Overlayed Path Segment Forwarding Problem Statement

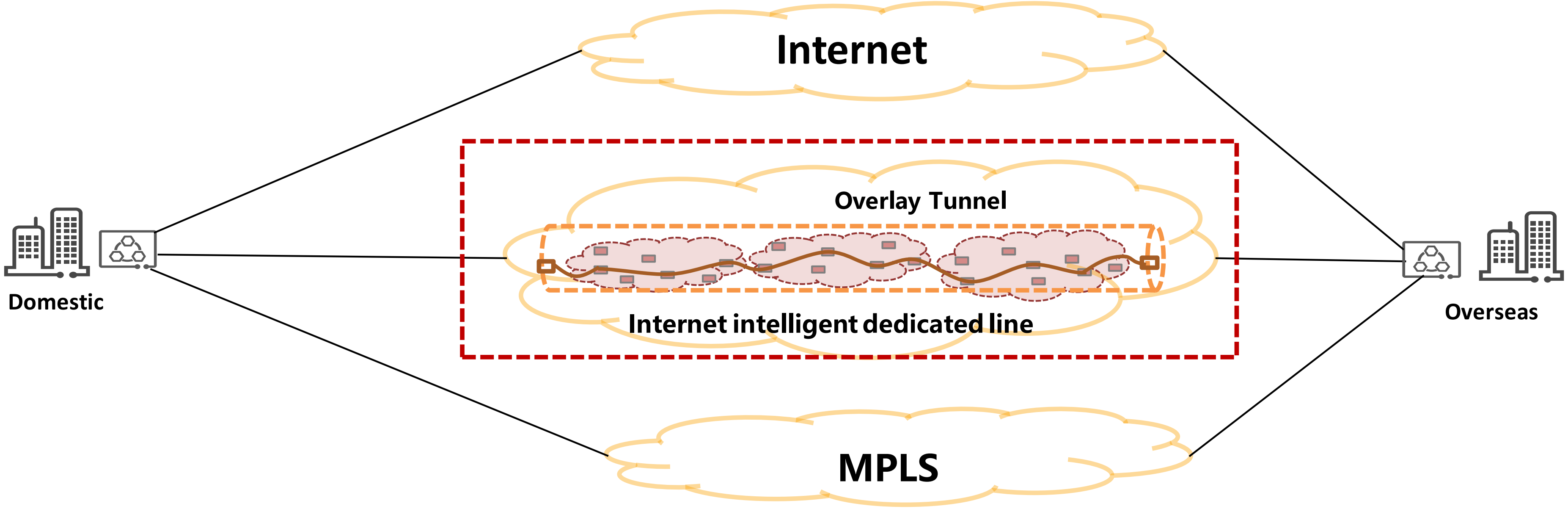
draft-li-overlayed-path-segment-forwarding

Yizhou Li

Xingwang Zhou

Carsten Bormann

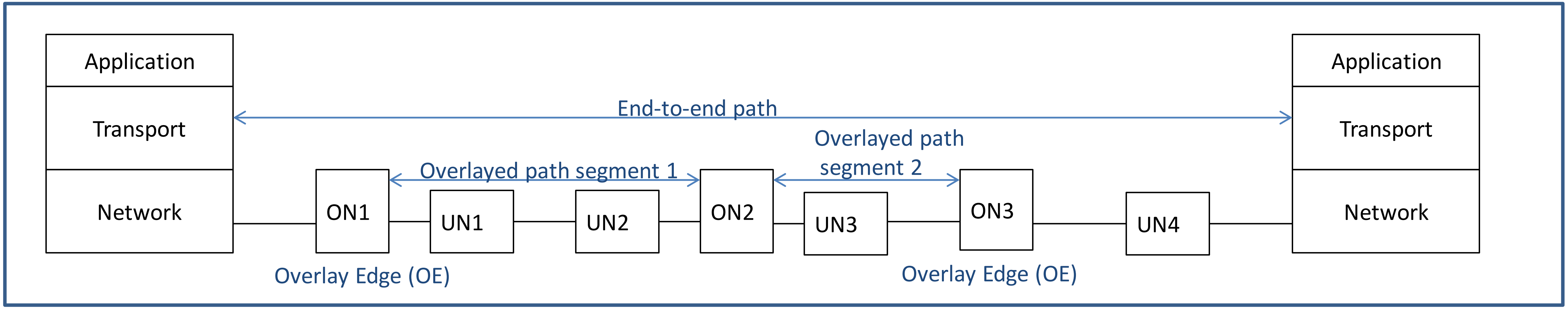
Motivation: Leverage cloud router nodes for best path selection to provide performance closer to leased lines



- Default path does not always give the best latency and throughput
- Now practical: Build a better path via nodes in different geographic sites in the cloud (inexpensive, easy provisioning and scaling,
  instances with "enhanced network performance" available from cloud provider)
- Experiments: 71% chance of finding a better overlay path based on 37 cloud routers globally

Take this opportunity to do **Localized Optimizations On Path Segment (LOOPS)** for better reliability and throughput

