Implementation of BGP in a Network Simulator

Tony Dongliang Feng   Rob Ballantyne   Ljiljana Trajković

Communication Networks Laboratory

http://www.ensc.sfu.ca/cnl

Simon Fraser University

Road map

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

Road map

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

Introduction

- Internet routing
  - Autonomous Systems
  - IGP: Interior Gateway Protocol (Intra-domain)
  - EGP: Exterior Gateway Protocol (Inter-domain)
- Border Gateway Protocol (BGP) weaknesses
  - routing instability
  - inefficient routing
  - scalability issues
- Employed approaches
  - empirical measurements
  - theoretical analysis
  - simulations

Internet routing

- 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

Road map

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

Background

- BGP version 4
- Network simulator ns-2
- BGP implementation in SSFNet
- Related work

BGP version 4

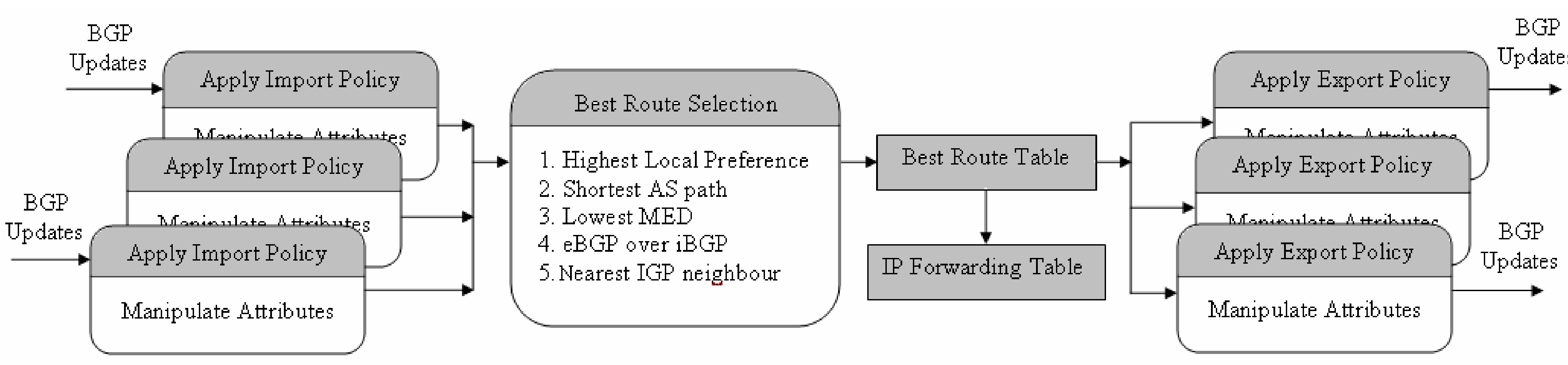
- RFC 1771, “A Border Gateway Protocol 4”, March 1995
- The *de facto* inter-domain routing protocol of the Internet
- Path vector protocol
- Incremental
- Relies on TCP

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

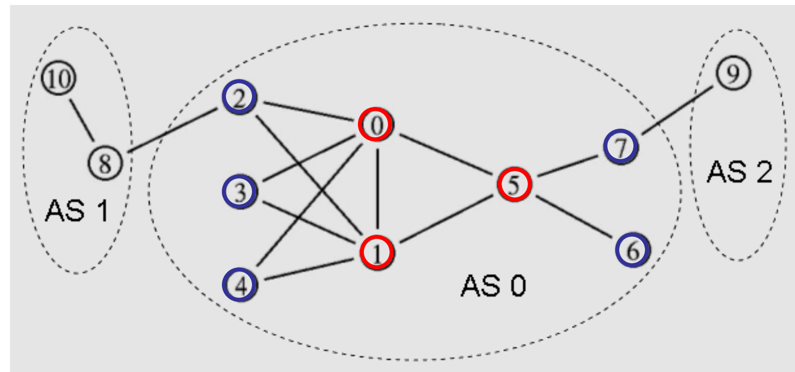
- Apply import policy
- Select a best route
- Install the best route
- Apply export policy and send out updates



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 ○ (red)
- client ○ (blue)

