A Reinforcement Learning-Based Algorithm for Deflection Routing in Optical Burst-Switched Networks

Soroush Haeri[1], Wilson Wang-Kit Thong[2], Guanrong Chen[2], and Ljiljana Trajkovic[1]

[1] Simon Fraser University, Vancouver, British Columbia, Canada

[2] City University of Hong Kong, Hong Kong SAR, China

Roadmap

- Introduction
- Optical burst-switched (OBS) networks
- Deflection routing in OBS networks
- Reinforcement learning for deflection routing
- Node Degree Dependent algorithm with Q-learning: Q-NDD
- Simulation results
- Conclusion
- References

Introduction

- Buffer-less nodes do not possess first-in-first-out (FIFO) buffers to queue data
- Buffer-less network architectures:
  - optical burst-switched networks (OBS)
  - on-chip networks
- Contention is the main source of packet loss in buffer-less networks

Roadmap

- Introduction
- Optical burst-switched (OBS) networks
- Deflection routing in OBS networks
- Reinforcement learning for deflection routing
- Node Degree Dependent algorithm with Q-learning: Q-NDD
- Simulation results
- Conclusion
- References

Optical Burst-Switched Networks

- Designed to share optical fiber resources
- Current optical switching technologies:
  - reserve the entire light-path from a source to a destination
  - a light-path may not be shared unless its reservation is explicitly released
- OBS architecture:
  - enables statistical resource sharing of a light-path among multiple traffic flows
  - Switching is performed optically:
    - requires no optical/electrical/optical conversion in the data plane

Roadmap

- Introduction
- Optical burst-switched (OBS) networks
- Deflection routing in OBS networks
- Reinforcement learning for deflection routing
- Node Degree Dependent algorithm with Q-learning: Q-NDD
- Simulation results
- Conclusion
- References

Deflection Routing in OBS Networks

- A routing protocol selects the optimal path between a source and a destination
- In case of contention:
  - multiple arriving traffic flows at a node need to be routed through a single outgoing link
- In the absence of a contention resolution scheme:
  - a flow is routed through the optimal outgoing link defined by the routing table
  - other flows are discarded because the node has no buffer

Deflection Routing in OBS Networks

- Deflection routing may be employed as a contention resolution scheme
- A deflection routing algorithm temporarily misroutes packets instead of buffering or discarding them
- Deflection routing and the underlying routing protocol co-exist in a network

Deflection Routing in OBS Networks

- If neighboring nodes exchange traffic information:
  - nodes may generate deflection decisions based on a better understanding of their environment
- Reinforcement learning techniques may be used to process the exchanged traffic information
- Scalability:
  - complexity of the existing reinforcement learning-based deflection routing techniques depends on the network size

Deflection Routing in OBS Networks

- Tradeoff:
  - Quality of deflection routing vs. the number of signaling messages that the deflection routing algorithm requires
- The underlying routing protocol generates a significant number of control signals:
  - deflection routing protocols should generate fewer control signals
- Goal:
  - achieve better performance while transmitting fewer control signals

Roadmap

- Introduction
- Optical burst-switched (OBS) networks
- Deflection routing in OBS networks
- Reinforcement learning for deflection routing
- Node Degree Dependent algorithm with Q-learning: Q-NDD
- Simulation results
- Conclusion
- References

Reinforcement Learning for Deflection Routing

- Reinforcement learning algorithms are random in nature
- Randomness enables a deflection routing protocol to make viable decisions while transmitting fewer control signals
- Deflection decision (action):
  - select an alternate outgoing link that a node may use to deflect a traffic flow
- Reinforcement learning approach for deflection routing requires two modules:
  - signaling
  - learning and decision making

Reinforcement Learning for Deflection Routing

- Signaling module:
  - is aware of the state of the system
  - implements the algorithm for generating and delivering feedback signals
- Learning and decision making module:
  - In case of contention:
    - receives the state of the system from the signaling module
    - generates a deflection decision
  - After generating a deflection decision:
    - receives feedback from the signaling module
    - employs this feedback to enhance future decisions

Q-Learning

- The Q-learning algorithm has been used to design a learning and decision making module
- Q-learning:
  - is a simple reinforcement learning algorithm
  - maintains a Q-value Q(s,a) in a Q-table for every state-action pair

C. J. C. H. Watkins and P. Dayan, “Technical note, Q-learning,” *Machine Learning*, vol. 8, no. 3, pp. 279–292, May 1992.

