Walking the tightrope: Responsive yet stable Traffic Engineering

Presented by Aparna Sundar
Problem Definition and current solutions

TE problem definition

Offline Methods – OSPF-TE, MPLS

Online Methods – MATE, TeXCP
Inadequacy of Offline methods

- Cannot react to real-time traffic reroutes.
- Load distribution not guaranteed to be optimal.
- Suboptimal reaction to failure.
Online methods

- Should react to real-time traffic demands and failures.
- Prior approaches – centralized, assuming a global oracle, lacking stability analysis. Eg MATE.
- TeXCP – distributed and stable.
Summary of Results

- For same traffic demands, TeXCP supports same utilisation and failure resilience with a third of the capacity as traditional offline methods.
- Network utilization is always within a few percentage points of optimal value.
- Prefers shorter routes while trimming long routes that are not useful.
Big Picture

• Two Components

• Load Balancer: multiple paths delivering demands from ingress to egress router, moving traffic from over-utilized to under utilized paths.

• Closed loop Feed Back controller: collects network feedback at faster time scale than LB to ensure traffic stability.
Diagram. Explanation of terms

Figure 1: For each Ingress-Egress (IE) pair, there is a TeXCP agent at the ingress router, which balances the IE traffic across available paths in an online, distributed fashion.

<table>
<thead>
<tr>
<th>Var</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_s$</td>
<td>Total Traffic Demand of IE pair $s$</td>
</tr>
<tr>
<td>$P_s$</td>
<td>Set of paths available to IE pair $s$</td>
</tr>
<tr>
<td>$r_{sp}$</td>
<td>Traffic of IE pair $s$ sent on path $p$, i.e., $R_s = \sum r_{sp}$</td>
</tr>
<tr>
<td>$x_{sp}$</td>
<td>Fraction of IE, $s$, traffic sent on path $p$, called path weight.</td>
</tr>
<tr>
<td>$u_{sp}$</td>
<td>The utilization of path $p$ observed by IE pair $s$</td>
</tr>
<tr>
<td>$u_l$</td>
<td>The utilization of link $l$</td>
</tr>
<tr>
<td>$C_l$</td>
<td>The capacity of link $l$</td>
</tr>
<tr>
<td>$P_l$</td>
<td>Set of paths that traverse link $l$</td>
</tr>
<tr>
<td>$\bar{u}_s$</td>
<td>Weighted average utilization of paths used by IE pair $s$</td>
</tr>
</tbody>
</table>

Table 2: The variables most-used in the paper. All of these variables are functions of time.
LP Formulation at each IE pair

\[
\min \max_{x_{sp}} \max_{l \in L} w_{l}, \quad (1)
\]

subject to the constraints:

\[
u_{l} = \sum_{s} \sum_{p \in P_s, p \in l} \frac{x_{sp} \cdot R_{s}}{C_{l}}, \quad (2)
\]

\[
\sum_{p \in P_s} x_{sp} = 1, \quad \forall s, \quad (3)
\]

\[
x_{sp} \geq 0, \quad \forall p \in P_s, \forall s. \quad (4)
\]
Path Selection, Probing Network state

- **Path Selection:**
  - ISP picks set of K shortest paths that it can use.

- **Probing Network state:**
  - Maintain probe timer, $T_p$, to maintain track of path utilization. $T_p > RTT$.
  - Probe packet with updatable utilization field sent by ingress node. Egress node sends it back to app agent.
  - Probe loss: estimate util to $\max(1, p u_{sp})$
Load Balancer

- Each agent maintains a decision timer, which fires every $T_d$ sec, $> 5T_p$.
- Each time the agent computes change in fraction of IE traffic sent on path $p$.
- At eqbm, $x_{sp}$ is constant, traffic is conserved, no negative traffic possible, updates should decrease max utilization.
Load Balancer (contd)

\[ \Delta x_{sp} = \begin{cases} \frac{r_{sp}}{\sum_{p'} r_{sp'}} (\bar{u}_s - u_{sp}) & \forall p, u_{sp} > u_{min} \\ \frac{r_{sp}}{\sum_{p'} r_{sp'}} (\bar{u}_s - u_{sp}) + \varepsilon & p, u_{sp} = u_{min}. \end{cases} \]

- Intuition: path whose util is greater than avg shd dec its rate while path whose util is below avg should increase its rate.
- Change in traffic is proportional to current traffic on path (which is prop to util).
- Use of epsilon – to re-use or restart util on a path.
Preventing Oscillations, Managing Congestion

- Two agents working independently may shift flow to link that was previously under-utilized.

- Solution (inspired by XCP)
  - Compute aggregate feedback: \( \Phi = \alpha \cdot T_p \cdot S - \beta \cdot Q \),
  
- Compute per IE flow feedback based on a Max-Min approach:
  
  \( \Phi \geq 0 \Rightarrow \delta^+ = \frac{\Phi}{N}, \delta^- = 0 \),
  
  \( \Phi < 0 \Rightarrow \delta^+ = 0, \delta^- = \frac{\Phi}{\phi_i} \),

- Positive feedback added, negative multiplied.
Preventing Oscillations, Managing Congestion (contd)

- Sending feedback to agents using probe. \( g_{sp} = g_{sp} + \delta^+ - \delta^- \times g_{sp} \)
- \( g_{sp} \) is allowed rate on path \( p \).
- Actual rate = \( \min(g_{sp}, x_{sp} R_s) \).
- Prefer to use shorter paths: Use weighted max-min fairness to push a preference for shorter paths.
- Heuristic: Shorter paths better for better network utilizations

---

<table>
<thead>
<tr>
<th>Used for load balancing</th>
<th>PATH_UTILIZATION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>POSITIVE_FEEDBACK</td>
</tr>
<tr>
<td></td>
<td>NEGATIVE_FEEDBACK</td>
</tr>
</tbody>
</table>

| Used to prefer shorter paths | WEIGHT |

**Figure 2**: Probe Packet. Feedback is returned in two fields because Positive Feedback is additive while Negative Feedback is multiplicative.
Analysis

- Computation of explicit feedback for each pair, by load balancer, that leads to more stable per-IE flow rates and subsequently utilizations.
- Effect of feedback on network, “mostly” done by the time load balancer kicks into action ie, the explicit feedback brings path util to 90% of desired value, before the next time any of the load balancers need to make a decision.
Results and Comparision

Figure 4: When traffic matches TM, TeXCP results in a max-utilization within a few percent of the optimal, and much closer to optimal than OSPF-TE or InvCap. Figure shows both average (thick bars) and maximum (thin bars) taken over 40 TMs.

Figure 5: When traffic deviates from TM by a margin, TeXCP stays within a few percent of the optimal max-utilization; OSPF-TE_base and OSPF-TE_Multi-TM lead to much larger max-utilization.
Results and comparison (contd)

Figure 6: Under failures, TeXCP's max-utilization is within a few percent of the optimal; InvCap, OSPF-TE_base, and OSPF-TE_failures become highly suboptimal. Figure shows the 90th percentile (thick) and maximums (thin) taken over multiple TM's.
Discussion

- Look at source-dest paths, instead of ingress-egress paths?
- Metric for network utilization
  
  \[ \text{Metric} = \frac{\text{max-utilization}_{\text{Tech.}}}{\text{max-utilization}_{\text{Oracle}}} \]

- Including estimate for egress-ingress links?