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

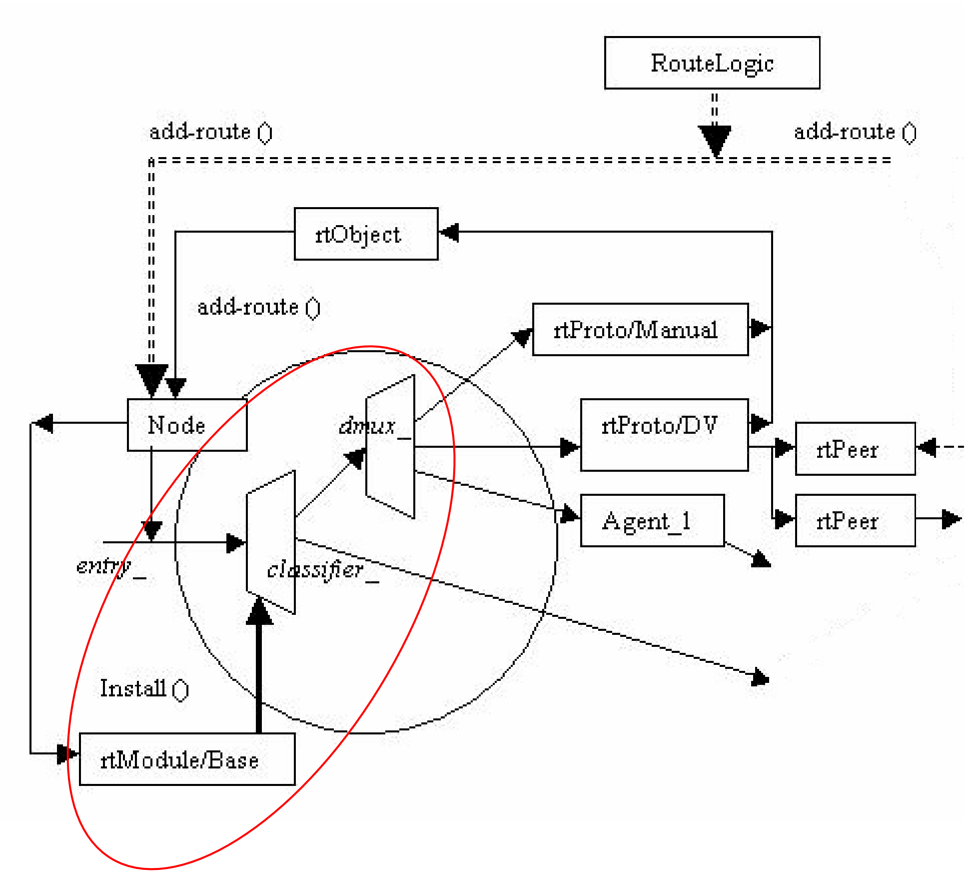
Road map

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

ns-2 unicast routing structure

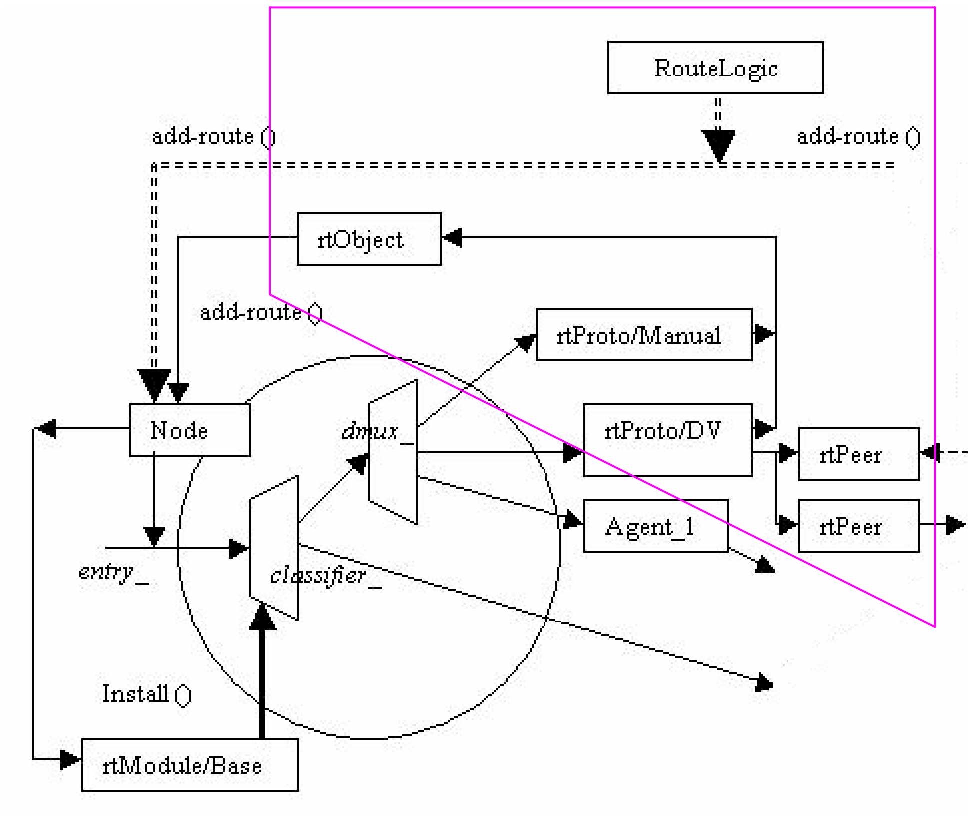
- Forwarding plane:
  - classify and forward packets
- Control plane:
  - routing info exchange, route computation, routing table creation and maintenance

Forwarding plane



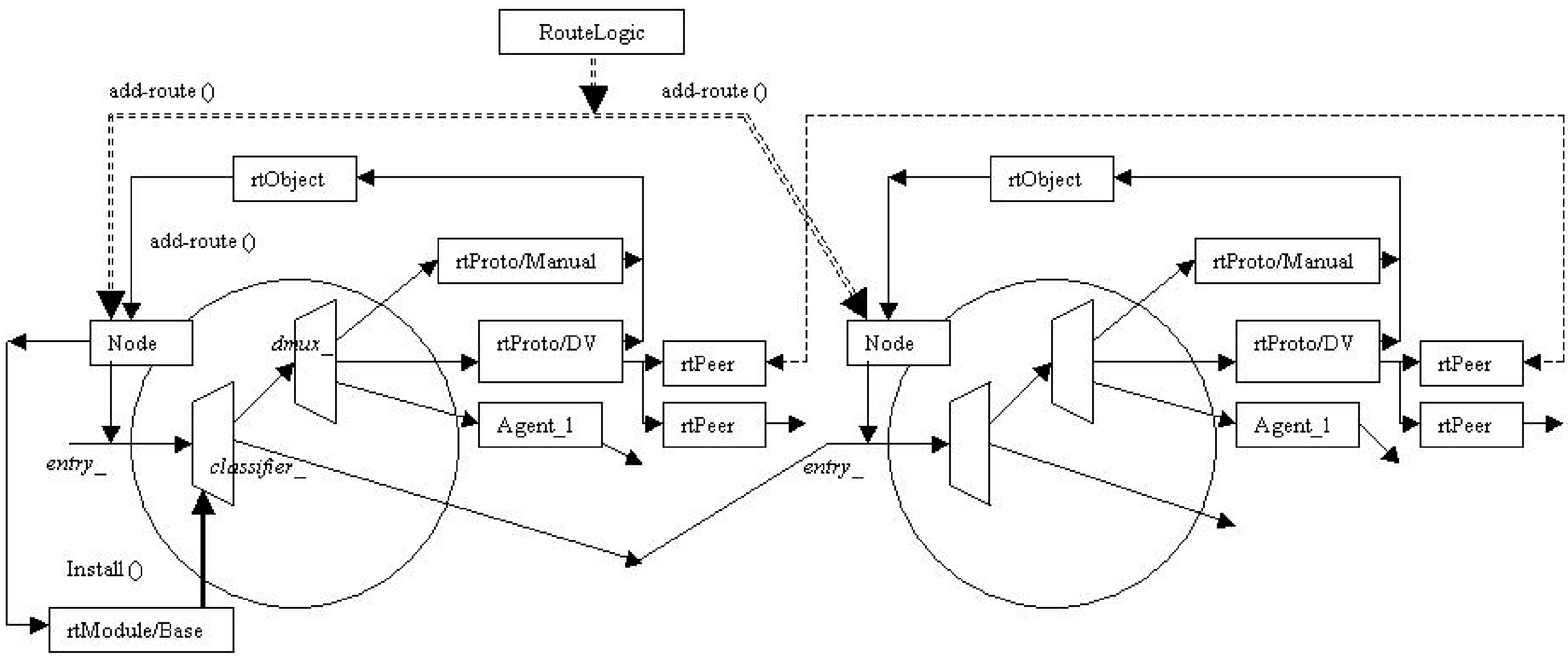
- Classifier (classifer_):
  - delivers the incoming packets either to the correct agent or to the outgoing link
- Routing Module (rtModule):
  - manages a node's classifier and provides an interface to the control plane