Q-learning

- Q-learning update rule:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \times \left[r_{t+1} + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \right]$$

  - s : state of the system
  - a : action
  - α : learning rate
  - r : reward
  - γ : discount factor
  - subscripts t and $t+1$ denote two consecutive decision instances

Roadmap

- Introduction
- Optical burst-switched (OBS) networks
- Deflection routing in OBS networks
- Reinforcement learning for deflection routing
- Node Degree Dependent algorithm with Q-learning: Q-NDD
- Simulation results
- Conclusion
- References

Node Degree Dependent Algorithm with Q-Learning: Q-NDD

- Complexity of the proposed Node Degree Dependent (NDD) algorithm depends on a node degree
- Consider OBS network with *N* nodes:
  - each node maintains a Q-table
  - all nodes are NDD compatible
- NDD defines a state of the system by:
  - states of the optical interfaces
  - output port defined by the routing table
- NDD defines an action as an output port number
- On burst transmission, a node:
  - inspects the routing table for the next hop
  - checks the status of its optical interfaces

Node Degree Dependent Algorithm with Q-Learning: Q-NDD

- Four cases may occur:
  1. desired optical interface is available
  2. desired optical interface is busy and the burst has not been deflected earlier by any other node
  3. desired optical interface is busy and the burst has been deflected earlier by another node
  4. all optical interfaces are busy and the burst has been deflected earlier by another node

Case 1 and Case 2

- Case 1: Desired optical interface is available:
  - optical cross-connects are configured according to the path defined by the routing table
- Case 2: Desired optical interface is busy and the burst has not been deflected earlier by another node:
  - node passes the state of the system to the Q-learning module
  - Q-learning module then:
    - passes to the NDD module the output port number (action) associated with the maximum Q-value
    - waits for feedback
    - makes no new decisions during the idle interval

Case 2 (cont.)

- NDD module:
  - adds to the burst header:
    - a unique ID number
    - the address of the node that initiated the deflection
    - a deflection hop counter DHC, which increments each time other nodes deflect the burst
  - records the current time as the deflection time (DfT) with the ID that has been added to the burst
  - initiates the drop notification (DN) timer

Case 2 (cont.)

- records the action selected by the Q-learning module
  - records are used if the node needs to deflect a burst:
    - that has been deflected earlier
    - during an idle interval
- waits for a feedback signal for DN_{max} seconds
  - if no feedback is received:
    - assumes that the burst has been successfully delivered
    - returns the maximum reward value to the Q-learning module for an update

Case 2 (cont.)

- if feedback is received:
  - calculates a reward value based on the feedback signal
  - returns the reward value to the Q-learning module for an update

Case 3

- Case 3: Desired optical interface is busy and the burst has been deflected earlier by another node:
  - NDD module:
    - checks the deflection hop counter (DHC) field in the burst header and if:
      - $DHC == DHC_{max}$: prepares a feedback signal and discards the burst
      - $DHC < DHC_{max}$: checks the state of the system and performs the latest action that the Q-learning module has generated for the current state

Case 4

- Case 4: All optical interfaces are busy and the burst has been deflected earlier by another node:
  - NDD module prepares a feedback signal and discards the burst

Feedback Signal

- Feedback signal is composed of:
  - burst ID number that was assigned to the burst by the node that initiated the deflection
  - DHC field value:
    - DHC is not equal to DHC_{max} when the burst is discarded because the node if fully congested
  - drop time DrT:
    - time instant when the burst was discarded

Reward

- Reward is generated based on the feedback signal received at the node that has initiated the deflection
  - NDD module:
    - calculates the total travel time:

      $$TTT = DrT - DfT$$

    - uses a decreasing function with the global maximum at (0,0) to map TTT and DHC to a real valued reward
    - example:

$$maxR, \alpha, \beta > 0$$

$$\begin{cases} maxR & DHC = 0 \text{ and } TTT = 0 \\ -\alpha \times DHC - \beta \times TTT & \text{otherwise} \end{cases}$$