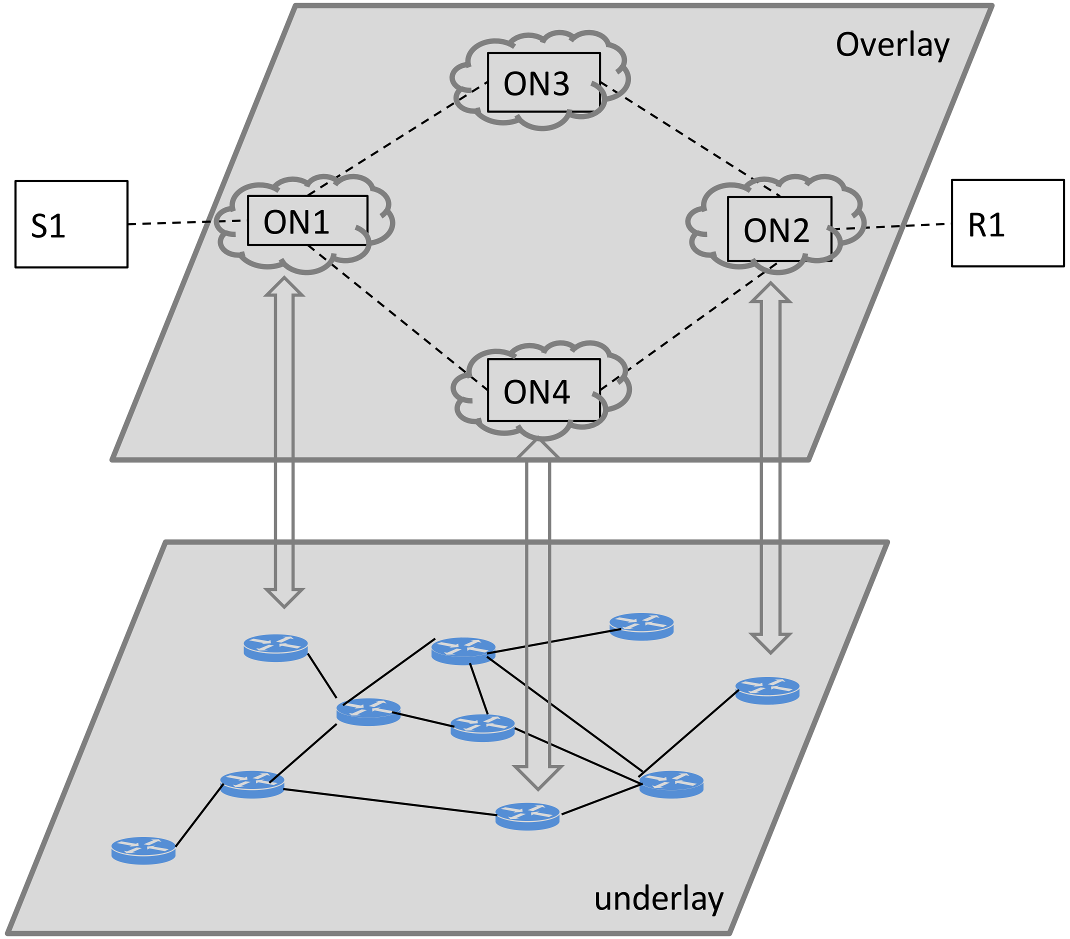
ON - Overlay node
UN - Underlay node



Problems/opportunities:

- Slow recovery over long haul
- Inaccuracy in sending rate decrease at sender
- Impairment/Temporary outage of virtual hop
- Limited capacity of virtual nodes

Elements of a solution



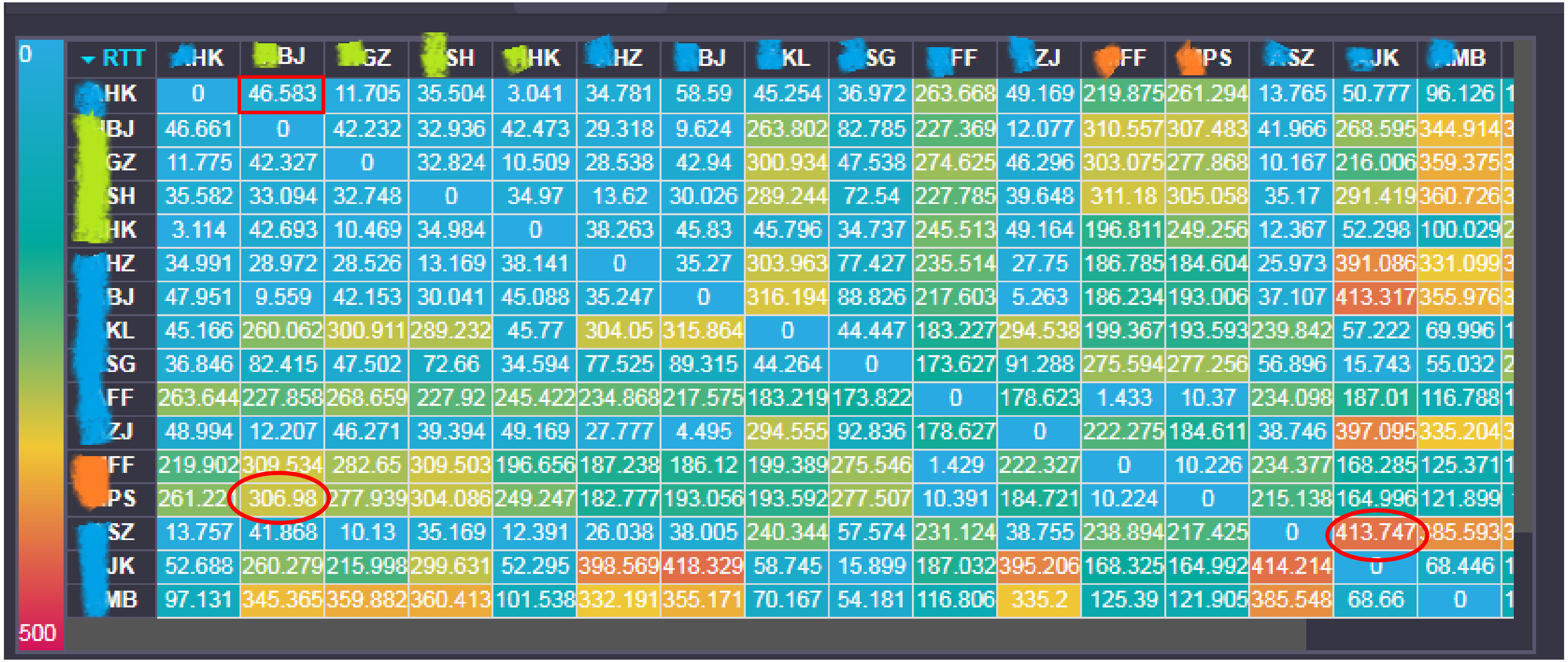
1. Local recovery
   - For entire tunnel (rather than individual flow)
   - Loss detection/indication
   - Measure segment RTT
   - Limited retransmission attempts
   - Control FEC/replication intensity
2. Congestion control interaction
   - Export appropriate CC signaling from LOOPS to e2e transport
   - Support ECN
3. Traffic splitting/recombining
   - For capacity
   - FEC over multiple path segments

