

ICON 2004 - IEEE International Conference On Networks

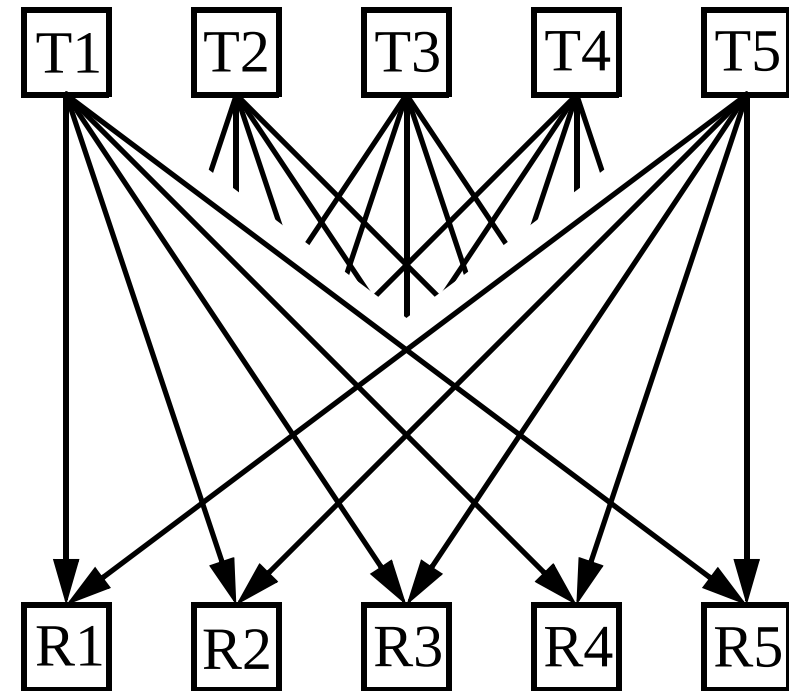
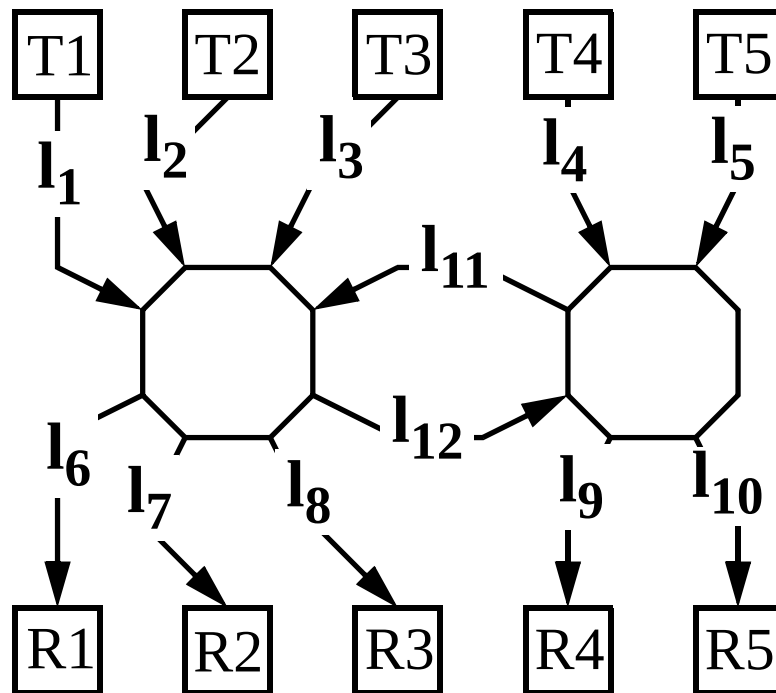
November 16-19, 2004, Singapore, Hilton