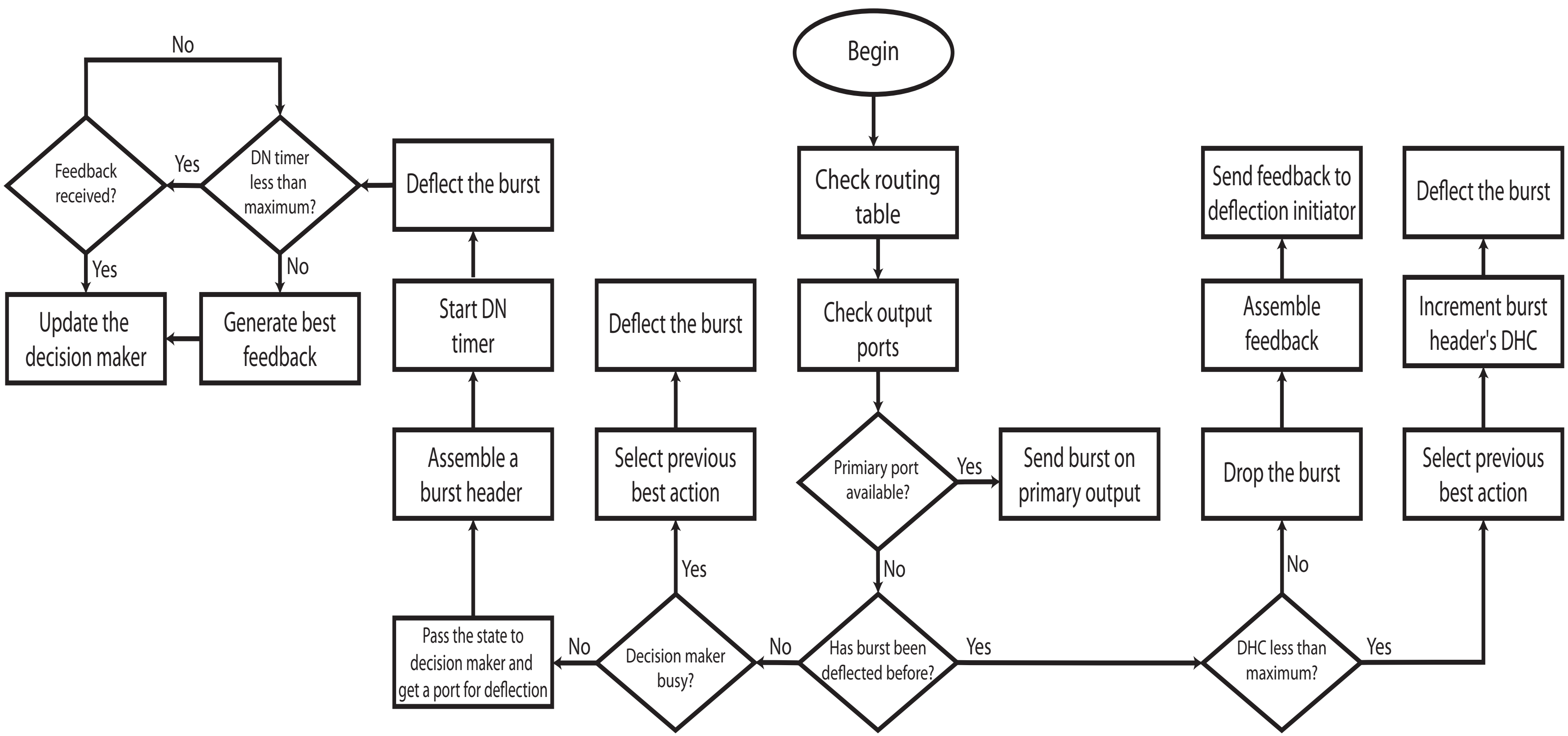
Q-Learning Module Update

- Q-learning module updates the Q-value of the current state and the selected action as:

$$Q(s,a) \leftarrow Q(s,a) + \alpha\big(r - Q(s,a)\big)$$

  - s : state of the system
  - a : action
  - α : learning rate
  - r : reward

NDD: Flow Chart



Roadmap

- Introduction
- Optical burst-switched (OBS) networks
- Deflection routing in OBS networks
- Reinforcement learning for deflection routing
- Node Degree Dependent algorithm with Q-learning: Q-NDD
- Simulation results
- Conclusion
- References

Simulation Results: Scenarios

- Q-NDD is compared with the existing Reinforcement Learning Deflection Routing Scheme (RLDRS):
  - National Science Foundation (NSF) network topology (64 wavelengths):
    - burst loss probability
    - average end-to-end delay
    - average number of hops
    - average number of deflections

A. Belbekkouche, A. Hafid, and M. Gendreau, "Novel reinforcement learning-based approaches to reduce loss probability in buffer-less OBS networks," *Comput. Netw.*, vol. 53, no. 12, pp. 2091–2105, Aug. 2009.

