Implementation of BGP in a Network Simulator

Tony Dongliang Feng
tdfeng@cs.sfu.ca

Communication Networks Laboratory
Simon Fraser University

Road map

- Introduction
- Background
- Design and implementation of ns-BGP
- Validation tests
- Scalability analysis
- Conclusions

Road map

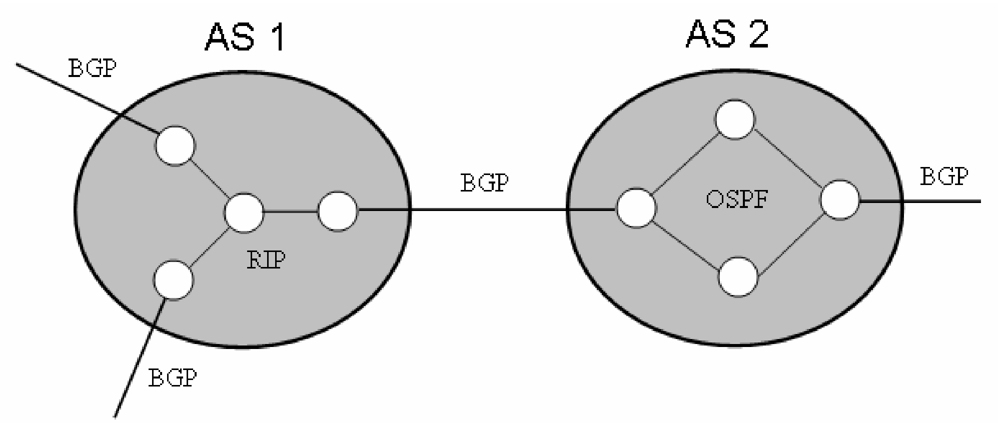
- Introduction
- Background
- Design and implementation of ns-BGP
- Validation tests
- Scalability analysis
- Conclusions

Introduction

- Internet routing
- Border Gateway Protocol (BGP) weaknesses
- Employed Approaches
- Contributions

Internet routing

- The Internet is organized as a collection of interconnected Autonomous Systems (AS)
- Routing in the Internet is performed on two levels
  - IGP: Interior Gateway Protocol (Intra-domain)
    - OSPF, IS-IS, EIGRP, RIP
  - EGP: Exterior Gateway Protocol (Inter-domain)
    - BGP



BGP weaknesses

- Poor integrity
  - vulnerable to malicious attacks and misconfiguration
- Slow convergence
  - up to tens of minutes
- Divergence
  - conflicts of routing policies can cause BGP to diverge, resulting in persistent route oscillations

Approaches

- Empirical measurements
  - expensive set-up
  - inflexible
- Theoretical analysis
  - highly simplified
  - inadequate in practical scenarios
- Simulations
  - full control over the system and flexible
  - cost effective
  - controlled experiments

Contributions

- Implemented a BGP-4 model (ns-BGP) in ns-2
- Implemented TcpSocket and IPv4Classifier in ns-2
- Validated the ns-BGP model
- Analyzed the scalability of ns-BGP

Road map

- Introduction
- Background
- Design and implementation of ns-BGP
- Validation tests
- Scalability analysis
- Conclusions

Background

- BGP overview
- Network simulator ns-2
- BGP implementation in SSFNet
- Related work

BGP version 4

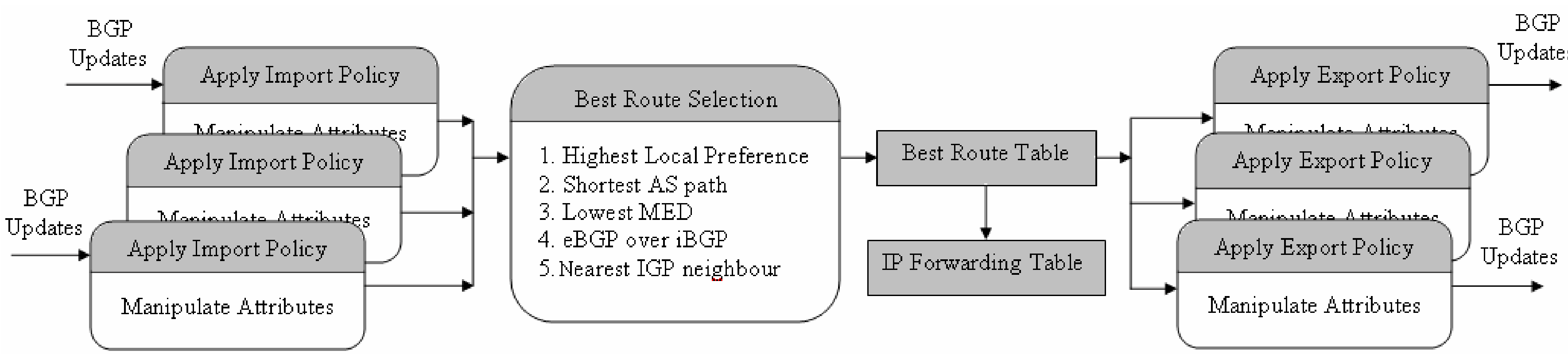
- RFC 1771, “A Border Gateway Protocol 4”, March 1995
- The *de facto* inter-domain routing protocol of the Internet
- Path vector protocol
- Incremental
- Relies on TCP

Four types of BGP messages

- Open: establish a peering session
- Keep alive: handshake at regular intervals
- Notification: report errors, shut down a peer session
- Update: announce new routes or withdraw previously announced routes
  - advertisement
    - destination prefix
    - route attributes (local preference, AS path)

Route processing

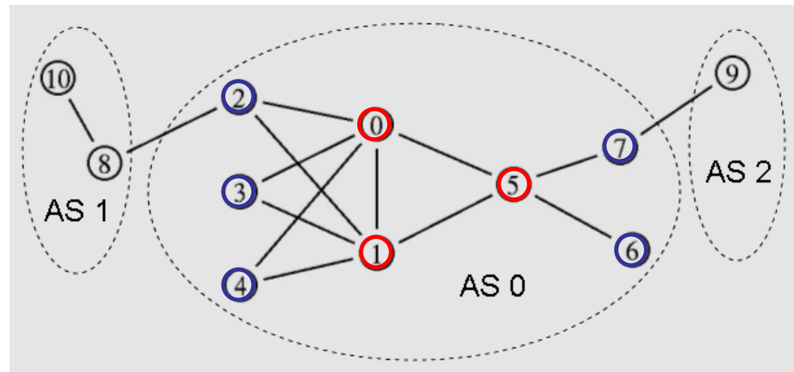
- Apply import policy
- Select a best route
- Install the best route
- Apply export policy and send out updates



MED: Multiple Exit Discriminator

BGP route reflection

- Two types of BGP peer connections:
  - external BGP (eBGP) connection
  - internal BGP (iBGP) connection
- BGP routers within an AS are required to be fully meshed with iBGP connections
- Route reflection provides one way to address the scalability issue of iBGP



- reflector ○
- client ○

Network Simulator ns-2

- One of the most popular network simulators
- Object oriented
  - written in C++ and OTcl
- Substantial support for TCP, routing, and multicast protocols
- Graphical animator: nam