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

ns-2 routing structure diagram



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

No Socket layer in current ns-2

- BGP is built on top of TCP layer
- Without a socket layer, BGP has to monitor the status of the TCP three-way handshake and connection termination process
- Solution: we ported to ns-2 TcpSocket, the socket layer implementation of SSFNet

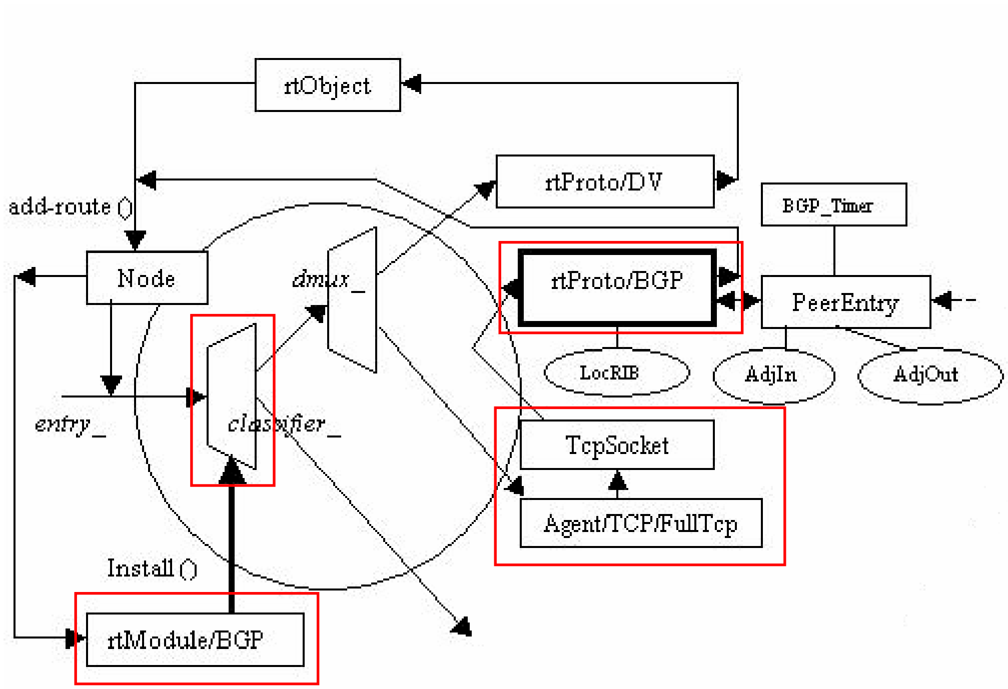
Simplified packet transmission

- Only packet headers (without data) are transmitted by the current TCP agent
- In order to exchange routing information, BGP need to transmit the whole packet
- Solution: we modified FullTcpAgent, the TCP agent for TcpSocket to support data transmission

No support for IPv4 addressing and packet forwarding schemes

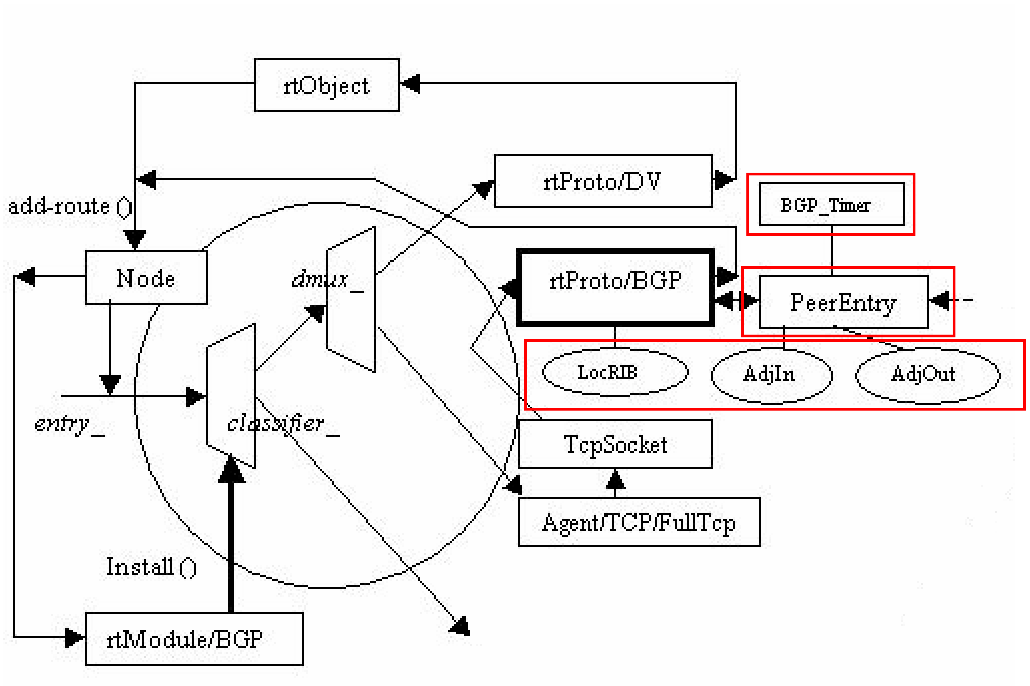
- BGP exchange routing information of IPv4 address blocks, called prefixes
- No support for IPv4 addressing and packet forwarding schemes in current ns-2.
- Solution: we created a new address classifier IPv4Classifier