Simulation Results: Scenarios

- Randomly generated Waxman network topologies:
  - memory usage
  - CPU usage
  - simulation time

Simulation Results: Network Architecture

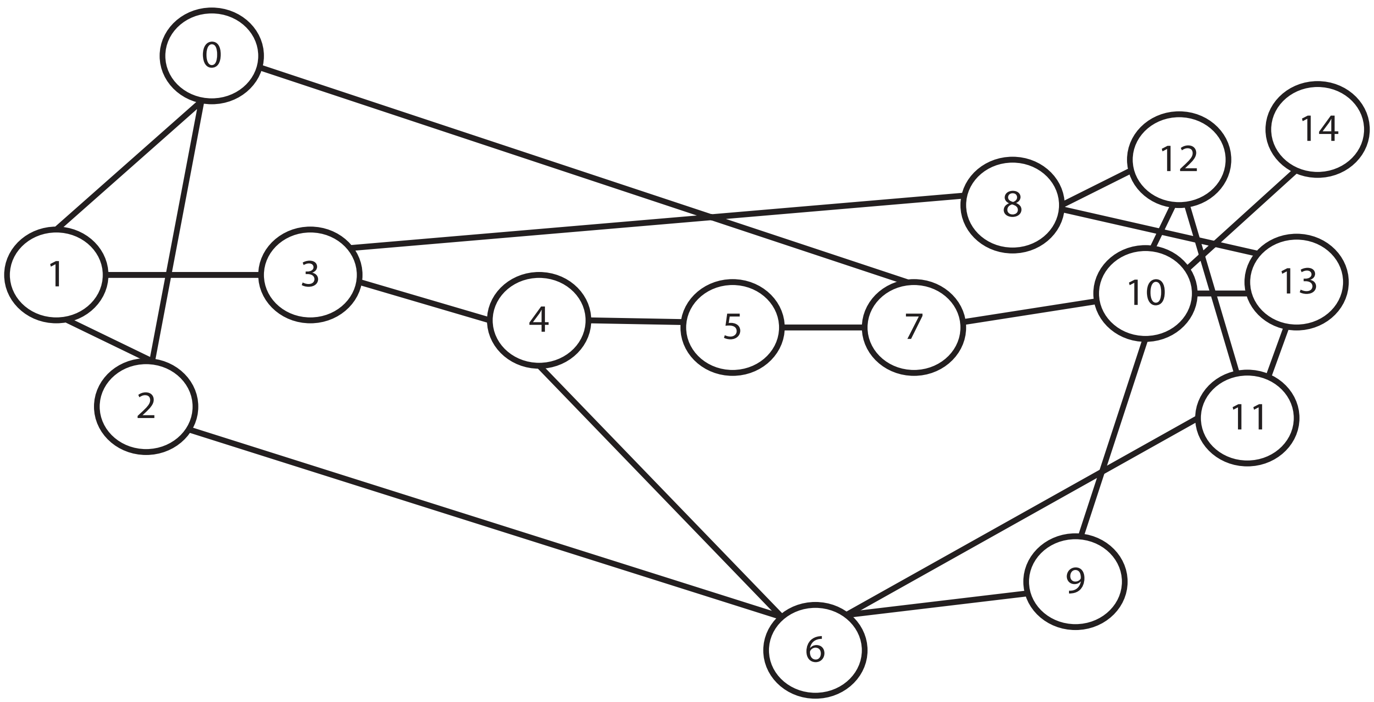
- Buffer-less optical burst switching architecture:
  - 1 Gbps fiber links
  - 64 wavelengths
- Traffic flows:
  - Poisson arrivals
  - 0.5 Gbps data rate
  - 50 bursts:
    - each burst carries 12.5 kB payload

Simulation Results: Parameters

- Learning rate: $\alpha = 0.1$
- Maximum deflection hop counter: DHC_{max}= 2
- Maximum deflection notification: DN_{max}= 50 ms