**EFFICIENT LIQUID SCHEDULE SEARCH STRATEGIES
FOR COLLECTIVE COMMUNICATIONS**

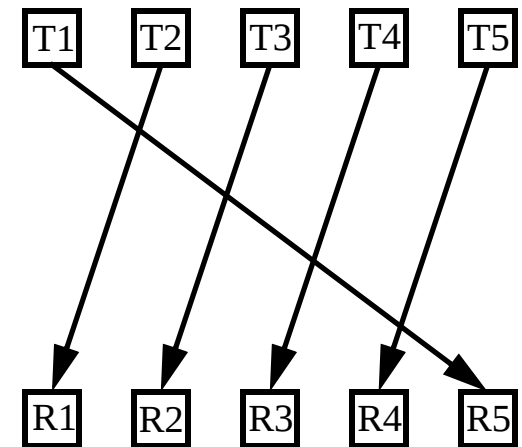
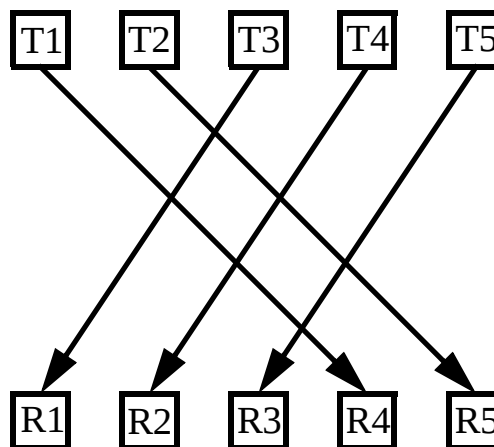
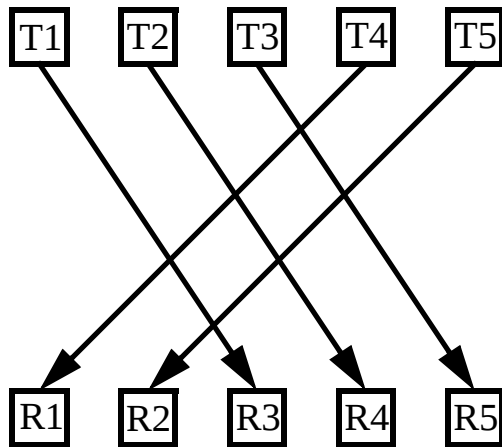
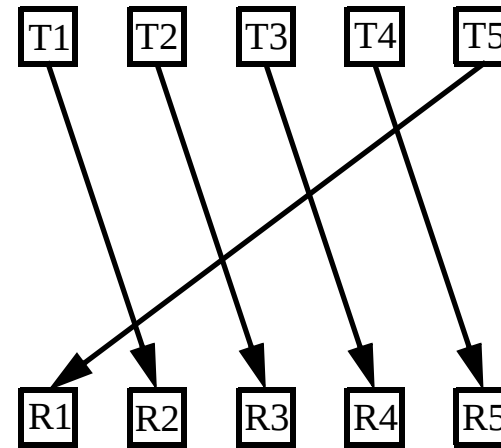
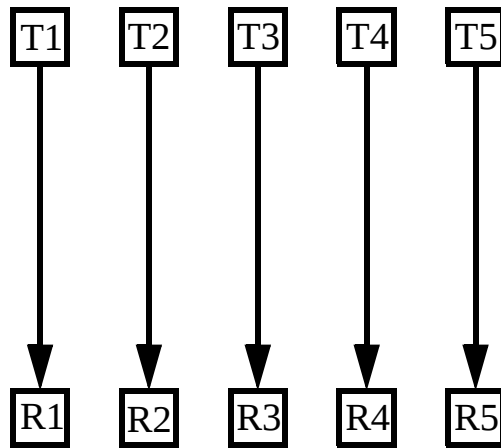
Emin Gabrielyan, Roger D. Hersch

