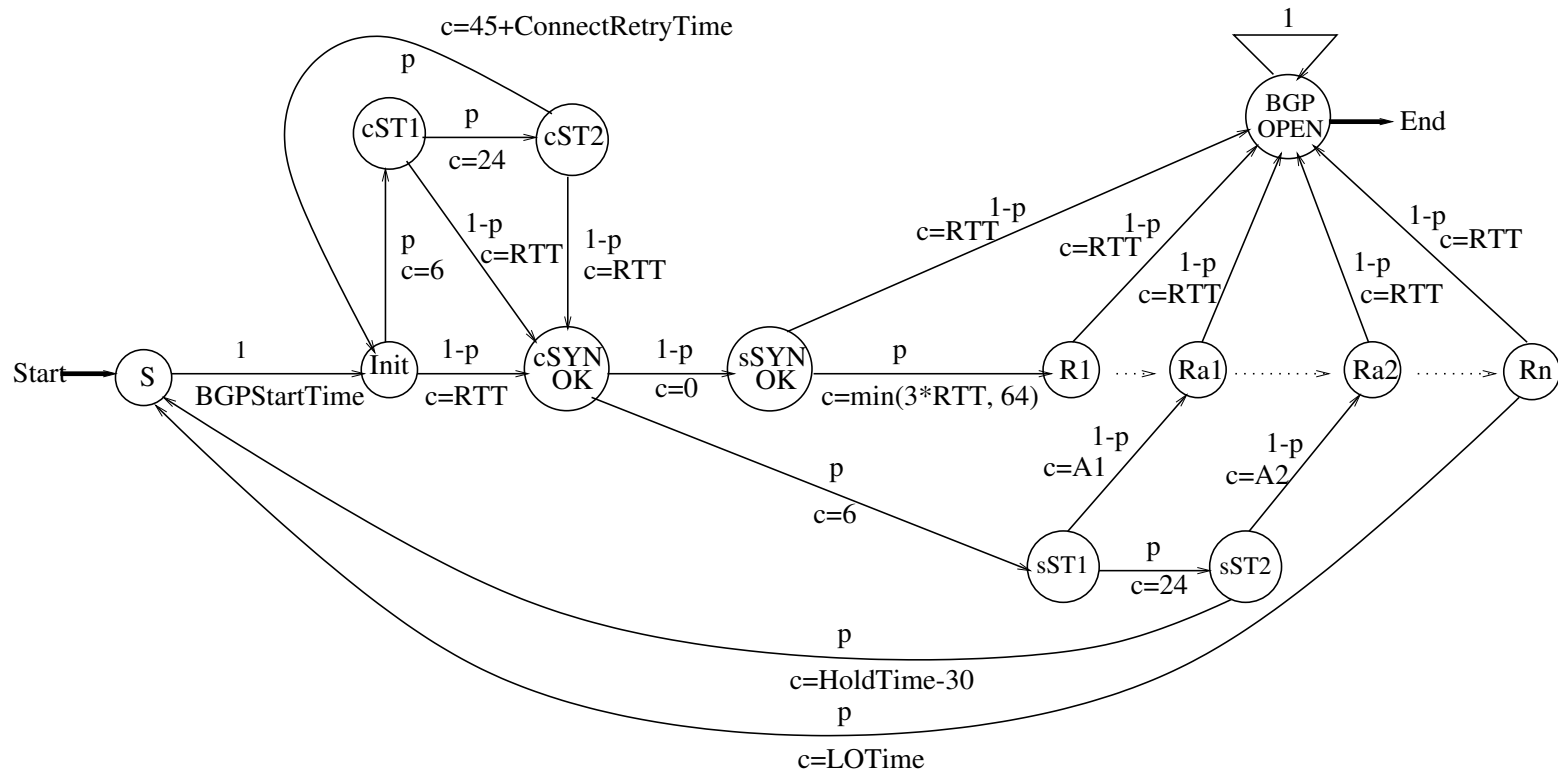
- **Generalize our results for other traffic and loss models.**
- **Tie these results with actual measurements in the Internet.**
- **Labovitz et al. have observed a significant correlation between BGP instability and network usage. See what fraction of this instability is due to congestion and routing packet losses.**
- **Extend the work to include other routing protocols like IS-IS or signaling protocols such as LDP, RSVP, etc.**

Network topology

- Detailed diagram of the topology used in the experiment ...