Simulation Results: NSF Topology

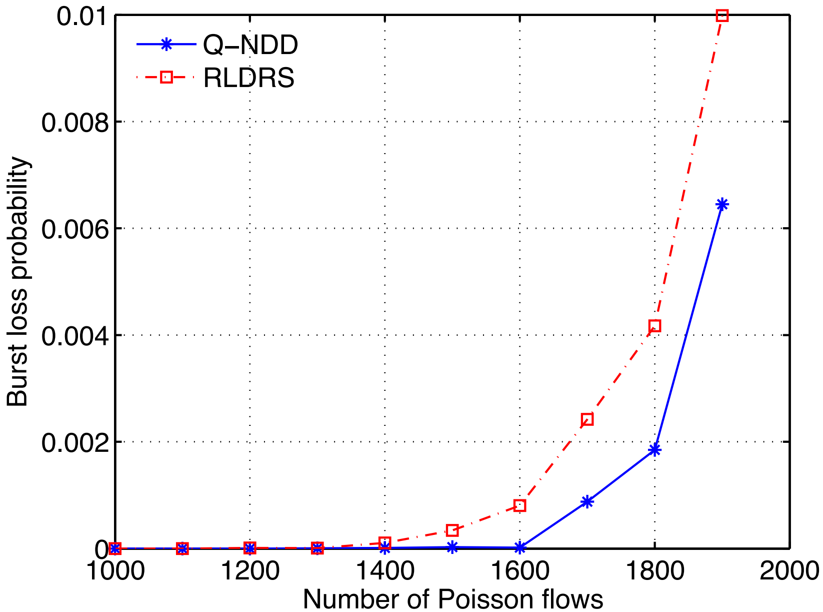
- National Science Foundation (NSF) network topology:



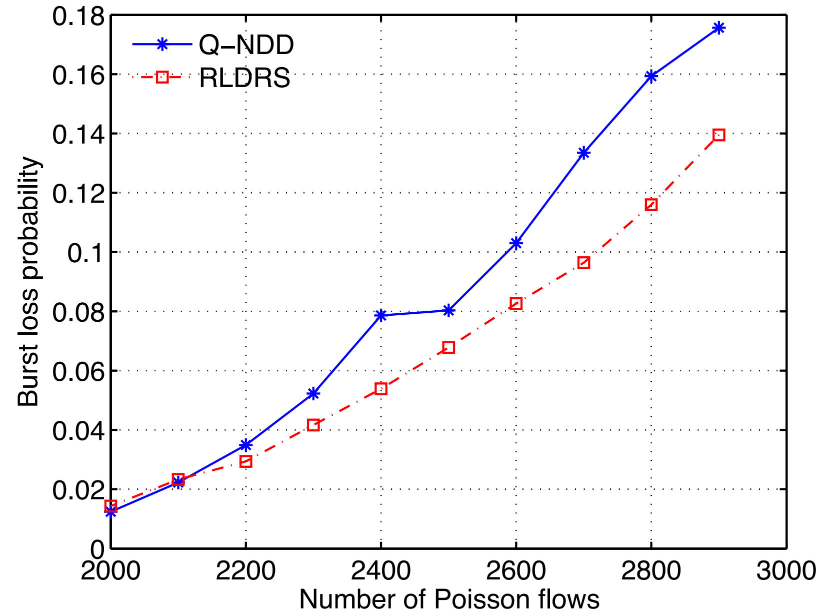
NSF network after the 1989 transition

Q-NDD and RLDRS: Comparison

- Burst loss probability: NSF topology with Poisson flows



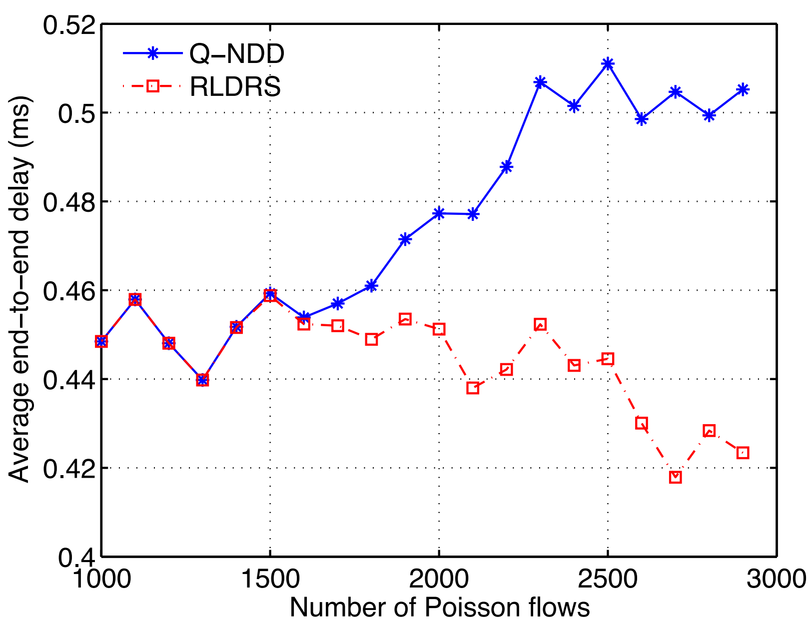
Burst loss probability: low to moderate loads



