Comparison Between Pre-computation and On-Demand Computation QoS Routing with Different Link State Update Algorithms

Haifeng Zhu

**Supervisor**  Professor Raimo Kantola
**Instructor**  Ph.D. Peng Zhang

Networking Laboratory, HUT
Contents

• Background
• Simulation Environment
• Simulation Results and Analysis
• Conclusions
• Future work
Background

• QoS Routing

• Path Computation Algorithm

• Inaccurate Routing Information
QoS Routing

• Routing for Best-Effort traffic
  OSPF and RIP use the shortest path to forward packets without considering the delay or bandwidth of the path

• QoS routing considers the quality constrains
  ➢ Delay
  ➢ Bandwidth
  ➢ Jitter

• Link State update
  ➢ Period-Based
  ➢ Threshold-Based
    • Equal Class Based
    • Unequal Class Based
Path Computation Algorithm

• Pre-Computation Algorithm
  Paths from each node to all of the destinations are computed periodically or after a number of link state updates.

• On-Demand Computation Algorithm
  The path to a specified destination is computed every time a request is initiated.

• Path caching architecture
  An extension to On-demand computation tends to reduce the processing costs.
Pre-Computation Algorithm

✓ Scalability
   The number of path computations is independent from the number of requests

✓ Fault tolerance
   Alternative routes for bypassing the failure parts in the network can be computed in advance

✓ Load Balancing
   Traffic can be balanced by directing different requests to several alternative routes properly

• Storage
   Storage for QoS routing table is needed

• Processing load
   For ad hoc network or network with fewer amounts of requests, most of pre-computed paths may never be used. Thus lots of processing capability is wasted
On-Demand Computation Algorithm

✓ Simpler implementation
  - Only one path to the specified destination is computed when new request is initiated
  - Routing table is not necessary

✓ Storage
  Not necessary to maintain routing table, thus storage for routing table is saved

• Scalability
  Not scale well for large network that have many requests

• Processing load
  Networks with many requests will generate large computation overhead
Path caching architecture

• Path caching for on-demand computation
  ➢ Cache for storing the paths that are computed on demands of the previous requests
  ➢ On-demand computation is triggered only if the route needed cannot be found from the path cache

• Benefits
  Processing cost can be reduced due to less path computations

• Storage
  The storage of cache is comparable to or even more than that of the pre-computation algorithm

• Granularity of cache
  Network with proper selected hybrid granularity scheme for cache performs well with proper amount of storage
Inaccurate Routing Information

• Source of inaccuracy
  - Link states are not updated on time
    Broadcasting the LSAs for all changes in the network is infeasible due to the huge overhead to the network
  - LSAs may be lost
  - Temporal conditions like congestion in the network
  - Information aggregation in large network

• Problem
  Path selection base on inaccurate routing information may be non-optimal or incorrect
  - Non-optimal path selection decreases the utilization of the network
  - Incorrect path selection leads to more blocking
Simulation Environment

• QoS Routing Simulator
  - QRS was developed at Helsinki University of Technology
  - QRS models the network as a combination of different kinds of components
  - A certain number is allocated for every main QOSPF action to simulate the practical cost in real implementation
  - Networks with different topologies are modeled with configuration files

• New routing algorithms and components
  - Pre-computation algorithm
  - On-demand computation algorithm
  - Traffic generator
    For generating traffic with bandwidth requirement
New routing algorithms (1/2)

Flowchart for computation of Bellman-Ford routing table

When a request is initiated, entries in the BF routing table are checked
New routing algorithms (2/2)

Flowchart of On-demand computation algorithm
Results and Analysis (1/3)

• Topologies

Matrix 2*2

Matrix 3*3

Matrix 4*4
Period-Based Link State Update Method with different periods (Pre-Computation)

- For same request rate, the blocking rate grows with the period of link state update.
- For link state period are 200ms and 100ms, the blocking rates are not much different between them, but cost of period=100ms is about twice as much as that of period=200ms
Results and Analysis (3/3)

Period-Based Link State Update Method with different periods (On-Demand Computation)

The blocking rates of on-demand computation algorithm are lower but the costs are higher compare to that of pre-computation algorithm
Conclusion

• In smaller networks, the on-demand computation has no obvious advantage in performance but has higher cost.
• The frequency of the link state update can affect the network performances and the costs significantly.
• The update triggering policy should be chosen carefully.
• Networks size is an important factor for performances and the costs.
Future Work

- More simulation works with larger and more complicated networks
  To model the practical network better

- Extension to the QRS simulator
  - More traffic models
  - More routing algorithms

- Study other factors that might affect the usage of the routing algorithms
Thank You!

Questions?