Side meeting

- Title: Localized Optimizations On Path Segment (LOOPS) Discussion
- Time: Tuesday (Nov 6) 18:30-19:30 (19:30-20:00 as buffer)
- Room: “Meeting 5” (7th floor)
- Purpose: discuss use cases and problems, potential solution ideas, what should and could be done in IETF
- Related drafts:
  - Overlayed Path Segment Forwarding (OPSF) Problem Statement (https://tools.ietf.org/html/draft-li-overlayed-path-segment-forwarding-ps-00)
  - Sub-path Transport Layer Problem Statement (https://tools.ietf.org/html/draft-herbert-sub-path-ps-00)

backups

Delays over default path are not always promising

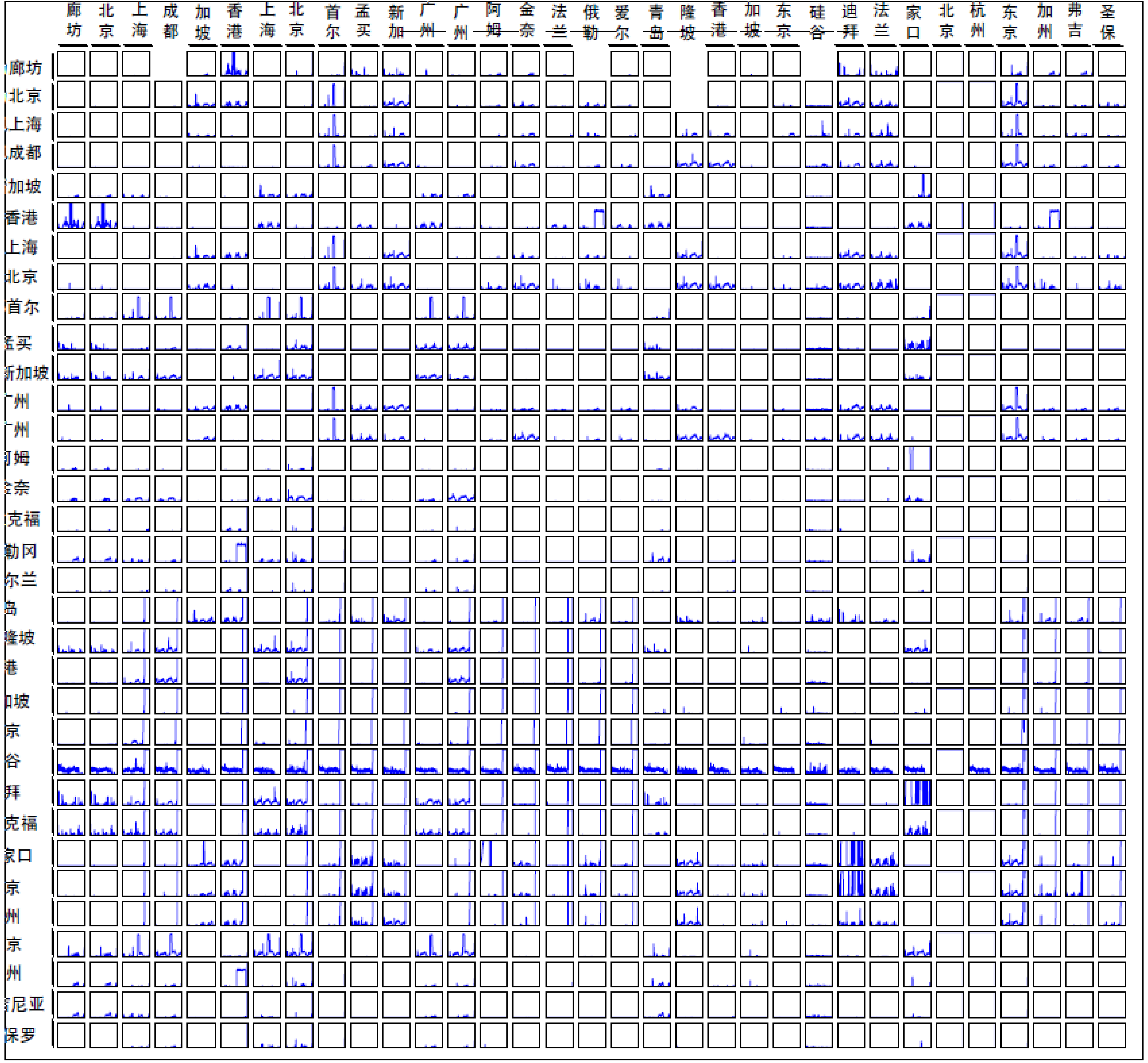


RTT	[redacted] HK	[redacted] BJ	[redacted] GZ	[redacted] SH	[redacted] HK	[redacted] HZ	[redacted] BJ	[redacted] KL	[redacted] SG	[redacted] FF	[redacted] ZJ	[redacted] FF	[redacted] PS	[redacted] SZ	[redacted] JK	[redacted] MB	
[redacted] HK	0	46.583	11.705	35.504	3.041	34.781	58.59	45.254	36.972	263.668	49.169	219.875	261.294	13.765	50.777	96.126	1
[redacted] BJ	46.661	0	42.232	32.936	42.473	29.318	9.624	263.802	82.785	227.369	12.077	310.557	307.483	41.966	268.595	344.914	3
[redacted] GZ	11.775	42.327	0	32.824	10.509	28.538	42.94	300.934	47.538	274.625	46.296	303.075	277.868	10.167	216.006	359.375	3
[redacted] SH	35.582	33.094	32.748	0	34.97	13.62	30.026	289.244	72.54	227.785	39.648	311.18	305.058	35.17	291.419	360.726	3
[redacted] HK	3.114	42.693	10.469	34.984	0	38.263	45.83	45.796	34.737	245.513	49.164	196.811	249.256	12.367	52.298	100.029	2
[redacted] HZ	34.991	28.972	28.526	13.169	38.141	0	35.27	303.963	77.427	235.514	27.75	186.785	184.604	25.973	391.086	331.099	3
[redacted] BJ	47.951	9.559	42.153	30.041	45.088	35.247	0	316.194	88.826	217.603	5.263	186.234	193.006	37.107	413.317	355.976	3