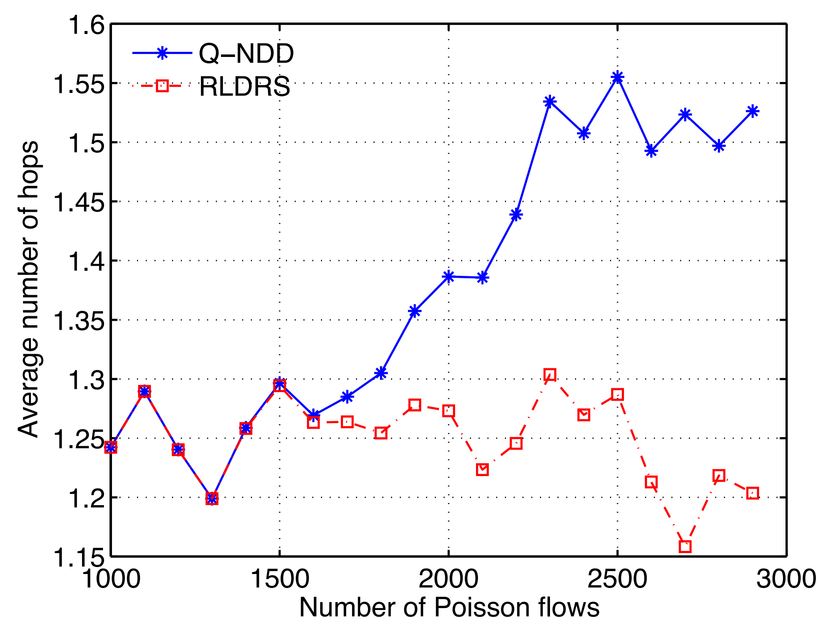
Burst loss probability: moderate to high loads