SSF.OS.BGP4: BGP implementation in SSFNet

- Scalable Simulation Framework Network Models (SSFNet) is a Java-based simulator
- SSF.OS.BGP4 is developed and maintained by Brian J. Premore from Dartmouth College
- We implemented a BGP-4 model (ns-BGP) in ns-2 by porting the BGP implementation from SSFNet

Related work

- OPNET BGP model
  - the difference between OPNET and ns-2
- BGP daemon of GNU Zebra
  - object oriented paradigm
- J-Sim BGP model
  - also ported from SSFNet

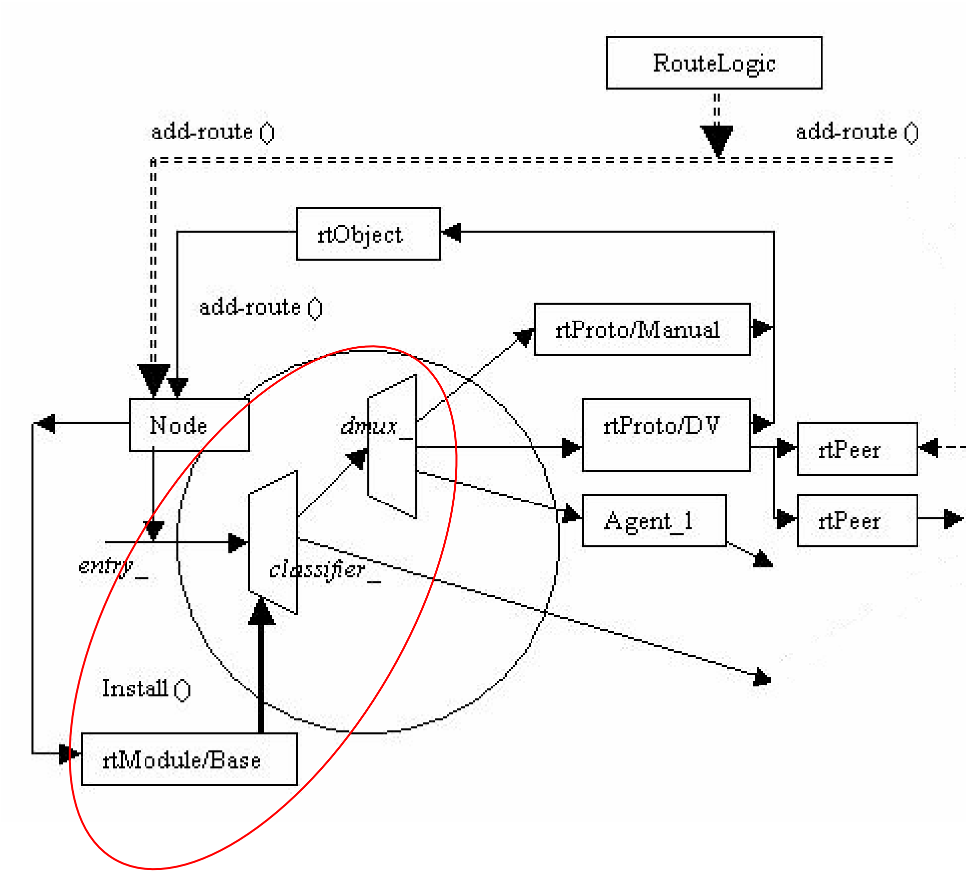
Road map

- Introduction
- Background
- Design and implementation of ns-BGP
- Validation tests
- Scalability analysis
- Conclusions

ns-2 unicast routing structure

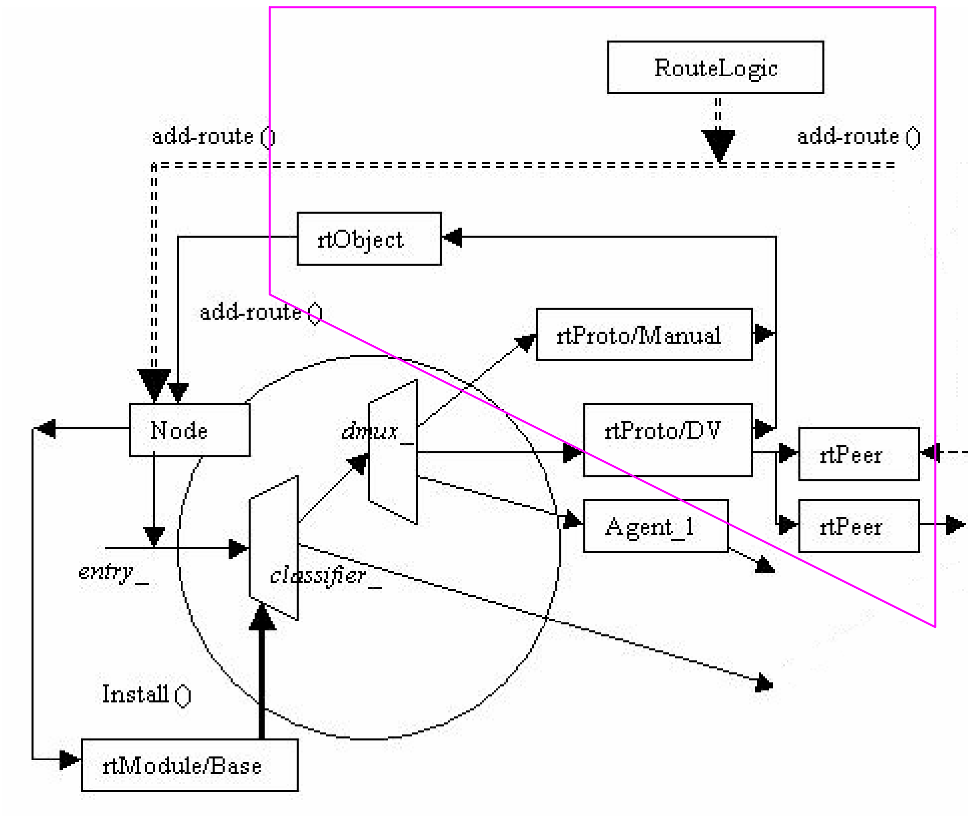
- Forwarding plane:
  - classify and forward packets
- Control plane:
  - routing info exchange, route computation, routing table creation and maintenance

Forwarding plane



- Classifier (classifer_):
  - delivers the incoming packets either to the correct agent or to the outgoing link
- Routing Module (rtModule):
  - manages a node's classifier and provides an interface to the control plane