[redacted] KL	45.166	260.062	300.911	289.232	45.77	304.05	315.864	0	44.447	183.227	294.538	199.367	193.593	239.842	57.222	69.996	1
[redacted] SG	36.846	82.415	47.502	72.66	34.594	77.525	89.315	44.264	0	173.627	91.288	275.594	277.256	56.896	15.743	55.032	2
[redacted] FF	263.644	227.858	268.659	227.92	245.422	234.868	217.575	183.219	173.822	0	178.623	1.433	10.37	234.098	187.01	116.788	1
[redacted] ZJ	48.994	12.207	46.271	39.394	49.169	27.777	4.495	294.555	92.836	178.627	0	222.275	184.611	38.746	397.095	335.204	3
[redacted] FF	219.902	309.534	282.65	309.503	196.656	187.238	186.12	199.389	275.546	1.429	222.327	0	10.226	234.377	168.285	125.371	1
[redacted] PS	261.22	306.98	277.939	304.086	249.247	182.777	193.056	193.592	277.507	10.391	184.721	10.224	0	215.138	164.996	121.899	1
[redacted] SZ	13.757	41.868	10.13	35.169	12.391	26.038	38.005	240.344	57.574	231.124	38.755	238.894	217.425	0	413.747	385.593	3
[redacted] JK	52.688	260.279	215.998	299.631	52.295	398.569	418.329	58.745	15.899	187.032	395.206	168.325	164.992	414.214	0	68.446	1
[redacted] MB	97.131	345.365	359.882	360.413	101.538	332.191	355.171	70.167	54.181	116.806	335.2	125.39	121.905	385.548	68.66	0	1

- Physical location matters but not always the top factor

* Around 120 virtual nodes.

Loss over default paths between node pairs has different characteristics and vary over time



Certain path has pretty high loss rate all the times

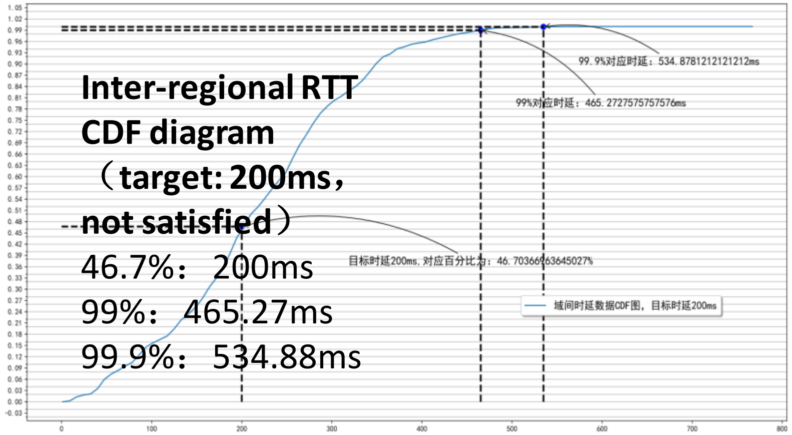
Collected over 3 days.

Overlay network performance analysis 1/2

- 3 sets of testings: Domestic, regional, inter-regional

20 sec as a cycle, 2000 Ping pkt each cycle, pairs out of 37 virtual nodes in cloud globally, 55 hours testing, metrics are loss rate and RTT,

compare with targeting QoS（domestic<40ms, regional<100ms, inter-regional<200ms, 99% percentile）