Q-NDD and RLDRS: Comparison

- End-to-end delay and average number of hops: NSF topology with Poisson flows



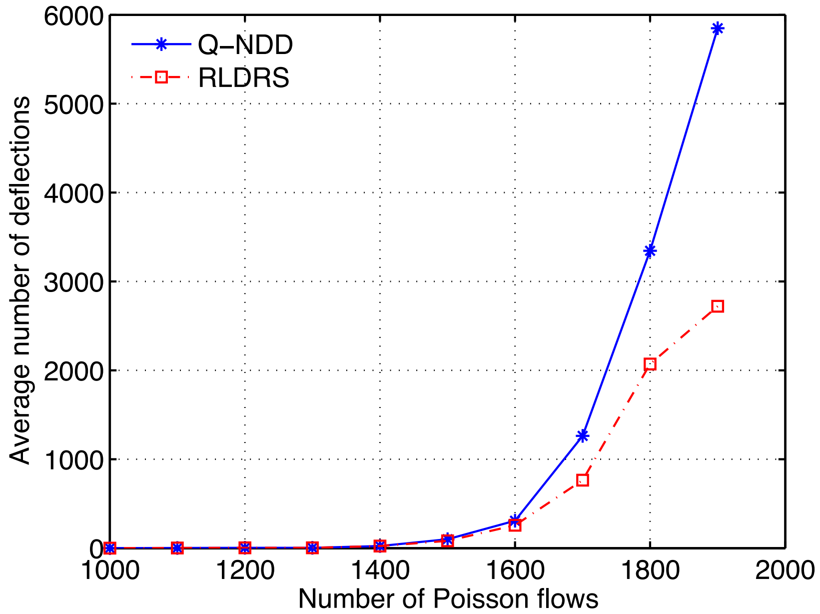
End-to-end delay



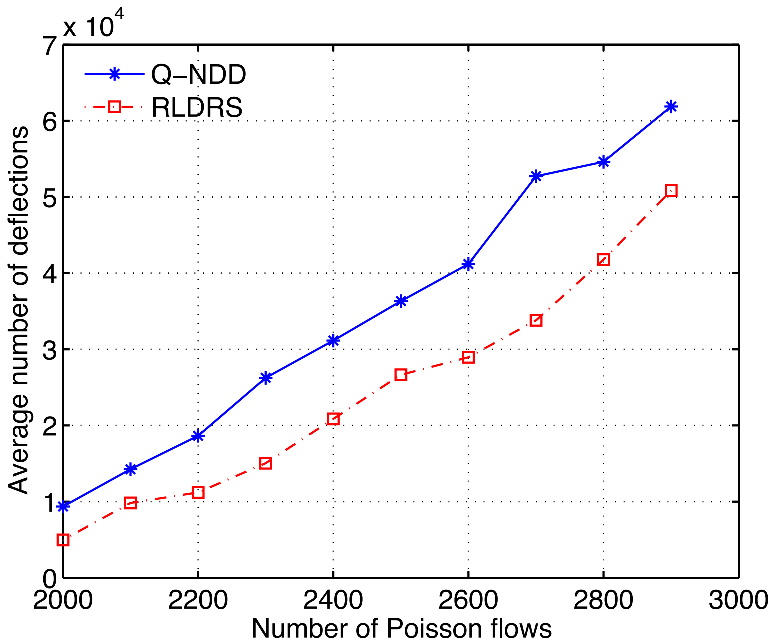
Average number of hops

Q-NDD and RLDRS: Comparison

- Average number of deflections: NSF topology with Poisson flows



Average number of deflections:
low to moderate loads



Average number of deflections:
moderate to high loads

Simulation Results: Waxman Topologies

- Number of nodes:
  - 500, 1,000, and 2,000
- Single wavelength
- Probability of a link connecting nodes u and v in a Waxman graph:

$$\Pr\left(\{u, v\}\right) = \eta \exp\left(\frac{-d(u, v)}{L\delta}\right)$$

- Parameters:
  - $\eta = 0.2$
  - $\delta = 0.15$

E. W. Zegura, K. L. Calvert, and M. J. Donahoo "A quantitative comparison of graph-based models for Internet topology," *IEEE/ACM Trans. Netw.*, vol. 5, no. 6, pp. 770–783, Dec. 1997.

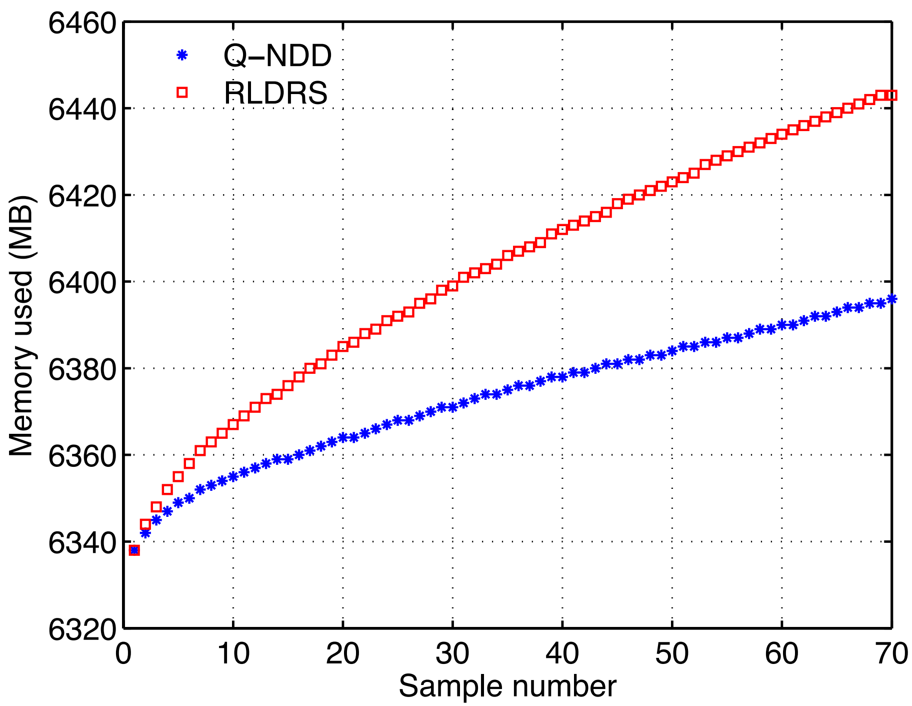
Q-NDD and RLDRS: Comparison

- Memory and CPU usage: Waxman topologies with Poisson flows

Algorithm	Number of nodes	Number of links	Number of flows	Min. memory usage (MB)	Max. memory usage (MB)	Total CPU time used (mm:ss)	Total simulation time (s)
Q-NDD	500	1,500	3,000	561	578	1:25.61	1,832.8
	1,000	3,000	6,000	1,723	1,754	6:03.67	4,042.5
	2,000	6,000	12,000	6,338	6,397	32:58.17	9,661.5
RLDRS	500	1,500	3,000	561	587	2:05.85	2,830.7
	1,000	3,000	6,000	1,723	1,775	8:02.16	6,026.0
	2,000	6,000	12,000	6,338	6,444	41:29.18	14,363.1