Swiss Federal Institute of Technology - Lausanne

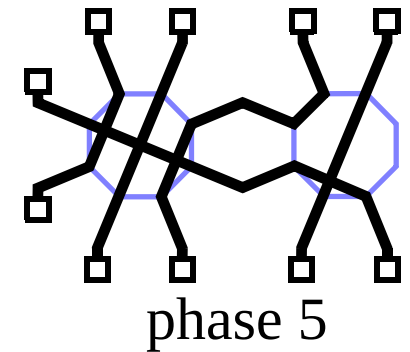
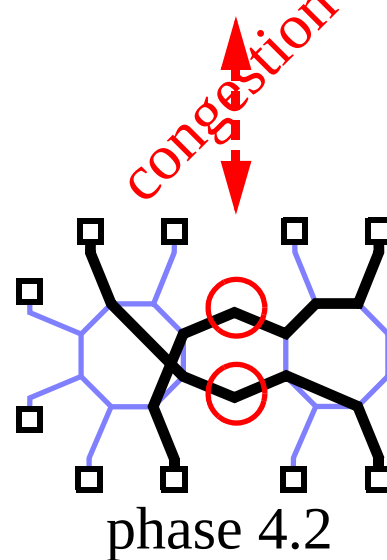
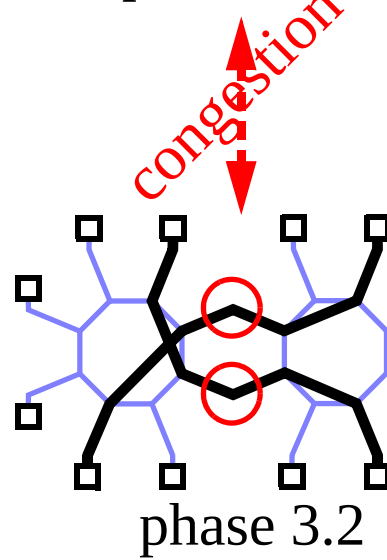
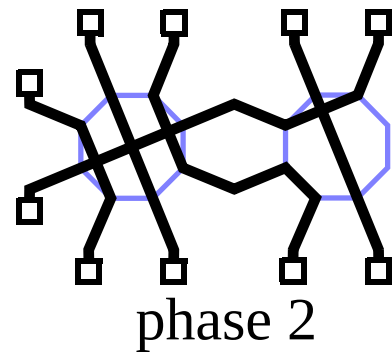
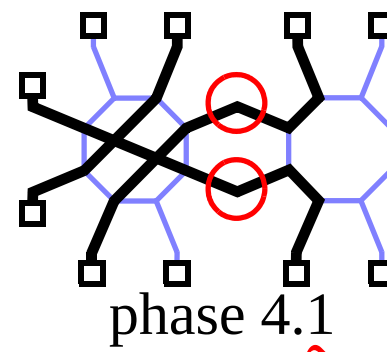
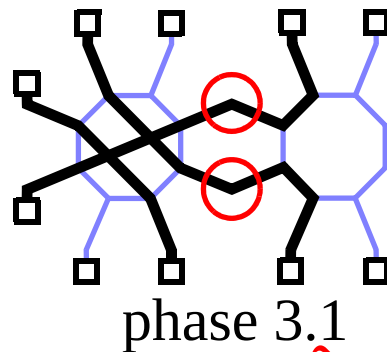
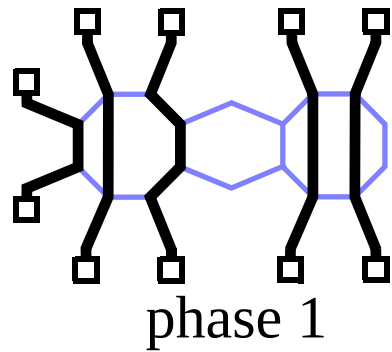
Example: 25 transmissions to be carried out