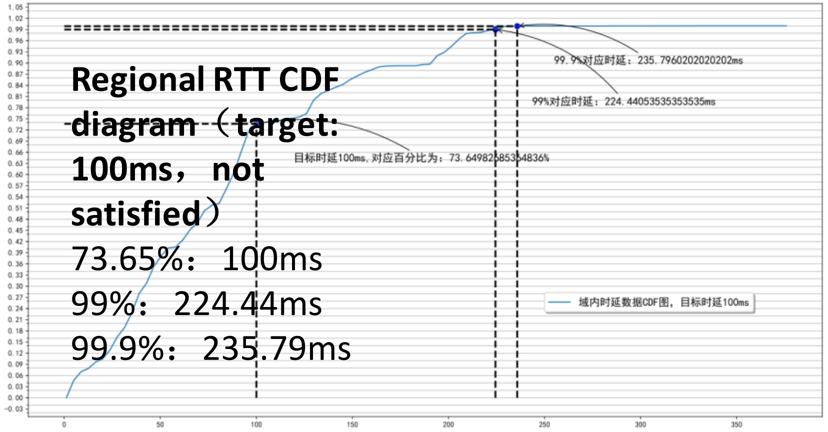


Inter-regional RTT CDF diagram（target: 200ms，not satisfied）

46.7%： 200ms

99%： 465.27ms

99.9%： 534.88ms

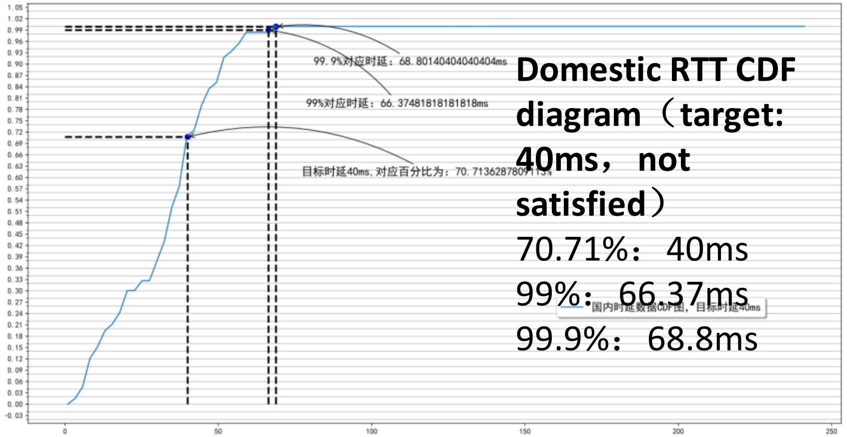


Regional RTT CDF diagram（target: 100ms， not satisfied）

73.65%： 100ms

99%： 224.44ms

99.9%： 235.79ms

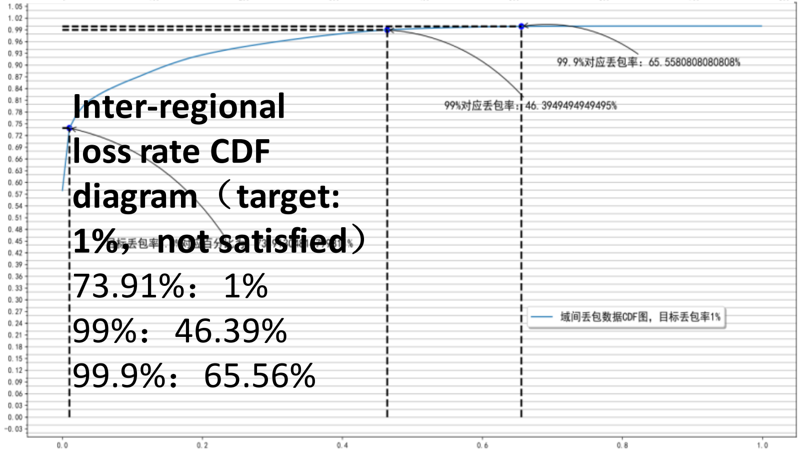


Domestic RTT CDF diagram（target: 40ms， not satisfied）

70.71%： 40ms

99%： 66.37ms

99.9%： 68.8ms

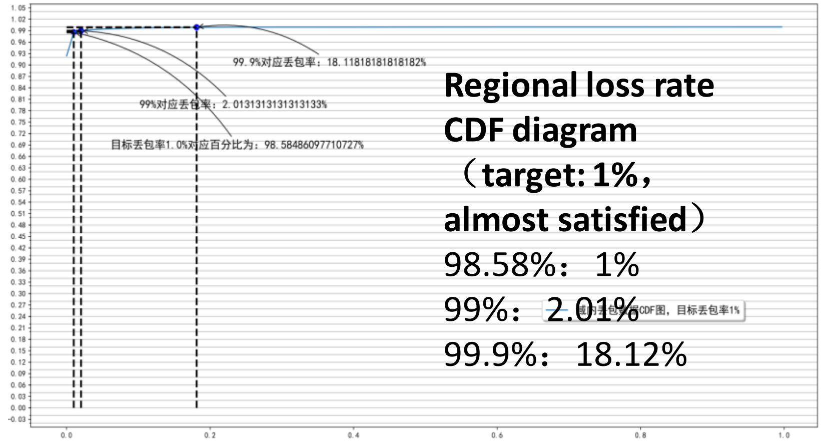