ns-BGP unicast routing structure



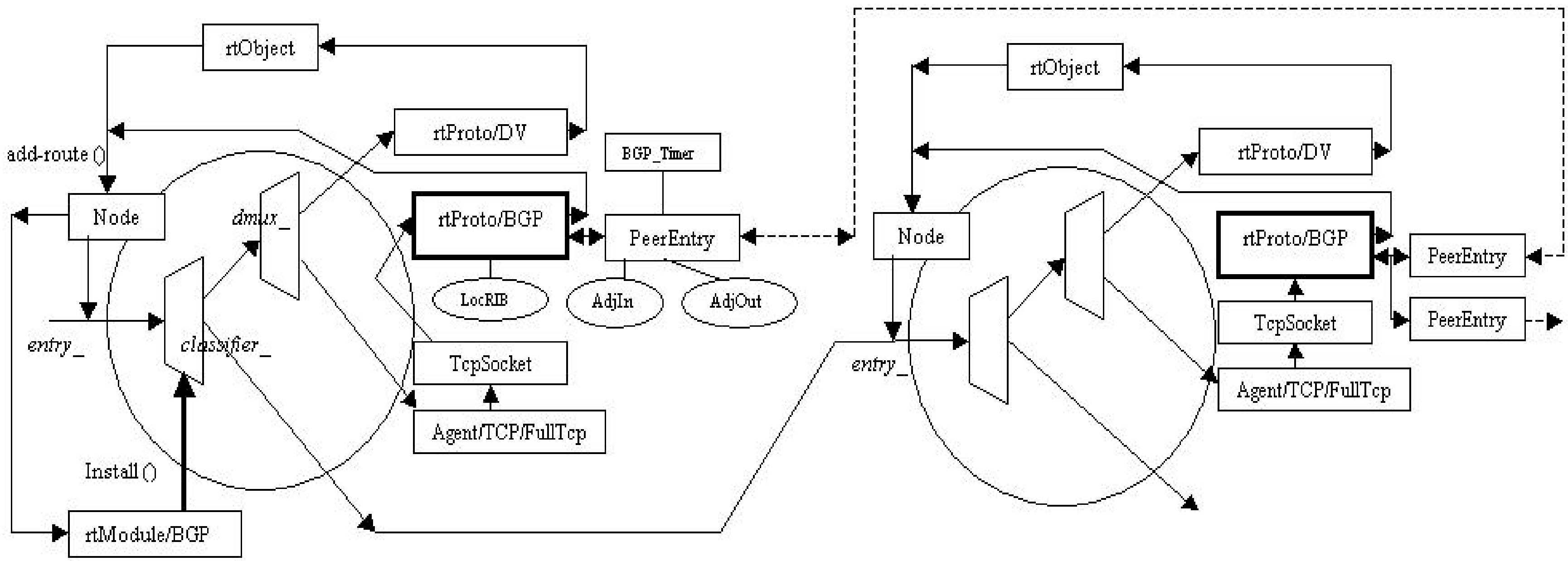
- IPv4Classifier (*classfier_*)
- BGP routing model (rtModule/BGP):
  - manages the IPv4Classifier
- TcpSocket:
  - encapsulating the TCP services into a socket interface
- BGP routing protocol (rtProto/BGP):
  - performs BGP operations

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)

ns-BGP unicast routing structure



Supported features

- Implemented all required features in RFC 1771
- Experimental features:
  - sender-side loop detection
  - withdrawal rate limiting
  - per-peer and per-destination rate limiting
- Optional features:
  - Multiple Exit Discriminator (MED)
  - aggregator
  - community
  - originator ID
  - cluster list

Road map

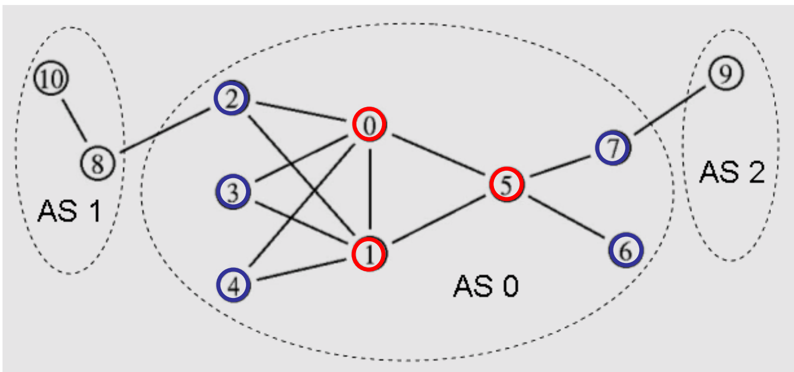
- Introduction
- Background
- Design and implementation of ns-BGP
- Validation test
- Scalability analysis
- Conclusions

Validation test

- Route reflection:
  - validates the behavior of multiple reflectors inside a BGP cluster

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

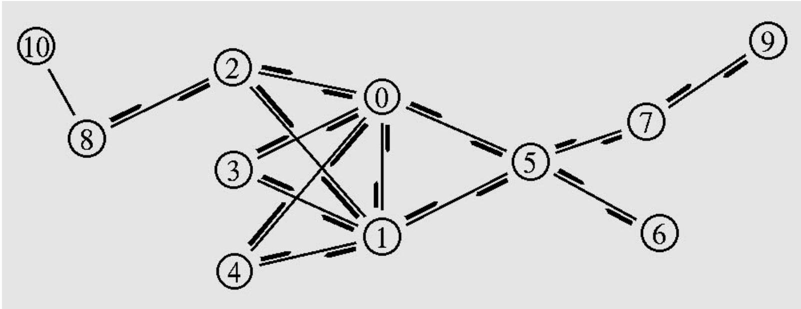
- BGP configuration:
  - AS 0 contains two clusters (0 and 1).
    - cluster 0 (nodes 0 – 4) contains 2 reflectors: nodes 0 and 1, with nodes 2, 3, and 4 as their clients
    - cluster 1 (nodes 5 -7) has one reflector (node 5), with nodes 6 and 7 as its clients
    - The three reflectors (nodes 0, 1,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