Round-robin schedule

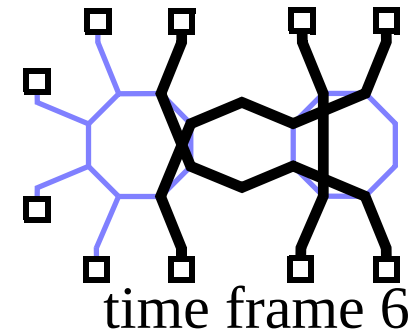
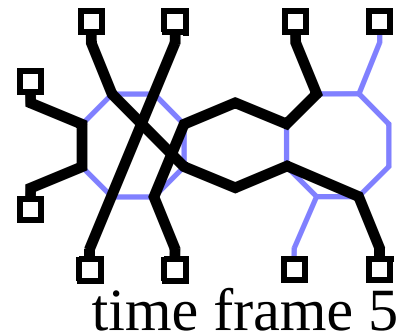
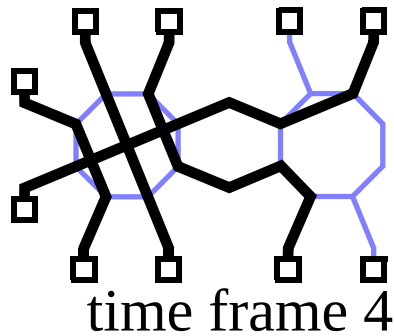
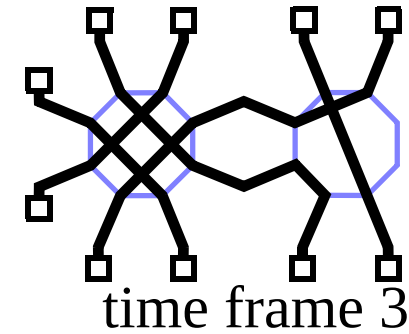
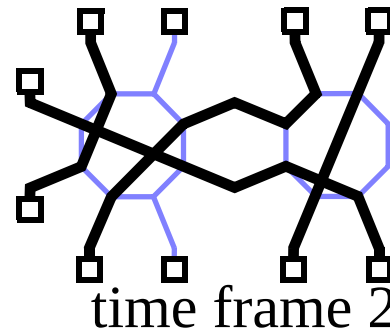
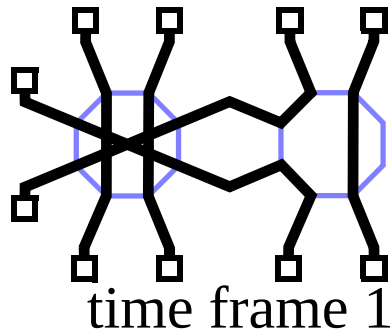


Round-robin Throughput



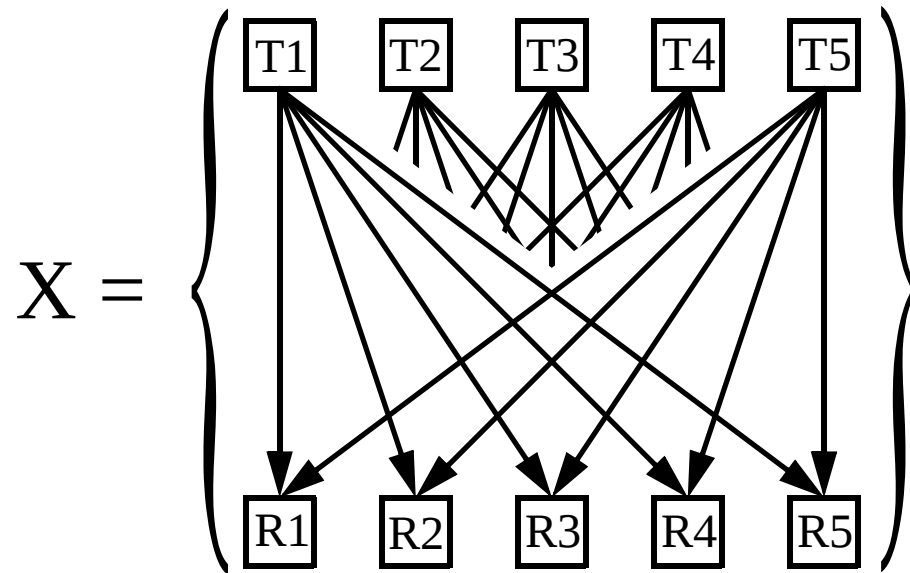
$$T_{roundrobin} = 25/7 \cdot 1Gbps = 3.57Gbps$$