Inter-regional loss rate CDF diagram（target: 1%，not satisfied）

73.91%： 1%

99%： 46.39%

99.9%： 65.56%

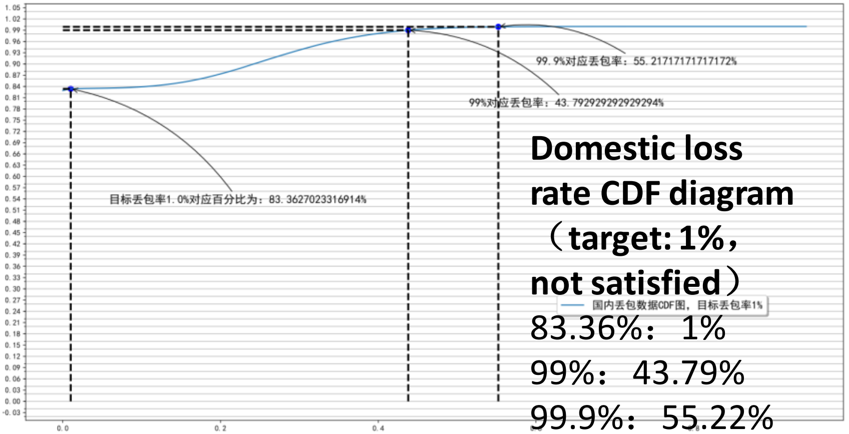


Regional loss rate CDF diagram（target: 1%，almost satisfied）

98.58%： 1%

99%： 2.01%

99.9%： 18.12%



Domestic loss rate CDF diagram（target: 1%，not satisfied）

83.36%： 1%

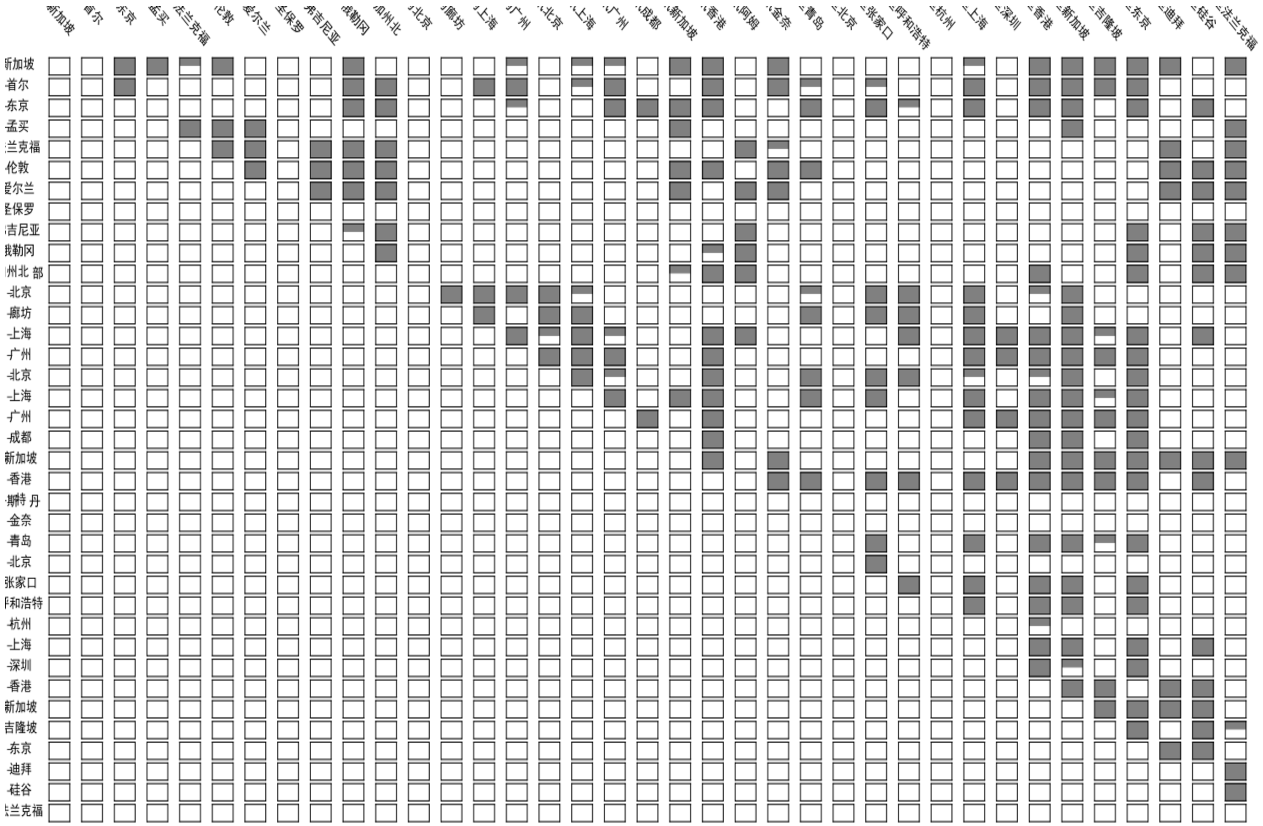
99%： 43.79%

99.9%： 55.22%

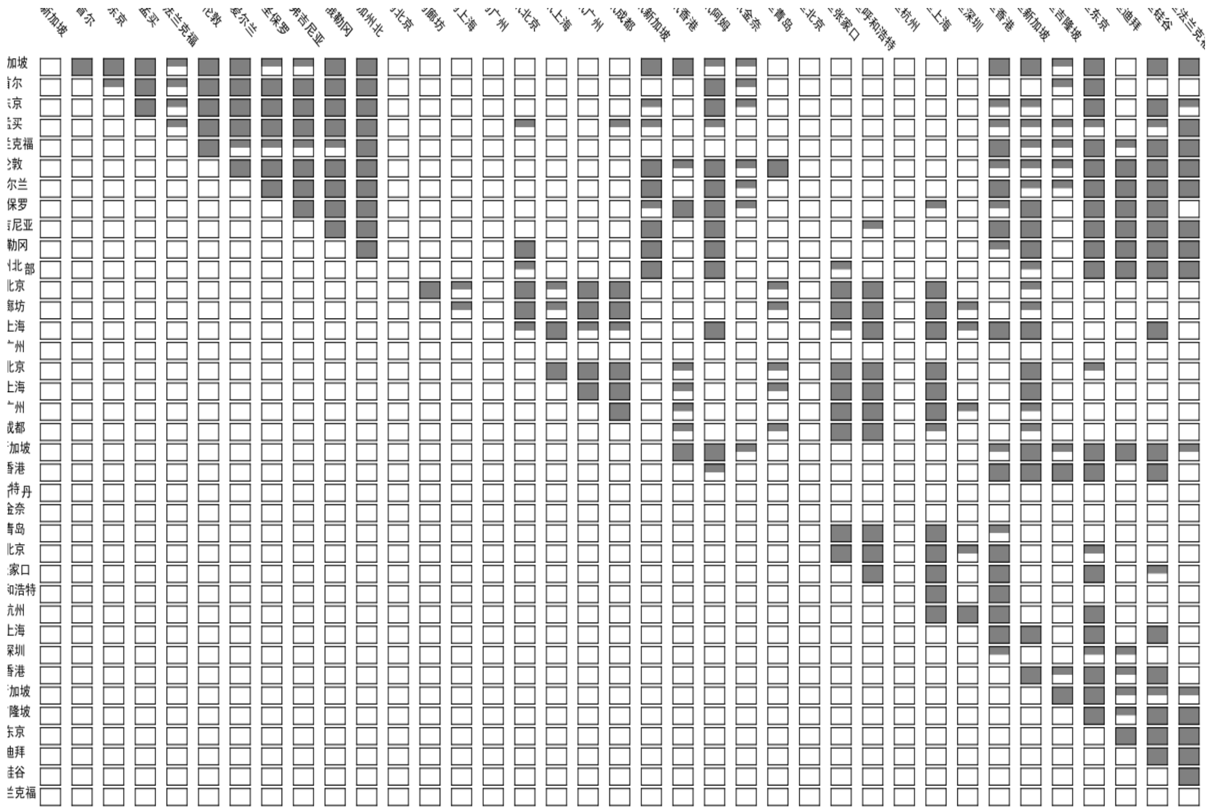
There is a gap between the performance of the default path and the target value.

