Hierarchical dynamic traffic engineering considering the upper bounds of link utilizations

Yuichi Ohsita, Takashi Miyamura, Shin'ichi Arakawa, Kohei Shiomoto, and Masayuki Murata

Background
Traffic Engineering

- Increase in Traffic Amount
- Significant changes of traffic
  - Traffic caused by peer-to-peer

⬇

- Network should efficiently accommodate traffic
- Network should handle the sudden traffic changes

⬇

- Dynamic Traffic Engineering
  - By dynamically changing the routes, we can accommodate the traffic efficiently even in the case of significant traffic changes

Background
Problem in traffic engineering in large-scale networks

- A large amount of traffic information is required to perform traffic engineering
  - The size of the traffic matrix is square of the number of nodes.
- Calculating the optimal routes takes long time.

⬇

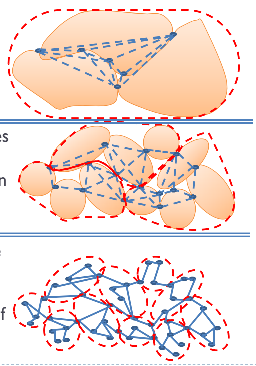
- It takes long time to mitigate the congestion

Goal of our work

- Goal
  - Traffic engineering method that can mitigate the congestion immediately after the traffic changes even when significant change occurs.
- Our Approach
  - Change the routes according to link loads
  - Divide the network hierarchically into multiple ranges

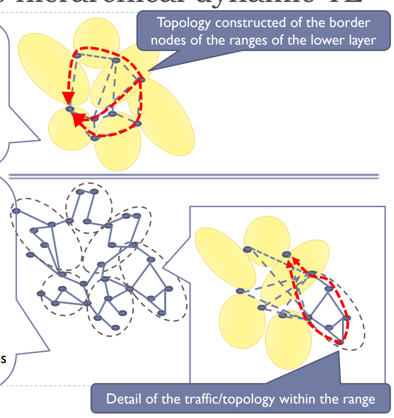
Overview of the hierarchical dynamic TE

- Divide the network hierarchically into multiple ranges
- Deploy a control server at each range
- Range of the lower layer
  - Constructed from small number of nodes
  - Server maintains the detail of the topology and link load information within the range
- Range of the upper layer
  - Constructed from multiple ranges of the lower layer
  - Server maintains the aggregated information obtained from the servers of the lower layer

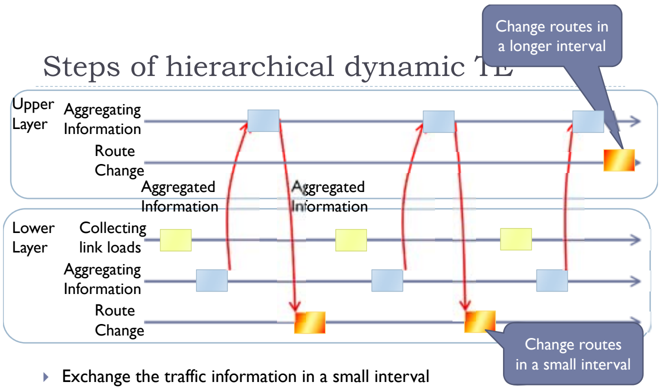


Overview of the hierarchical dynamic TE

- Upper layer
  - Maintains the aggregated traffic information
  - Routes are calculated over the aggregated topology
- Lower layer
  - Maintains the detail of the traffic/topology information
  - Changes the following routes
    - The routes within the range
    - Border nodes passed by the flow from/to the nodes within the range



Steps of hierarchical dynamic TE



- Exchange the traffic information in a small interval
  - Only the aggregated information is exchanged
- We can change the routes in a small interval at the lower layer
  - The number of flows affected by the lower layer is small

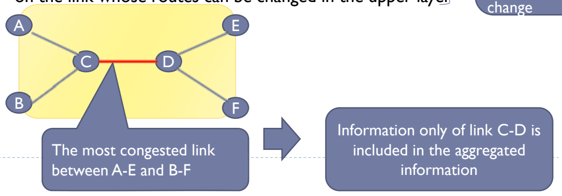
Information Aggregation

- Method to aggregate traffic information
  - Method should aggregate traffic so that
    - We can detect congestion from only the aggregated information
    - We can forecast the link utilization after the route change from only the aggregated information
- Method to change routes using the aggregated information
  - We can avoid the new congestion by forecasting the link loads from the aggregated information