Traffic source and event scheduling

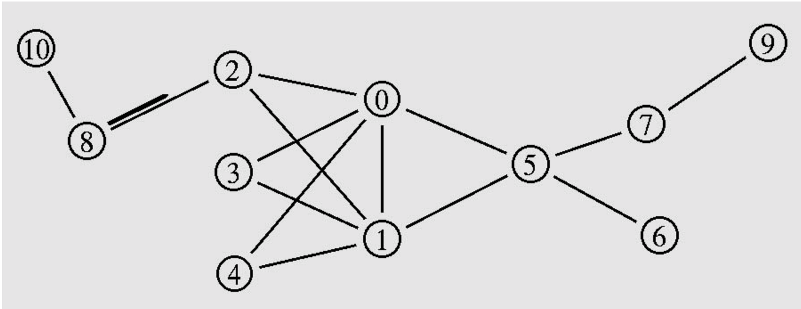
- Traffic source:
  - attached to node 4
  - constant bit rate (CBR)
  - transport protocol: UDP
  - sends segments of 20 bytes/ms to node 10 (10.1.10.1).
- Event scheduling:
  - traffic source begins sending at 0.23 s and stops at 20.0 s
  - 0.25 s: node 8 sends a route advertisement for network 10.1.10.0/24 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)

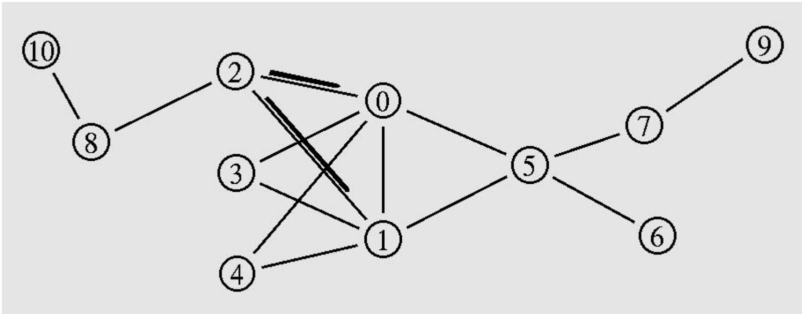
- 0.0503 s, TCP SYN segments are exchanged



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

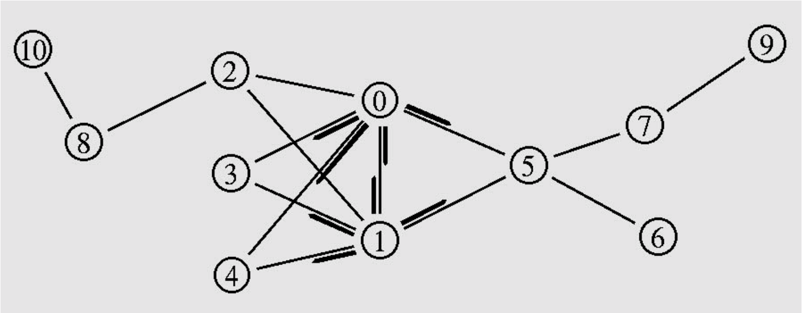


- 0.2525 s, node 2 propagates the route to nodes 0 and 1

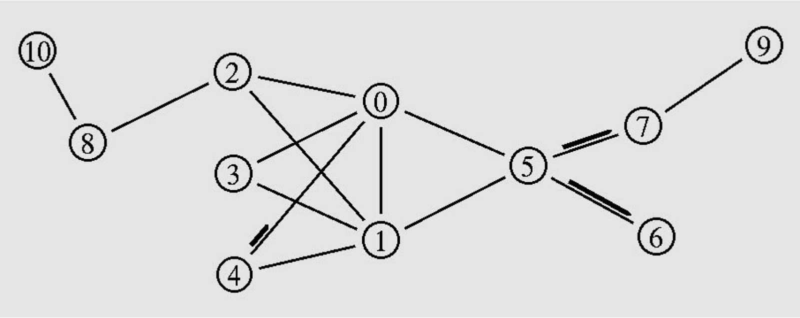


Simulation results: nam snapshots (2)

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

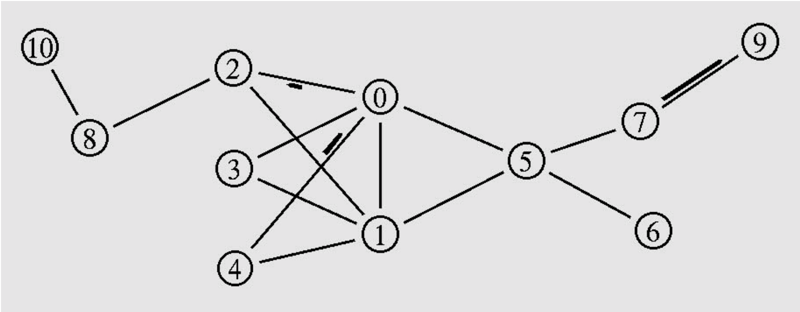


- 0.2568 s, node 5 reflects the route to nodes 6 and 7. Node 4 now knows the route to network 10.1.10.0/24, the UDP segment will be forwarded to node 10