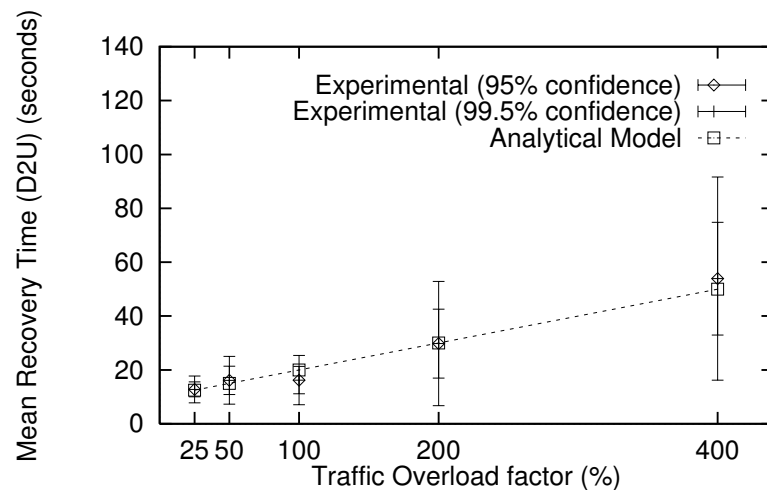


BGP D2U model: client->server

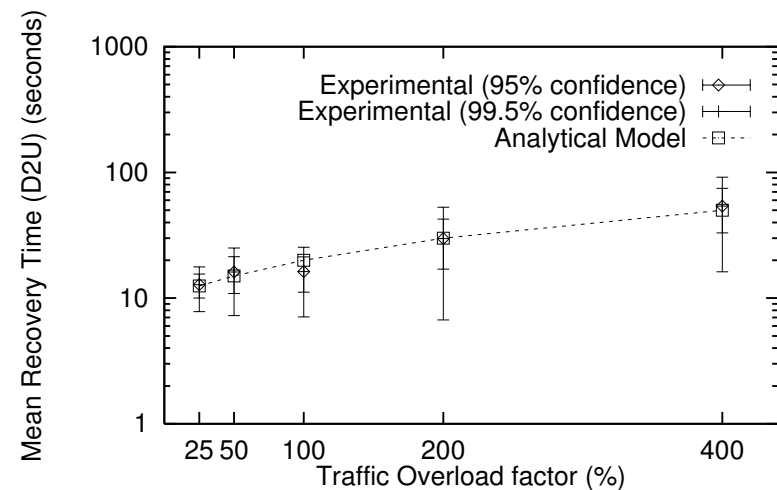


Experimental results

- 2-node OSPF experiment: packet size = 256 bytes, buffer size = 4 MB.
- Recovery time (D2U) versus overload factor.