Method to aggregate traffic Information

- We select the most congested link passed by each flow between border nodes
- We include the following information for each selected link
  - Total traffic amount on the link
  - Fraction of traffic of between the pair of border nodes whose flow passing the link
  - Upper/Lower bounds of the traffic amount of flows on the link whose routes can be changed in the upper layer

Used to forecast the link utilization after the route change



Forecast the link utilization after the route change

- Obtained by solving the linear programming
  - Objective
    - maximize $\sum_p F_{p,l}^{new} T_p + T_l^{lower}$ (Traffic on link l)
  - Constraints
    - $\sum_p F_{p,l}^{old} T_p + T_l^{lower} = X_l^{all}$ (Sum of the traffic on link l before the route change)
    - $\forall l:\ X_l^{min} \le \sum_p F_{p,l}^{old} T_p \le X_l^{max}$ (Sum of the traffic whose routes can be changed)

Input
- Route after the change: $F_{p,l}^{new}$
- Route before the change: $F_{p,l}^{old}$
- Aggregated Information
  - Traffic amount on link l: X_l^{all}
  - Upper/Lower Bounds of the traffic whose route can be changed: X_l^{max}, X_l^{min}

Variables
- Traffic of flow p: T_p
- Traffic on link l whose route cannot be changed: T_l^{lower}

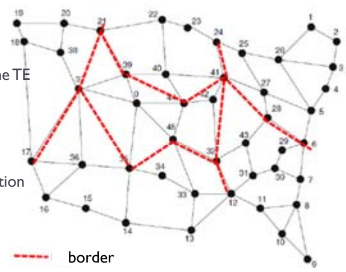
- We can forecast the link utilization after the route change from the aggregated information
  - We can change the routes without new congestion.

TE using the aggregated information

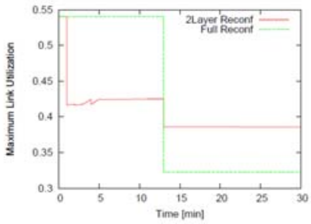
1. Detect the node pair where the congestion occurs
   - Detect the link whose link utilization exceeds the threshold
   - Node pair of the both edge of the detected links are regarded as the node pair where the congestion occurs
2. Identify the flows passing the congested links
   - Flows passing the detected node pairs are identified by using the routing information of the range
3. Calculate the new routes for the identified flows
   1. Calculate routes by the Shortest Path First
   2. Calculate the upper bounds of the link utilization after the route change
   - Are all the calculated upper bounds less than the threshold? Yes → End; No →
   3. Delete links whose calculated upper bounds of the link utilization are lager than the threshold

Evaluation Environment

- Topology
  - US Topology
- Initial routes
  - Initial traffic: generated randomly so as to follow the lognormal distribution
  - Initial routes: set so as to accommodate the initial traffic without congestion
- Traffic during the TE
  - Newly generated so as to follow the lognormal distribution
  - The traffic is constant during the TE
- Interval of the route change
  - Lowest layer: 1minute
  - Top layer: 13 minutes
- Compared method
  - Method using whole link utilization
    - Interval : 13 minutes



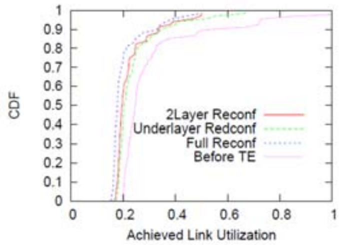
Link Utilization after the traffic change



- Our method reduce the link utilization immedately after the traffic change
  - Due to the small interval of the route change
- Our method reduce the link utilization more by the route changes of the top layer.

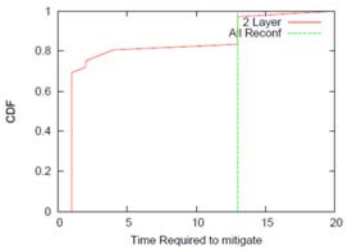
Achieved Link utilization

- Our method reduces the maximum link utilization significantly even by the route changes of the lowest layer only
- Our method reduces the maximum link utilization more by changing the routes also in the top layer
  - The achieved link utilization by our method is similar to that of the method using the whole information



Time required to make the link utilization less than 0.4

- Our method reduces the link utilization soon after the traffic change
  - Our method reduces the link utilization significantly even by the route changes of the lowest layer only



Conclusion and future work

- Conclusion
  - Propose the heretical TE
    - Method to aggregate traffic information
    - Method to change the routes using the aggregated traffic information
  - Our method reduce the link utilization soon after the traffic change
  - Our method achieves the link utilization similar to that achieved by the method using the all link loads
- Future work
  - Evaluation of our method in the larger topology