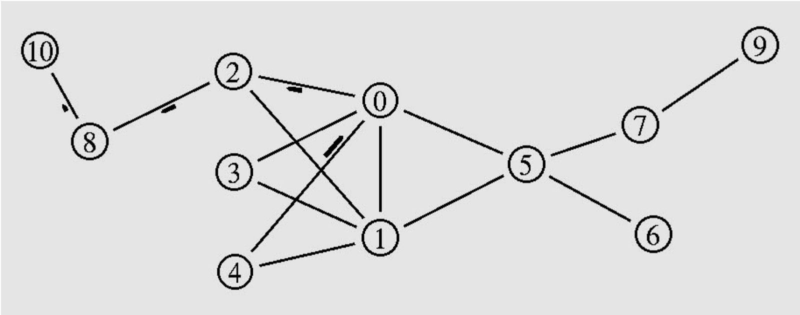


Simulation results: nam snapshots (3)

- 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

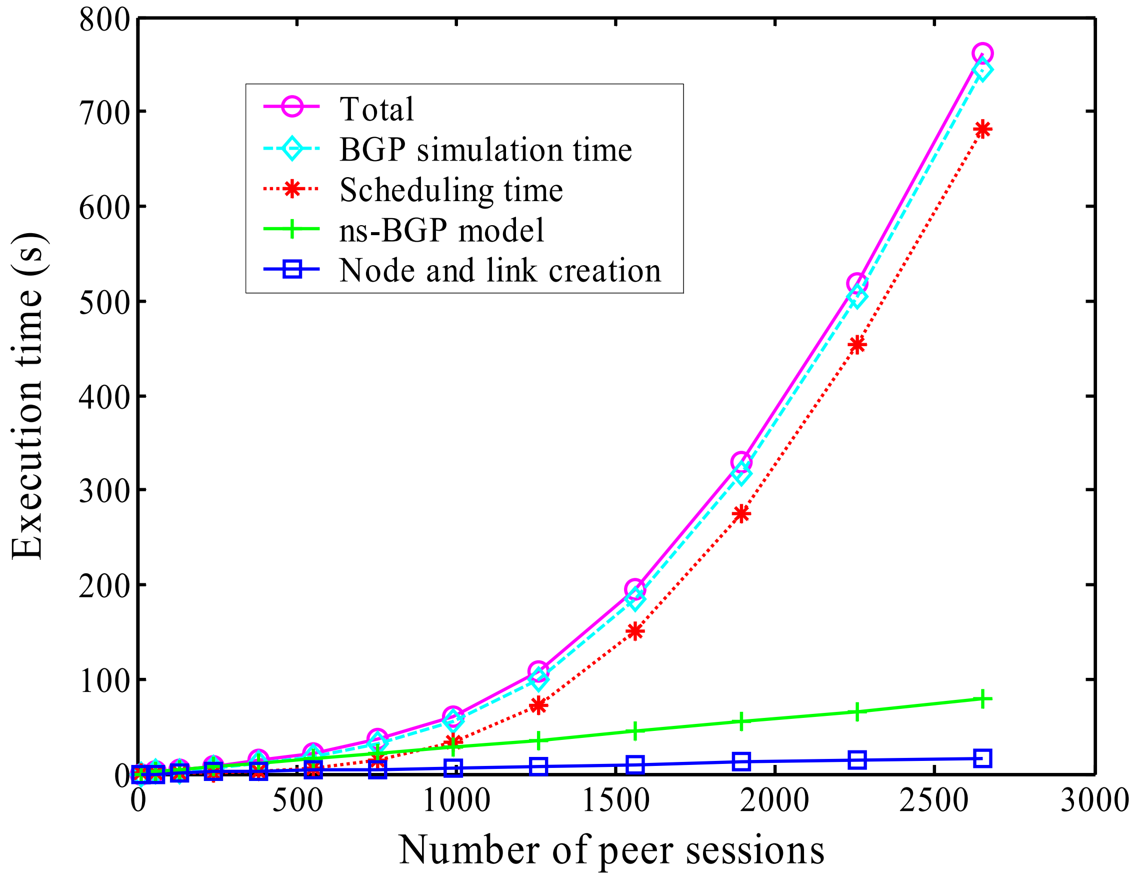
- Introduction
- Background
- Design and implementation of ns-BGP
- Validation test
- Scalability analysis
- Conclusions

Scalability analysis

- Scalability properties:
  - execution speed
  - memory requirements
- Scalability: number of peer sessions
- Scalability: size of routing tables
- Hardware platform:

  1.6 GHz Xeon host with 2 GBytes of memory

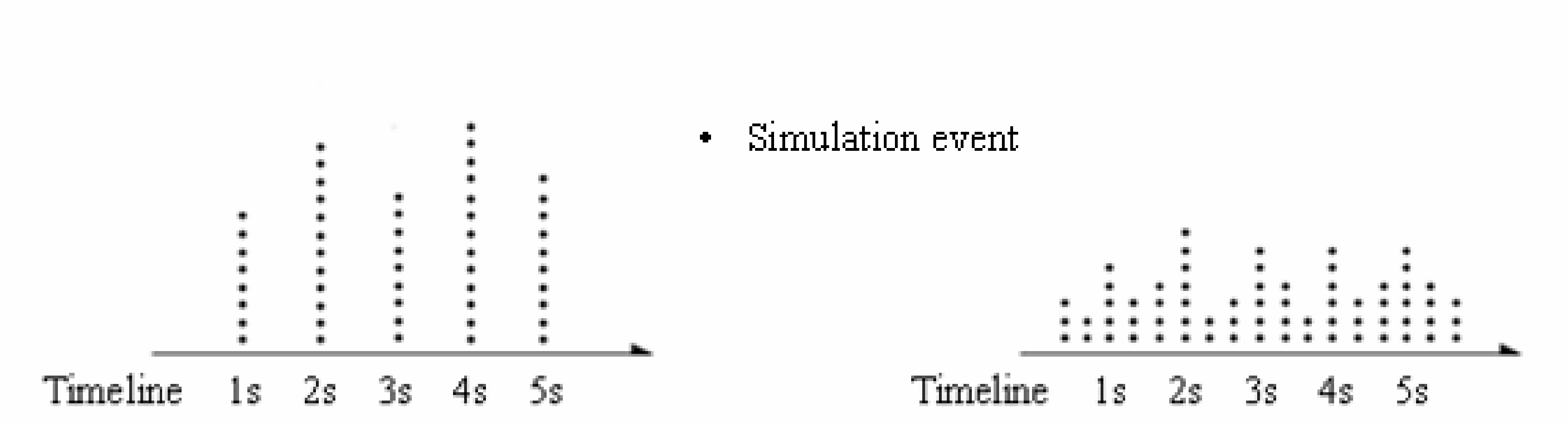
Scalability: number of peer sessions



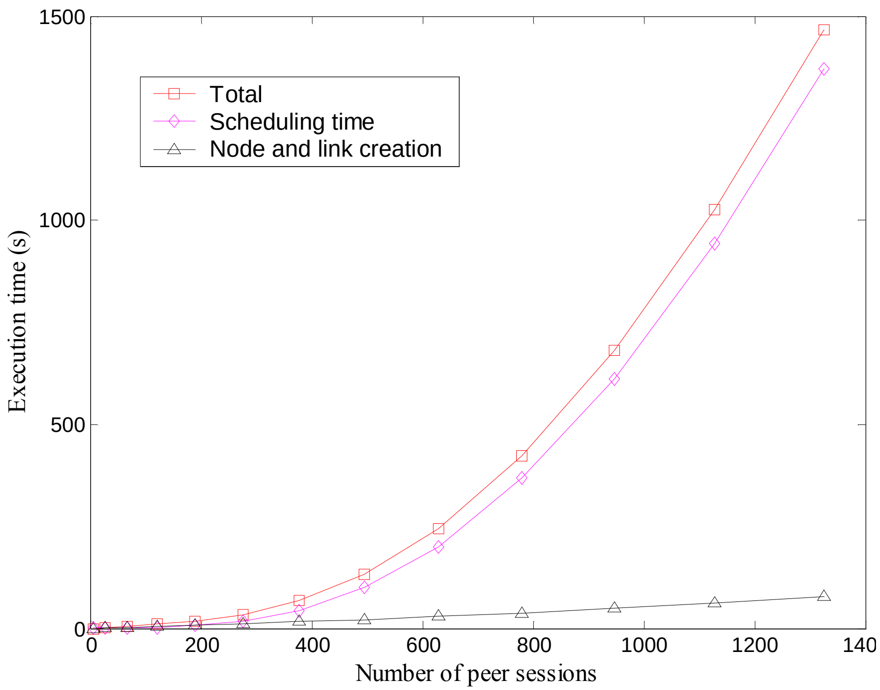
ns-2 calendar scheduler

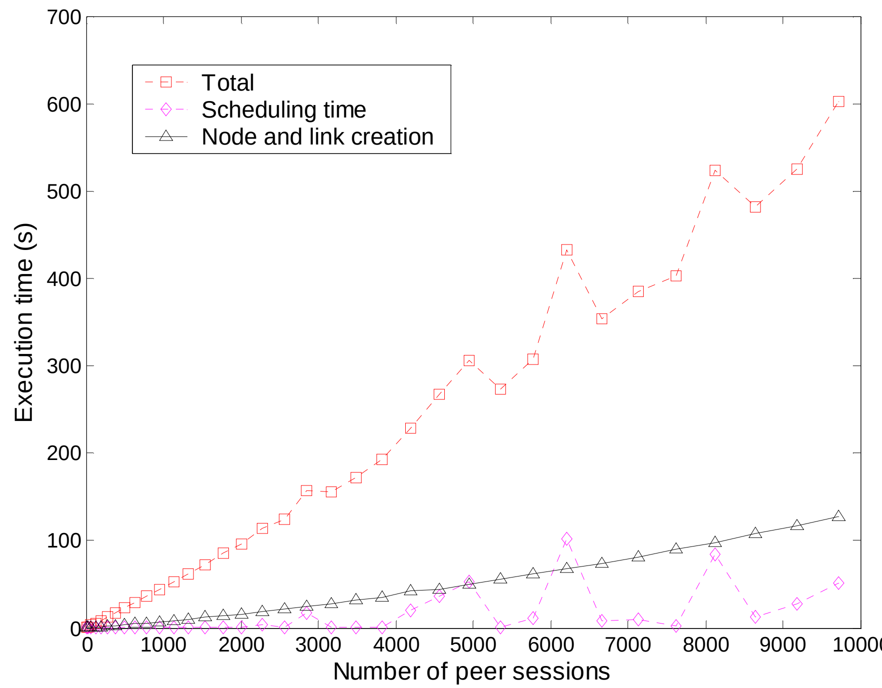
- Performance is affected by the distribution of the event times
- Large number of events scheduled at the same time instance can cause the scheduling time to increase exponentially
- Solution: we jittered BGP timers (start-up, keep-alive) to scatter simulation events
- While the jittered scheduling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