Overlay network performance analysis 2/2 – Gap

upper right - **Gray: satisfied; White: not satisfied**
Upper half of grid: 99%， lower half of grid: 99.9%



Gap of RTT

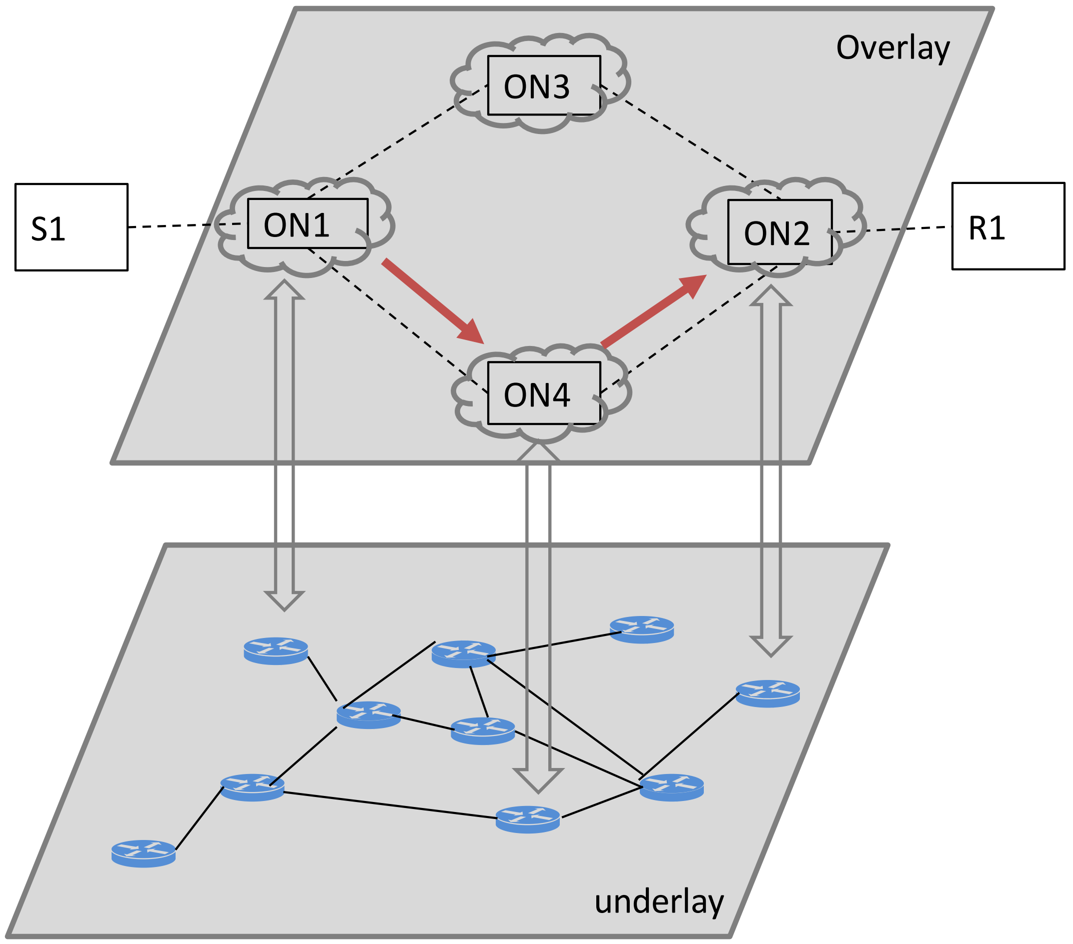


Gap of Loss Rate

All nodes
Satisfaction rate:
37.21% at 99%
32.81% at 99.9%

All nodes
Satisfaction rate:
44.27% at 99%
29.51% at 99.9%

Elements of a solution 1/3 – Local recovery

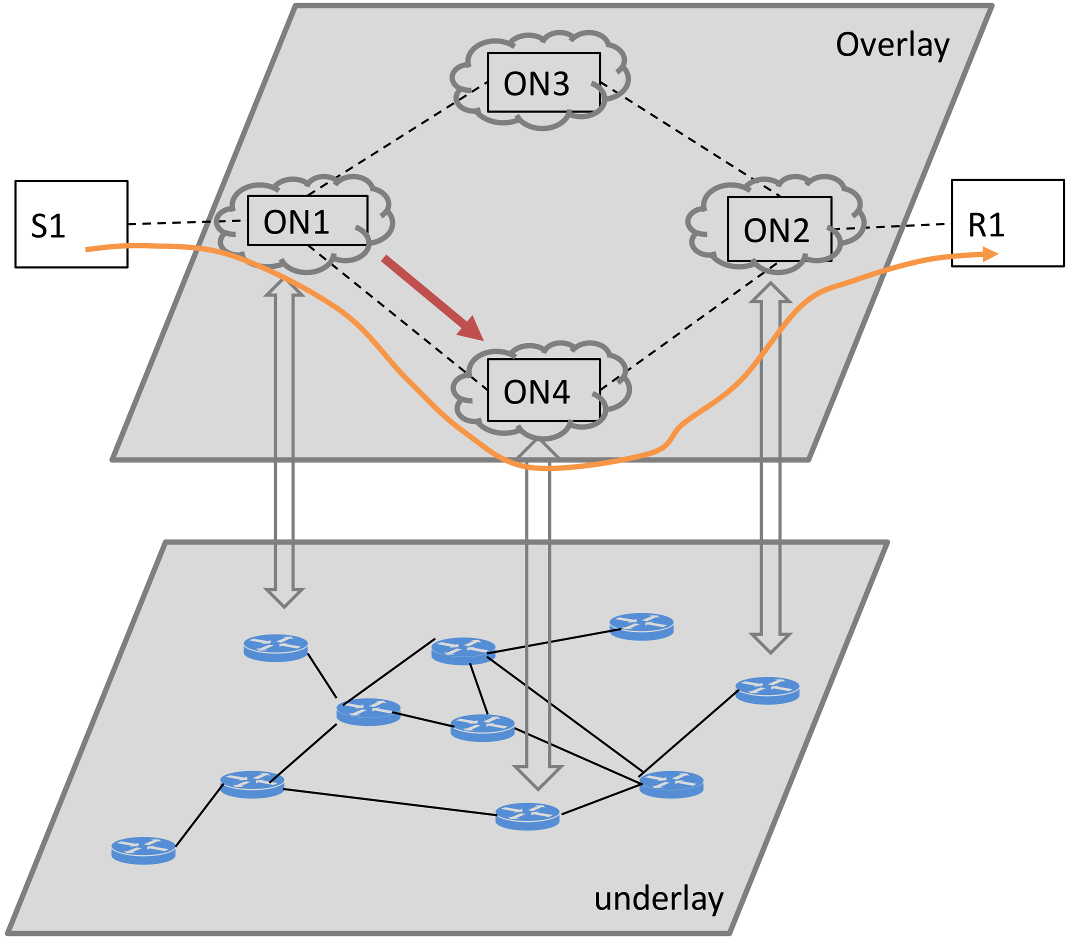


1. Path segment between two ONs maintains sequence numbers on packet base.
2. Measure segment RTT: use real traffic, in-band timestamp.
   - iOAM-like timestamp?
   - ACK/NACK to indicate the lost packets
   - Timestamp echoed back.

Issues:

1. Retransmitted packet may increase RTT variation at the sender.
   - Wireless has similar issue, new researches are targeting it
   - Optimize RTT measurement?
2. Hurt other non locally recovered flows
   - Use ECN to implicitly adjust sending rate?
3. Out of order pkt buffer at the egress edge