Control plane

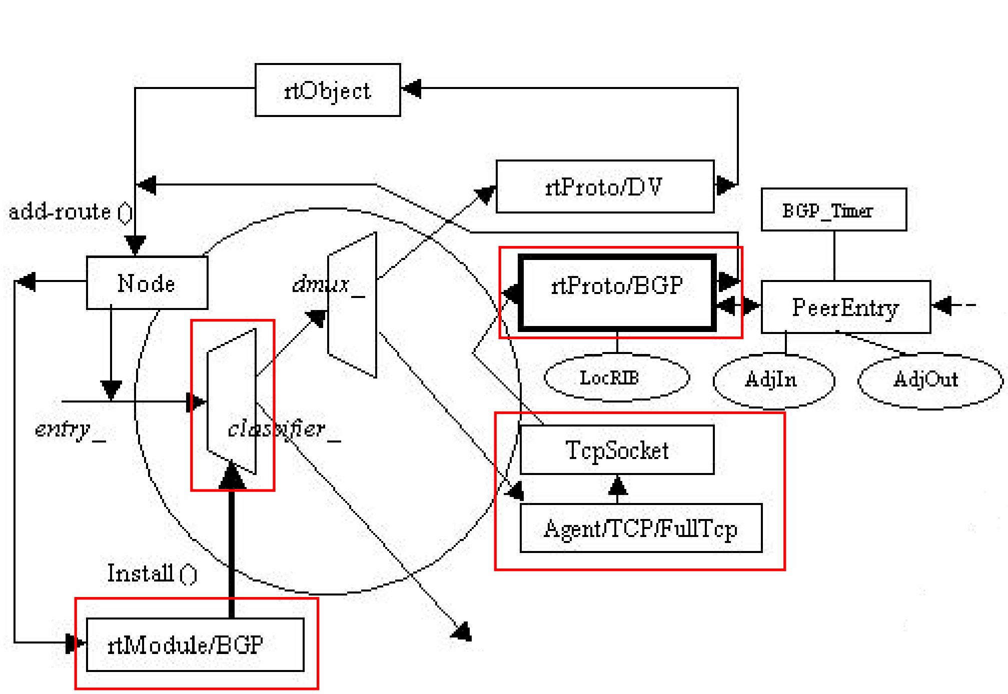


- Route logic (RouteLogic):
  - the centrally created routing table
- Routing protocol (rtProto):
  - manual, DV, LS
  - implements specified routing algorithm
- Route peer (rtPeer):
  - stores the metric and preference for each route it advertised
- Route object (rtObject):
  - a coordinator for the node's routing instances

Modifications to ns-2

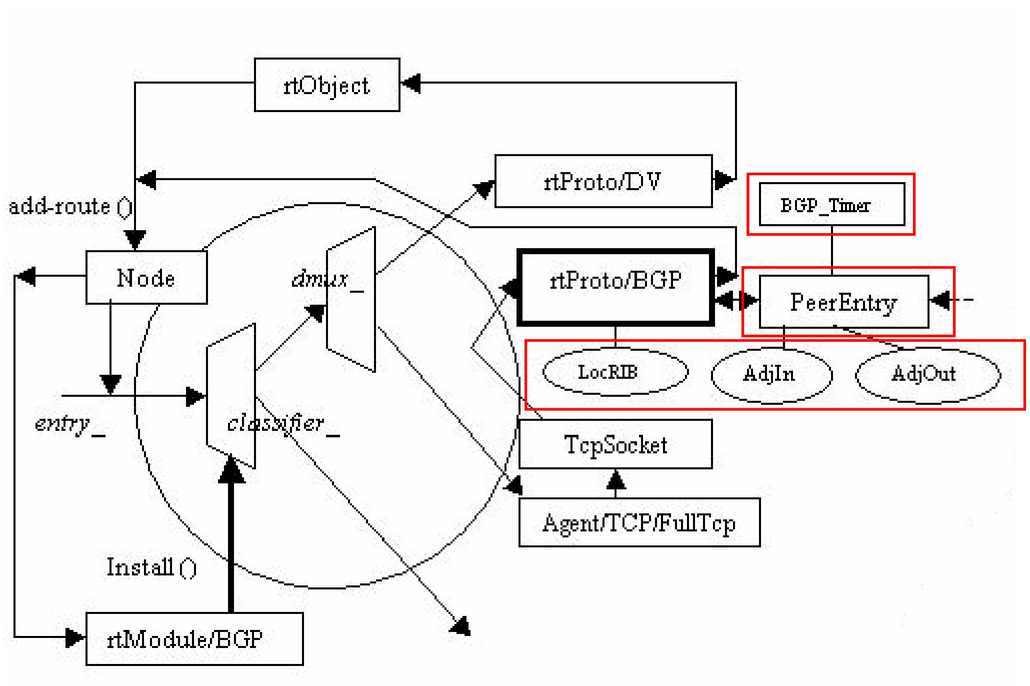
- No socket layer in current ns-2:
  - Solution: we ported to ns-2 TcpSocket - the socket layer implementation of SSFNet
- Simplified packet transmission:
  - Solution: we modified FullTcpAgent, the TCP agent for TcpSocket to support data transmission
- No support for IPv4 addressing and packet forwarding schemes:
  - Solution: we created a new address classifier IPv4Classifier

ns-BGP unicast routing structure



- IPv4Classifier (*classfier_*)
- BGP routing model (rtModule/BGP):
  - manages the IPv4Classifier
- TcpSocket:
  - encapsulating the TCP services into a socket interface
- BGP routing protocol (rtProto/BGP):
  - performs BGP operations

ns-BGP unicast routing structure



- BGP peer (PeerEntry):
  - establishes and closes a peer session, exchanges messages with a peer
- BGP routing tables (LocRIB, AdjIn, and AdjOut):
  - correspond to the BGP Routing Information Base (RIB): Loc-RIB, Adj-RIB-In, and Adj-RIB-Out
- BGP Timer (BGP_Timer):
  - provides supports for the BGP timing features (timers)

Road map

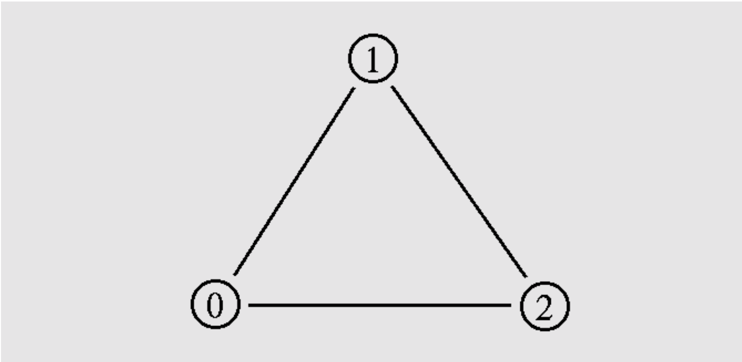
- Introduction
- Background
- Design and implementation of ns-BGP
- Validation tests
- Scalability analysis
- Conclusions

Validation tests

- Route selection:
  - validates that a BGP speaker chooses routes properly when it knows more than one path to a specific destination
- Route reflection:
  - validates the behavior of multiple reflectors inside a BGP cluster