Scattering events



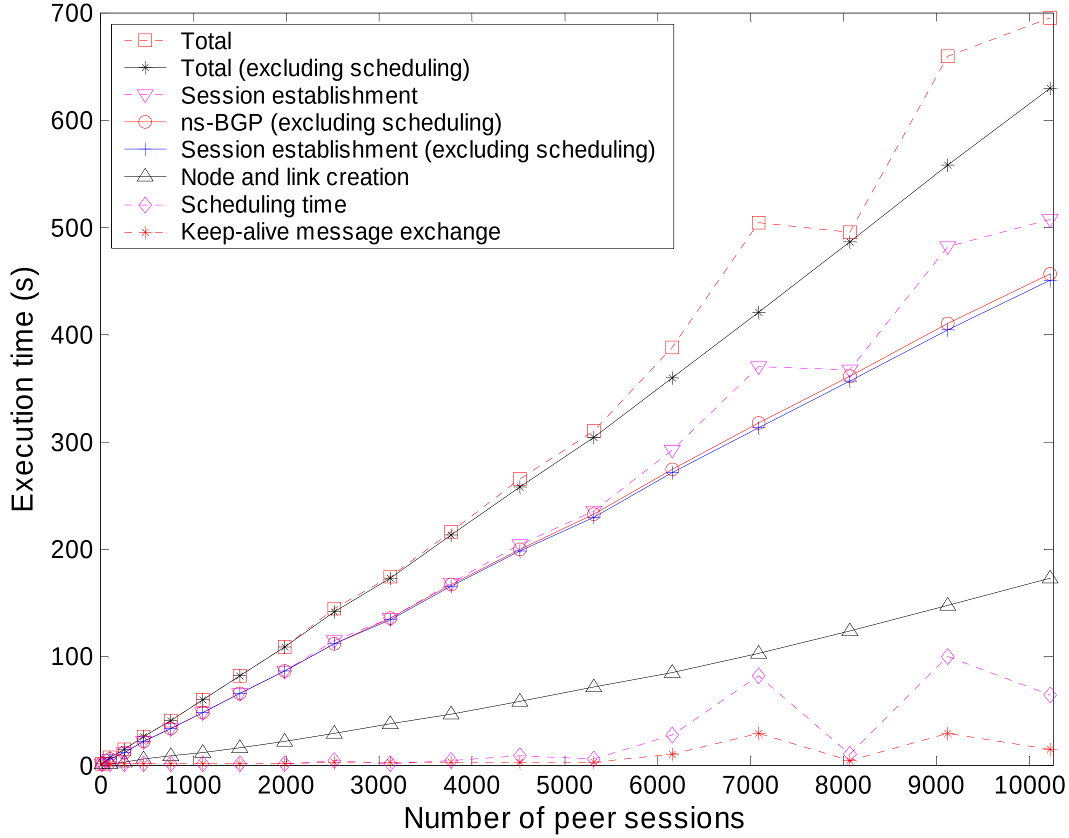
Scheduling times



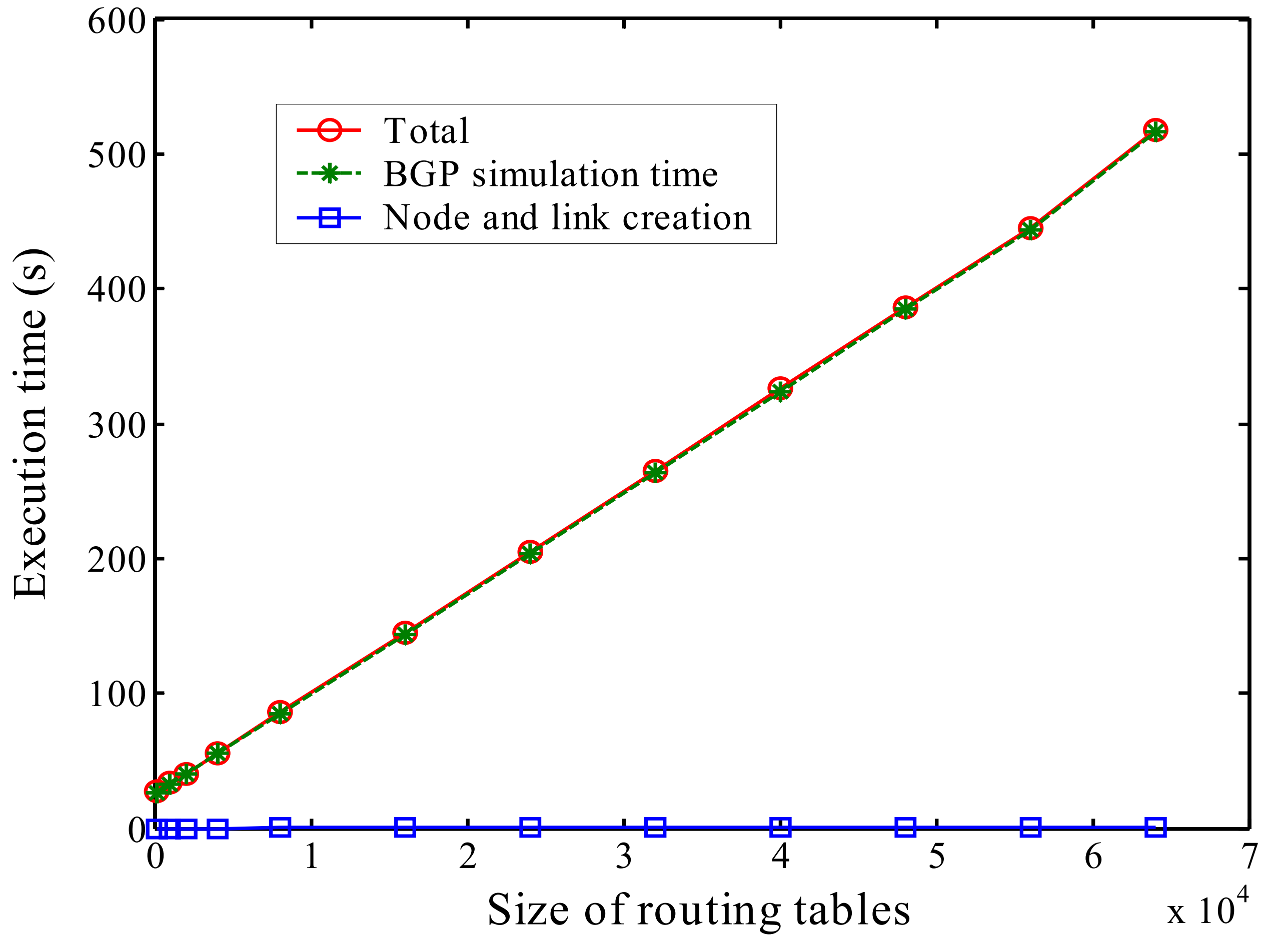


Execution time vs. number of peer sessions

- 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



Scalability:
Size of routing tables



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.

References

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

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

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

[4] 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.

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

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

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

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

BGP implementations

[9] 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.

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

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

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

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

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

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

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

Acknowledgements

- Zhang Wang for the implementation of TcpSocket
- Brian J. Premore for providing the SSF.OS.BGP4 implementation
- Fellow CNL members for their valuable comments