   - With buffer: ensure in-sequence; increase RTT variation; block other flows
   - Without buffer: out-of-sequence, rtx of both local and end-to-end (could be handled by well behaved TCP sender)

Elements of a solution 2/3 – Congestion Control interaction



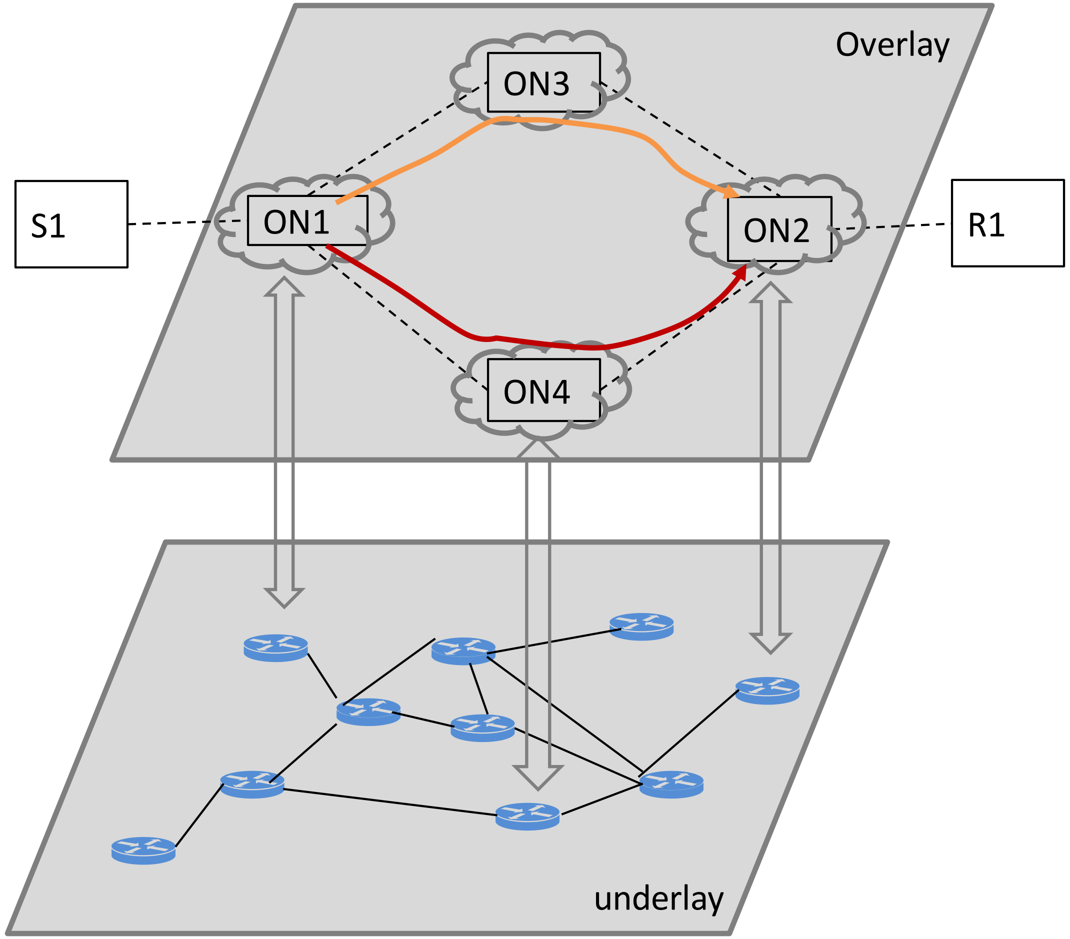
1. Congestion awared on the segment by ONs.
2. Congestion information should be delivered to the sender if CC would be needed. E.g. ECN

Issues:

1. Local retransmission attempts should be limited. How persistent should it be?
   - Number of attempts? time?
   - Remaining rtx credit based on application requirement?

* RFC3366 Advice to link designers on link Automatic Repeat reQuest (ARQ)

Elements of a solution 3/3 – Traffic splitting/recombining



1. During impairment, replicating packet could be enabled to allow using two disjoint paths.
2. Virtual edge node (ON2) should remove/recombine the replication.

Issues:

1. Should complex topology be considered? Like multiple merge points?
2. Dynamic FEC over multiple path segments?