Route selection validation test

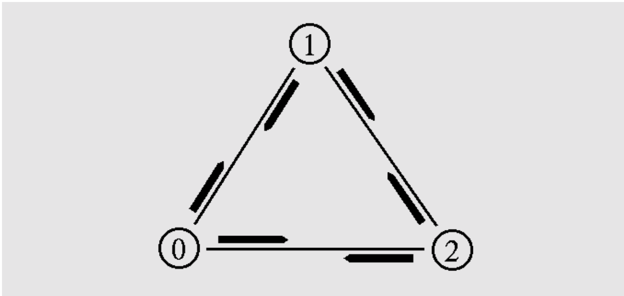
- Network topology



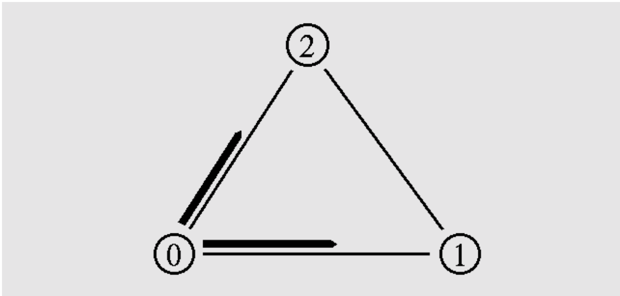
- The network contains three ASs:
  - AS 0 contains node 0 (10.0.0.1)
  - AS 1 contains node 1 (10.1.1.1)
  - AS 2 contains node 2 (10.2.2.1)
- Addressing scheme:
  - 10.(AS number).(node number).1
- BGP configuration:
  - BGP agents are configured for each node
  - BGP agents are fully meshed with eBGP connections
- Event scheduling:
  - 0.25 s: node 0 announces a route for 10.0.0.0/24
  - 39 s: display the contents of all routing tables
  - 40 s: simulation terminates

Simulation results: nam snapshots

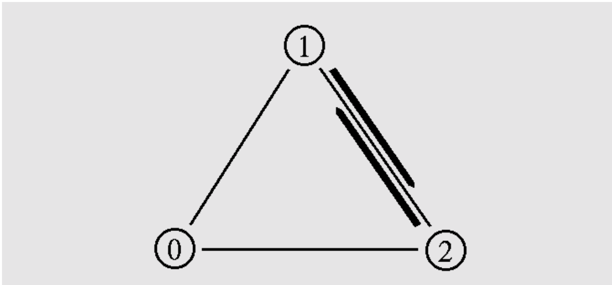
- 0.0503 s, TCP SYN segments are exchanged



- 0.2507 s, node 0 originates update messages for network 10.0.0.0/24



- 0.2525 s, nodes 1 and 2 propagate the routes to each other.



Simulation results: routing tables

Node 1 and node 2 both chose the route directly received from node 0.

```
BGP routing table of node1
BGP table version is 1, local router ID is 10.1.1.1
Status codes: * valid, > best, i - internal.
     Network          Next Hop       Metric  LocPrf  Weight Path
*>   10.0.0.0/24       10.0.0.1          -       -        -      0
```

```
BGP routing table of node2
BGP table version is 1, local router ID is 10.2.2.1
Status codes: * valid, > best, i - internal.
     Network          Next Hop       Metric  LocPrf  Weight Path
*>   10.0.0.0/24       10.0.0.1          -       -        -      0
```

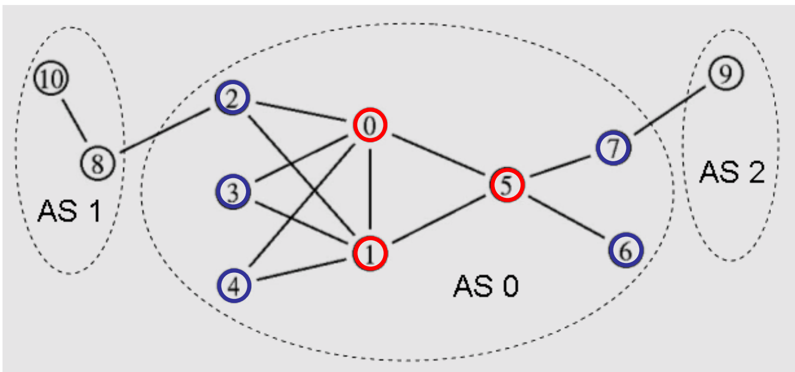
node name

AS path

destination address

Route reflection validation test

- Network topology



- The network contains three ASs:
  - AS 0 has eight nodes (0 to 7), with IP addresses 10.0.0.0 - 10.0.7.0
  - AS 1 has two nodes (8 and 10), with IP addresses 10.1.8.0 and 10.1.10.0
  - AS 2 has a single node (9), with IP address 10.2.9.0
- Addressing scheme:

  10.(AS number).(node number).1

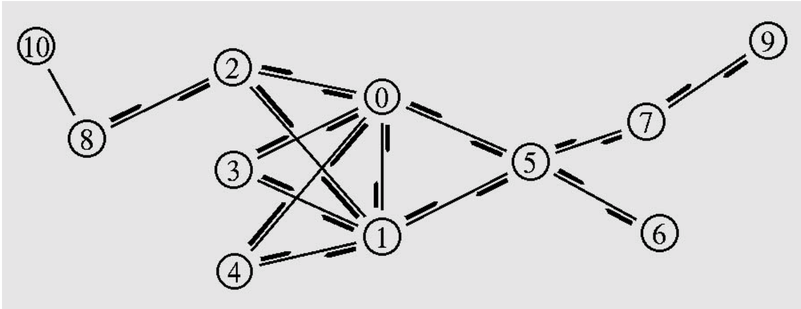
- BGP configuration:
  - AS 0 contains two clusters (0 and 1).
    - cluster 0 (nodes 0 – 4) contains 2 reflectors: nodes 0 and 1, with nodes 2, 3, and 4 as their clients
    - cluster 1 (nodes 5 -7) has one reflector (node 5), with nodes 6 and 7 as its clients
    - The three reflectors (nodes 0, 1, and 5) are fully connected via iBGP connections
  - eBGP connections:
    - nodes 2 and 8
    - nodes 7 and 9

Traffic source and event scheduling

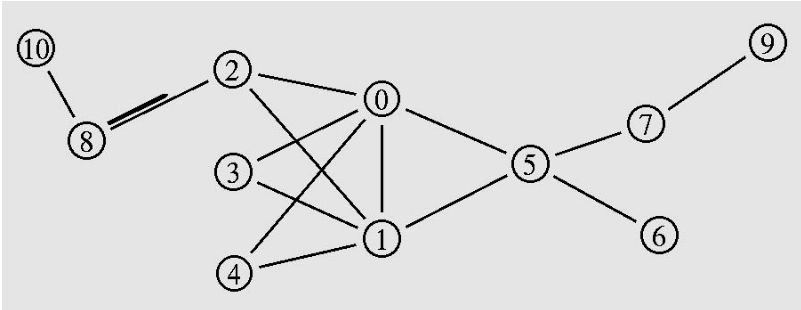
- Traffic source:
  - attached to node 4
  - constant bit rate (CBR)
  - transport protocol: UDP
  - sends segments of 20 bytes/ms to node 10 (10.1.10.1).
- Event scheduling:
  - traffic source begins sending at 0.23 s and stops at 20.0 s
  - 0.25 s: node 8 sends a route advertisement for network 10.1.10.0/24 that is within its AS (AS 1)
  - 0.35 s: node 9 sends a route advertisement for network 10.2.9.0/24
  - 39.0 s: displays all routing tables for BGP agents
  - 40.0 s: the simulation terminates