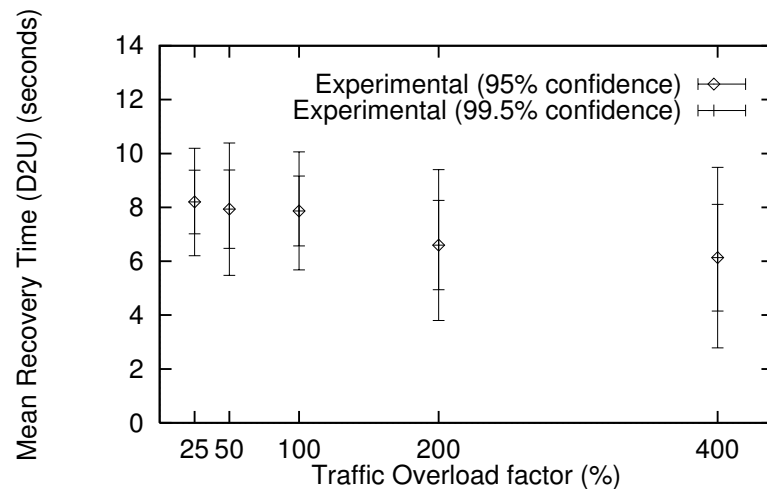
(a) linear scale



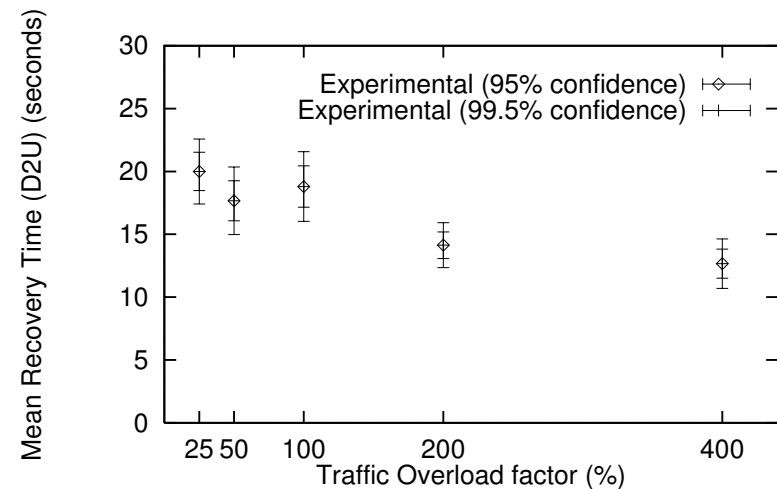
(b) log scale

Experimental results

- 3-node OSPF experiment: packet size = 256 bytes.
- Recovery time (D2U) versus overload factor for two buffer sizes (4 and 16 MB).



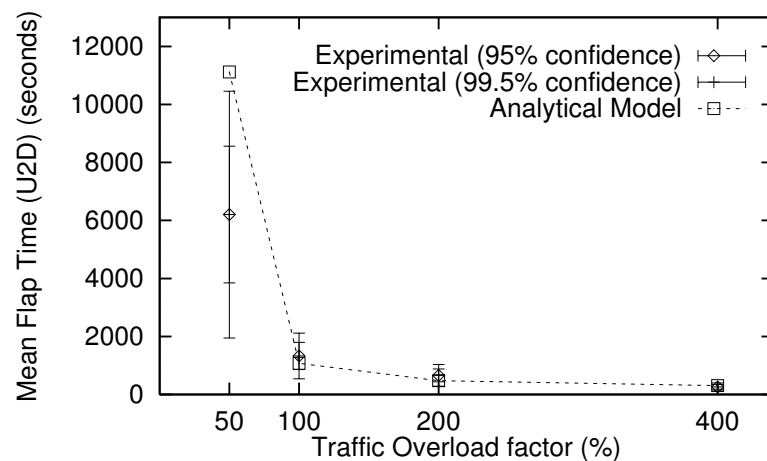
(a) Buffer Size = 4 MB



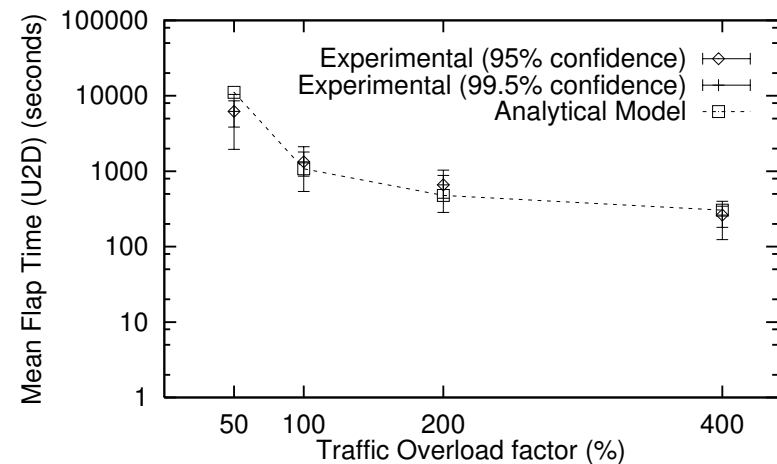
(b) Buffer Size = 16 MB.

Experimental results

- 2-node BGP experiment: packet size = 256 bytes, buffer size = 4 MB.
- Flap time (U2D) versus overload factor.



(a) linear scale



(b) log scale