Q-NDD and RLDRS: Comparison

- Memory usage: Waxman topology with Poisson flows
  - 2,000 nodes (70 sample points)



Roadmap

- Introduction
- Optical burst-switched (OBS) networks
- Deflection routing in OBS networks
- Reinforcement learning for deflection routing
- Node Degree Dependent algorithm with Q-learning: Q-NDD
- Simulation results
- Conclusion
- References

Conclusion

- The proposed Q-NDD algorithm employs Q-learning to generate optimal deflection decisions:
  - may be employed in buffer-less networks to reduce burst loss due to contention
  - is scalable and its complexity depends on the degree of a node
- When compared to RLDRS, Q-NDD:
  - requires less memory and processing time
  - reduces the burst loss probability in case of low to moderate traffic loads
  - deflects bursts more frequently
  - requires additional hops and causes larger end-to-end delay

Roadmap

- Introduction
- Optical burst-switched (OBS) networks
- Deflection routing in OBS networks
- Reinforcement learning for deflection routing
- Node Degree Dependent algorithm with Q-learning: Q-NDD
- Simulation results
- Conclusion
- References

References

- L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: a survey," *J. of Artificial Intell. Research*, vol. 4, pp. 237–285, 1996.
- R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*. Cambridge, MA, USA: MIT Press, 1998.
- C. J. C. H. Watkins and P. Dayan, "Technical note, Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, May 1992.
- J. A. Boyan and M. L. Littman, "Packet routing in dynamically changing networks: a reinforcement learning approach," in *Advances in Neural Inform. Process. Syst.*, vol. 6, pp. 671–678, 1994.
- S. P. M. Choi and D.-Y. Yeung, "Predictive Q-routing: a memory-based reinforcement learning approach to adaptive traffic control," in *Advances in Neural Inform. Process. Syst.*, vol. 8, pp. 945–951, 1998.

References

- L. Peshkin and V. Savova, "Reinforcement learning for adaptive routing," in *Proc. Int. Joint Conf. Neural Netw.*, Honolulu, HI, USA, May 2002, vol. 2, pp. 1825–1830.
- A. Nowe, K. Steenhaut, M. Fakir, and K. Verbeeck, "Q-learning for adaptive load based routing," in *Proc. IEEE Int. Conf. Syst., Man, and Cybern.*, San Diego, CA, USA, Oct. 1998, vol. 4, pp. 3965–3970.
- Y. Kiran, T. Venkatesh, and C. Murthy, "A reinforcement learning framework for path selection and wavelength selection in optical burst switched networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 9, pp. 18–26, Dec. 2007.
- A. Belbekkouche, A. Hafid, and M. Gendreau, "Novel reinforcement learning-based approaches to reduce loss probability in buffer-less OBS networks," *Comput. Netw.*, vol. 53, no. 12, pp. 2091–2105, Aug. 2009.

References

- C. Qiao and M. Yoo, “Optical burst switching (OBS)—a new paradigm for an optical Internet,” *J. of High Speed Netw.*, vol. 8, no. 1, pp. 69–84, Mar. 1999.
- A. Zalesky, H. Vu, Z. Rosberg, E. W. M. Wong, and M. Zukerman, “Modelling and performance evaluation of optical burst switched networks with deflection routing and wavelength reservation,” in *Proc. INFOCOM*, Hong Kong SAR, China, Mar. 2004, vol. 3, pp. 1864–1871.
- X. Yu, J. Li, X. Cao, Y. Chen, and C. Qiao, “Traffic statistics and performance evaluation in optical burst switched networks,” *J. Lightw. Technol.*, vol. 22, no. 12, pp. 2722–2738, Dec. 2004.
- B. M. Waxman, “Routing of multipoint connections,” *IEEE J. Sel. Areas Commun.*, vol. 6, no. 9, pp. 1617–1622, Dec. 1988.
- (2012, Sept.) BRITE topology generator. [Online]. Available: http://www.cs.bu.edu/brite.