Simulation results: nam snapshots (1)

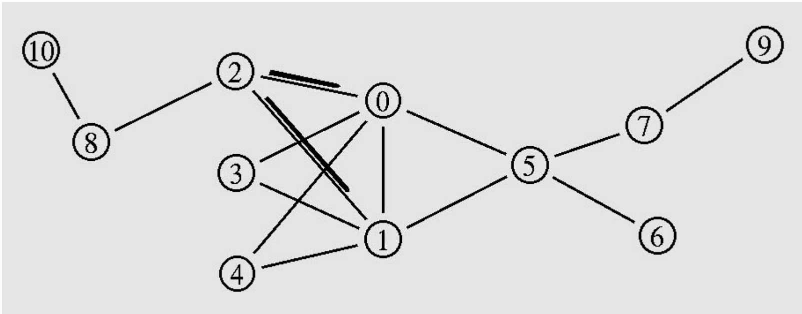
- 0.0503 s, TCP SYN segments are exchanged



- 0.2505 s, node 8 originates an update message for network 10.1.10.0/24

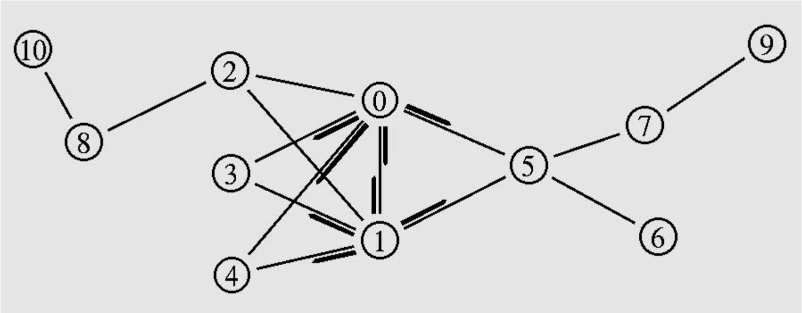


- 0.2525 s, node 2 propagates the route to nodes 0 and 1

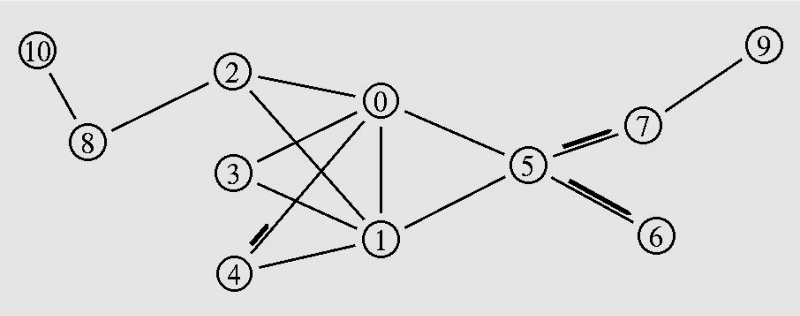


Simulation results: nam snapshots (2)

- 0.2561 s, nodes 0 and 1 reflect the route to nodes 3 and 4 and to their iBGP peers

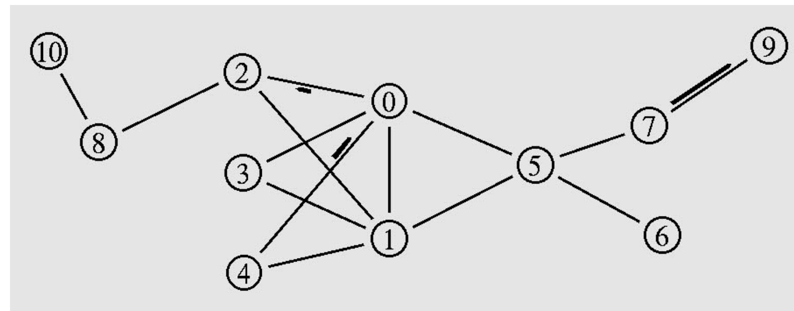


- 0.2568 s, node 5 reflects the route to nodes 6 and 7. Node 4 now knows the route to network 10.1.10.0/24, the UDP segment will be forwarded to node 10

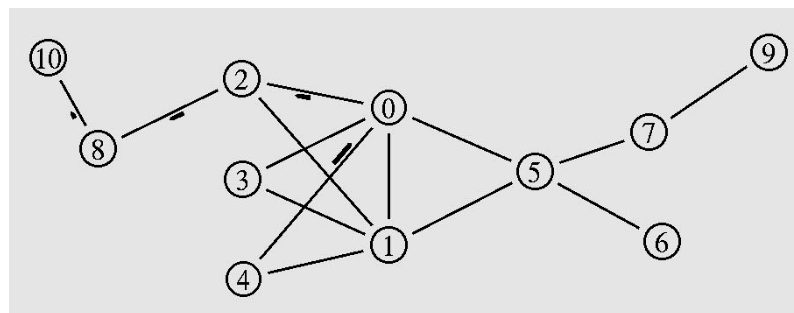


Simulation results: nam snapshots (3)

- 0.2578 s, the second UDP segment is sent to the node 10. Node 7 propagates the route to node 9



- 0.2580 s, UDP segments are delivered to node 10



Simulation results: routing tables

All ten Nodes learned the routes to IP addresses 10.1.10.0/24 and 10.2.9.0/24.

BGP routing table of node0
BGP table version is 2, local router ID is 10.0.0.1
Status codes: * valid, > best, i - internal.

	Network	Next Hop	Metric	LocPrf	Weight	Path	
*>	10.1.10.0/24	10.0.2.1	-	-	-	1	i
*>	10.2.9.0/24	10.0.7.1	-	-	-	2	i

.
.
.

BGP routing table of node9
BGP table version is 3, local router ID is 10.2.9.1
Status codes: * valid, > best, i - internal.

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	10.1.10.0/24	10.0.7.1	-	-	-	0 1
*>	10.2.9.0/24	self	-	-	-	

Road map

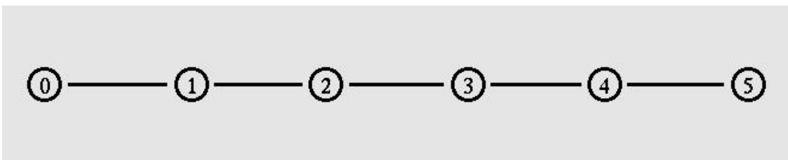
- Introduction
- Background
- Design and implementation of ns-BGP
- Validation tests
- Scalability analysis
- Conclusions

Scalability analysis

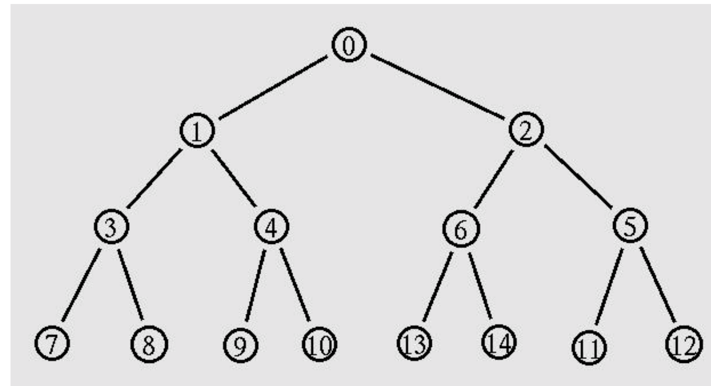
- Scalability properties:
  - execution speed
  - memory requirements
- Model configuration:
  - topology families
  - simulation phases
  - measurements
  - ns-2 calendar scheduler
- Scalability: number of peer sessions
- Scalability: size of routing tables
- Hardware platform:

  1.6 GHz Xeon host with 2 GBytes of memory

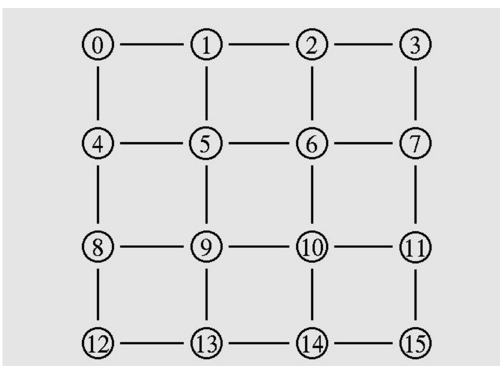
Topologies



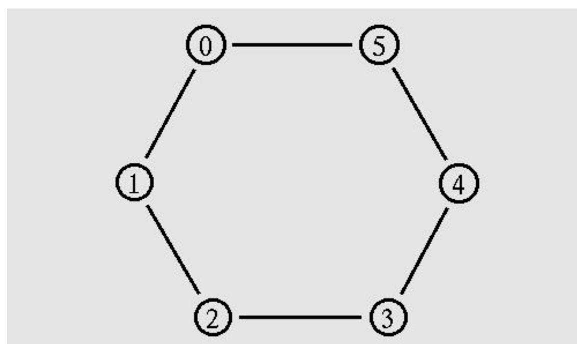
line



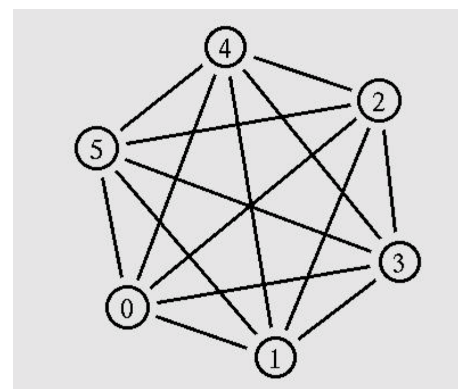
binary tree



grid



ring



clique

Simulation phases

- Phase 1: ns-2 simulator instance creation
- Phase 2: node creation
- Phase 3: link creation
- Phase 4: enabling auto-configuration
- Phase 5: simulator initialization
- Phase 6: BGP session establishment
- Phase 7: BGP message exchange

Measurements

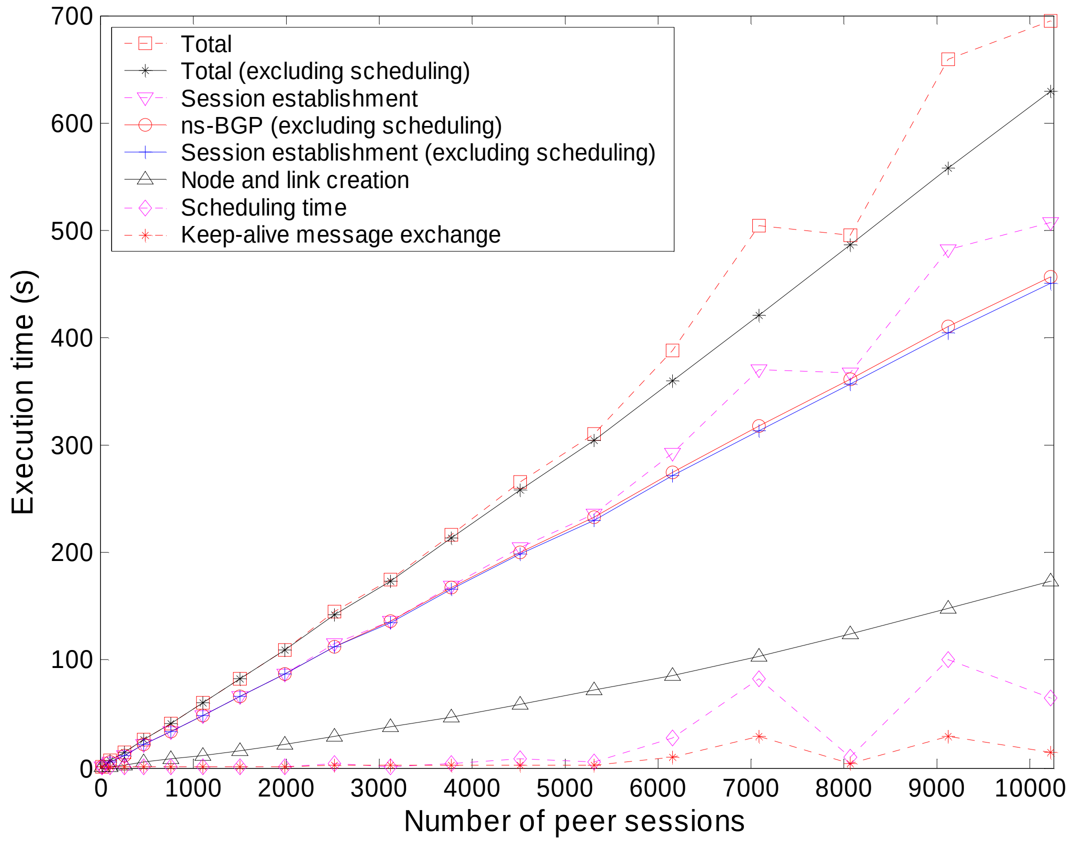
- Execution time and memory usage
- Total
  - Phases 1 to 7
- Node and link creation
  - Phase 2: node creation
  - Phase 3: link creation
- ns-BGP
  - Phase 6: BGP session establishment
  - Phase 7: BGP message exchange

ns-2 calendar scheduler

- Performance is affected by the distribution of the event times
- Large number of events scheduled at the same time instance can cause the scheduling time to increase exponentially
- Solution: we jittered BGP timers (start-up, keep-alive) to scatter simulation events
- While the jittered scheduling times no longer increase exponentially, they are affected by the introduced jitter factors

Execution time vs. number of peer sessions

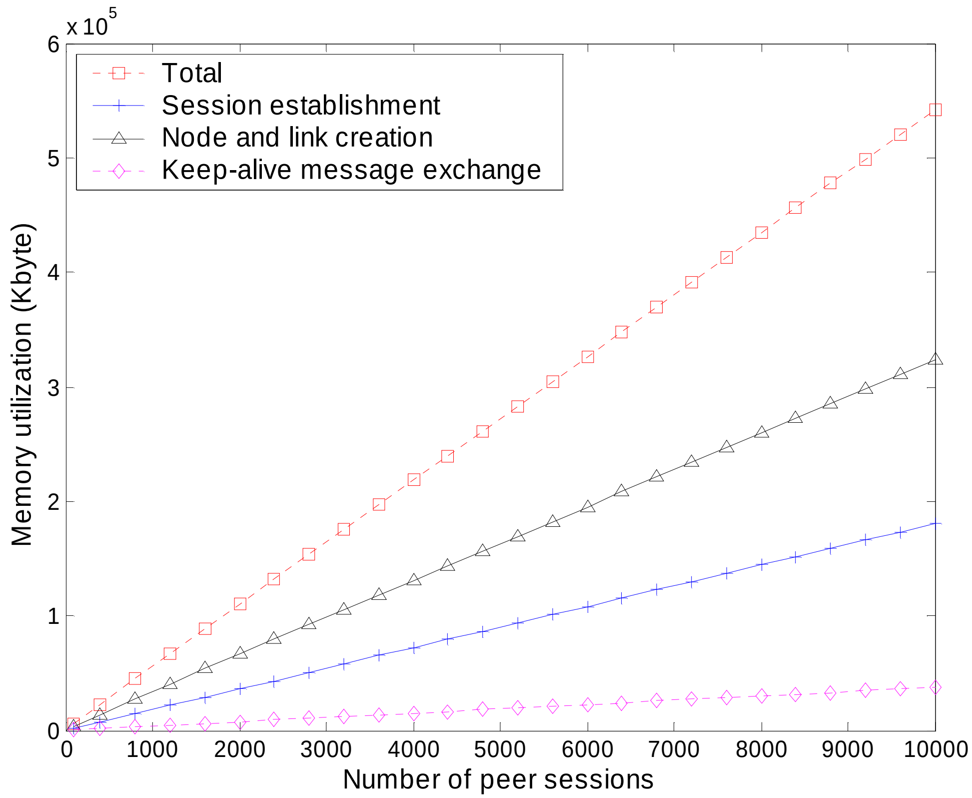
- Line topology
  - total execution time
  - scheduling time
  - ns-BGP (excluding scheduling) execution time increases linearly
  - node and link creation time
- Ring, binary tree, grid, and clique topology
  - ns-BGP (excluding scheduling) execution times increase linearly



Memory utilization vs. number of peer sessions

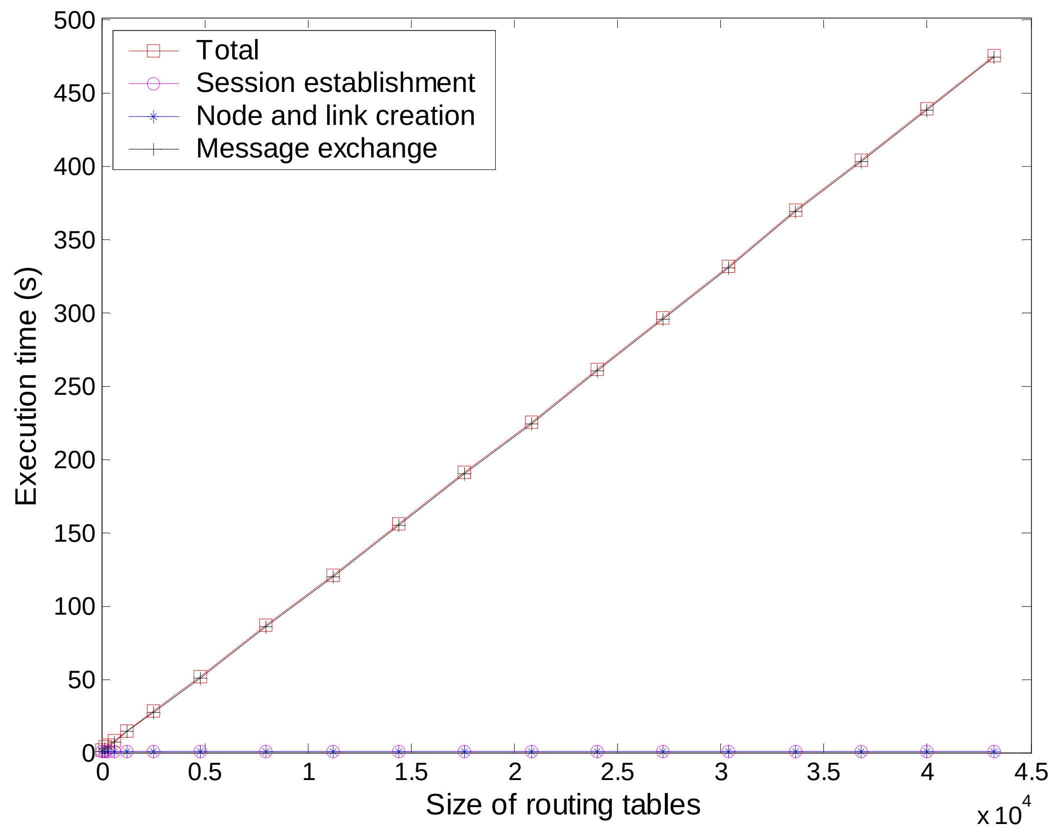
- Line topology:
  - memory usage increases linearly
  - 54.21 Kbytes/peer
- Ring, binary tree, grid, and clique topology:
  - memory usage increases linearly

Topology	Total memory usage (Kbytes/peer)
Line	54.21
Ring	54.21
Binary tree	54.3
Grid	47.1
Clique	43.8



Execution time vs. size of routing tables

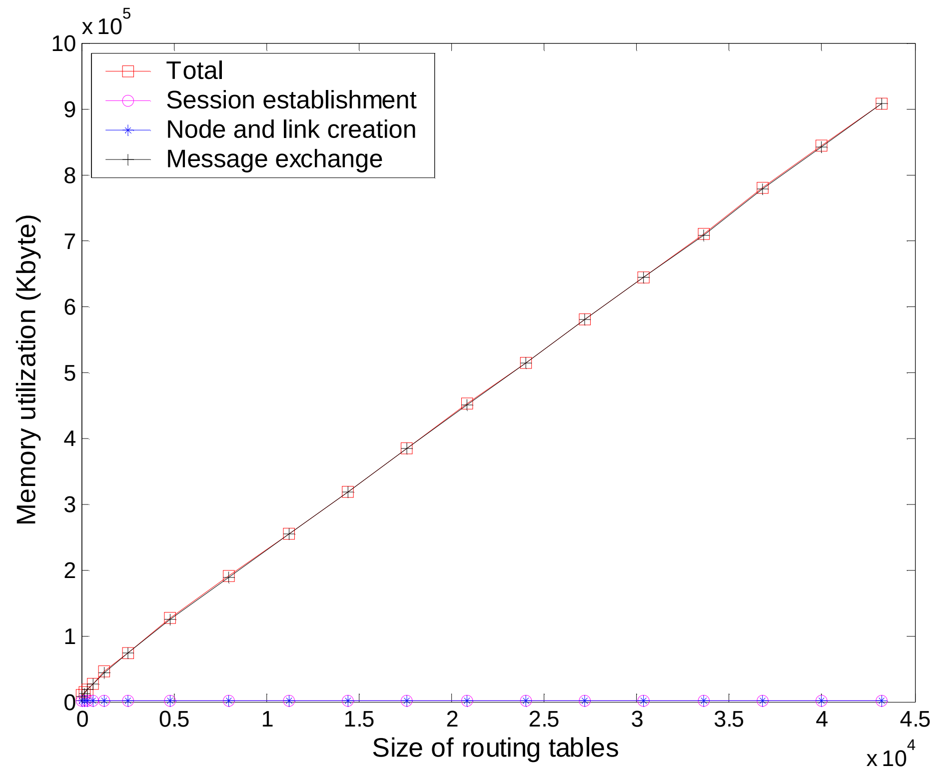
- Topologies:
  - line, ring, grid, and clique: size 16
  - binary tree: size 15
- Line topology:
  - total execution time increases linearly
  - node and link creation time
- Ring, binary tree, grid, and clique topology:
  - total execution time increases linearly



Memory utilization vs. size of routing tables

- Line topology
  - memory usage increases linearly
  - 20.88 Kbytes/route
- Ring, binary tree, grid, and clique topology
  - memory usage increases linearly

Topology	Total memory usage (Kbytes/route)
Line	20.88
Ring	24.97
Binary tree	19.28
Grid	60.54
Clique	67.25



Conclusions

- We presented the architecture and implementation of ns-BGP, a BGP-4 model for the ns-2 network simulator.
- ns-BGP enables simulation and evaluation of BGP protocol and its variants.
- Validation tests illustrated the validity of the ns-BGP implementation.
- Our scalability analysis showed that the internal data structures and employed algorithms are scalable with respect to the number of peer sessions and the size of routing tables.
- New features, such as route flap damping, could be added in the future.

Selected BGP RFCs

[1] T. Bates, R. Chandra, and E. Chen, "BGP route reflection – an alternative to full mesh IBGP," RFC 2796, April 2000.

[2] T. Bates, Y. Richter, R. Chandra, and D. Katz, "Multiprotocol extensions for BGP-4," RFC 2858, June 2000.

[3] Y. Rekhter and T. Li, "A border gateway protocol 4 (BGP-4)," RFC 1771, March 1995.

[4] C. Villamizar, R. Chandra, and R. Govindan, "BGP route flap damping," RFC 2439, November 1998.

Internet routing and BGP dynamics (1)

[5] T. Bu, L. Gao, and D. Towsley, "On routing table growth," in *Proc. Of Global Internet Symposium*, Taipei, Taiwan, November 2002.

[6] N. Feamster and H. Balakrishnan, "Towards a logic for wide-area Internet routing," in *Proc*. *SIGCOMM*, Karlsruhe, Germany, August 2003, pp. 88-100.

[7] L. Gao and J. Rexford. "Stable internet routing without global coordination," in *Proc*. *SGIMETRICS*, pp. 307-317.

[8] T. Griffin and B. Premore, "An experimental analysis of BGP convergence time," in *Proc. ICNP*, Riverside, CA, November 2001, pp. 53-61.

[9] T. Griffin and G. Wilfong, "An analysis of BGP convergence properties," in *Proc*. *SIGCOMM*, August 1999, pp. 277-288.

[10] T. Griffin and G. Wilfong, " A safe path vector protocol," in *Proc*. *INFOCOM*, April 2001, pp. 490-499.

[11] T. Griffin and F. Shepherd, G. Wilfong, "Policy disputes in path-vector protocols," in *Proc*. *ICNP*, October 1999, pp. 21-30.

[12] T. Griffin, F. Shepherd, and G. Wilfong, "The stable paths problem and interdomain routing," *IEEE Transactions on Networking*, vol. 10, no. 2, April 2002, pp. 232-243.

[13] S. Halabi and D. McPherson, *Internet Routing Architectures*. Indianapolis, IN: Cisco Press, 2000.

[14] C. Huitema, *Routing in the Internet*. Upper Saddle River, NJ: Prentice Hall, 2000.

Internet routing and BGP dynamics (2)

[15] Z. Mao, R. Govindan, G. Varghese, and R. Katz. "Route flap damping exacerbates Internet routing convergence," in *Proc. SIGCOM*, Pittsburgh, PA, August 2002, pp. 221-233.

[16] S. Murphy, "BGP security vulnerabilities analysis," Internet draft, June 2003.

[17] D. Nicol, "Scalability of network simulators revisited," in *Proc of CNDS*, Orlando, FL, February 2003.

[18] C. Labovitz, G. Malan, and F. Jahanian "Origins of Internet routing instability," in *Proc*. *INFOCOM*, New York, NY, March 1999, pp. 218-226.

[19] C. Labovitz, R. Wattenhofer, S. Venkatachary, and A. Ahuja, "The impact of Internet policy and topology on delayed routing convergence," in *Proc. INFOCOM*, Anchorage, AK, April 2001, pp. 537-546.

[20] B. Premore, *An Analysis of Convergence Properties of the Border Gateway Protocol Using Discrete Event Simulation*, PhD thesis, Dartmouth College, May 2003.

[21] J. Stewart III. BGP4: *Inter-Domain Routing in the Internet*, Addison-Wesley, 1998.

[22] K. Varadhan, R. Govindan, and D. Estrin, "Persistent route oscillations in inter-domain routing," Univ. of Southern California Information Sciences Inst., Los Angeles, Tech. Rep. 96-631, February 1996.

BGP software

[23] T. D. Feng, R. Ballantyne, and Lj. Trajkovic, "Implementation of BGP in a network simulator," to be presented at the Applied Telecommunication Symposium, ATS '04, Arlington, Virginia, April 2004.

[24] BGP++: http://www.ece.gatech.edu/research/labs/MANIACS/BGP++. Accessed: April 10, 2004.

[25] GNU Zebra: http://www.zebra.org. Accessed: April 10, 2004.

[26] GNU Zebra BGP daemon: http://www.zebra.org/zebra/BGP.html#BGP. Accessed: April 10, 2004.

[27] ns manual: http://www.isi.edu/nsnam/ns/doc/index.html. Accessed: April 10, 2004.

[28] OPNET BGP: http://www.opnet.com/products/library/bgp.html. Accessed: April 10, 2004.

[29] B. Premore, SSFNet BGP User's Guide: http://www.ssfnet.org/bgp/user-guide-ps.zip. Accessed: April 10, 2004.

[30] SSFNet: http://www.ssfnet.org/homePage.html. Accessed: April 10, 2004.

Acknowledgements

- Special thanks to my supervisor, Dr. Ljiljana Trajković, for her guidance, support and encouragement during my studies in Simon Fraser University.
- Zhang Wang for the implementation of TcpSocket
- Rob Balantyne for the contribution of IPv4Classifier
- Brian J. Premore for providing the SSF.OS.BGP4 implementation
- Fellow CNL members for their support and wonderful friendship