Liquid schedule

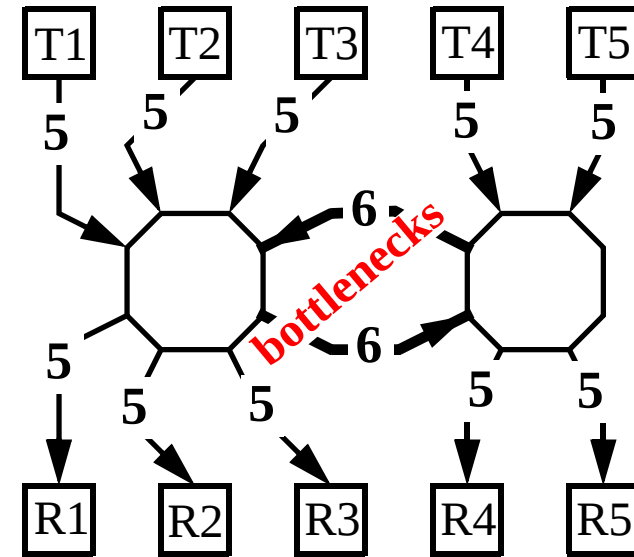


$$T_{liquid} = 25/6 \cdot 1Gbps = 4.16Gbps$$

Transfers and Load of Links



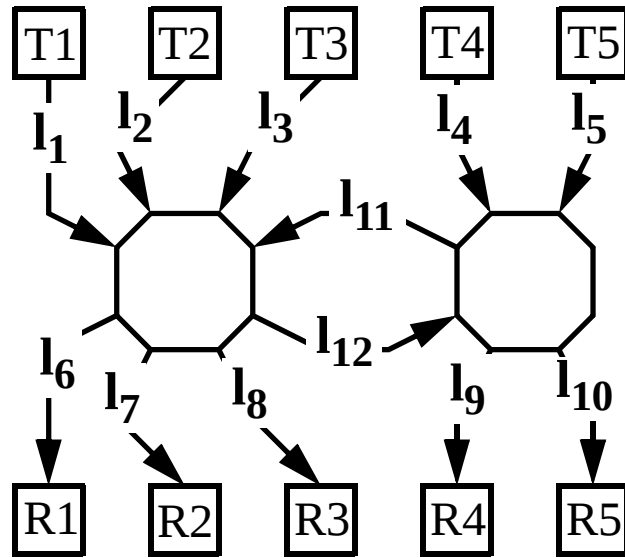
The 25 transfer traffic



$$\lambda(l_1, X) = 5, \dots \lambda(l_{12}, X) = 6$$

Transfers: $\{l_1, l_6\}, \dots \{l_1, l_{12}, l_9\}, \dots$

Duration of Traffic



$$\lambda(l_1, X) = 5, \dots \lambda(l_{10}, X) = 5$$

$$\lambda(l_{11}, X) = 5, \dots \lambda(l_{12}, X) = 6$$

$$\Lambda(X) = 6$$

$$X = \left\{ \begin{array}{l} \{l_1, l_6\}, \{l_1, l_7\}, \{l_1, l_8\}, \{l_1, l_{12}, l_9\}, \{l_1, l_{12}, l_{10}\}, \\ \{l_2, l_6\}, \{l_2, l_7\}, \{l_2, l_8\}, \{l_2, l_{12}, l_9\}, \{l_2, l_{12}, l_{10}\}, \\ \{l_3, l_6\}, \{l_3, l_7\}, \{l_3, l_8\}, \{l_3, l_{12}, l_9\}, \{l_3, l_{12}, l_{10}\}, \\ \{l_4, l_{11}, l_6\}, \{l_4, l_{11}, l_7\}, \{l_4, l_{11}, l_8\}, \{l_4, l_9\}, \{l_4, l_{10}\}, \\ \{l_5, l_{11}, l_6\}, \{l_5, l_{11}, l_7\}, \{l_5, l_{11}, l_8\}, \{l_5, l_9\}, \{l_5, l_{10}\} \end{array} \right\}$$

Liquid Throughput

$$X = \left\{ \begin{array}{l} \{l_1, l_6\}, \{l_1, l_7\}, \{l_1, l_8\}, \{l_1, l_{12}, l_9\}, \{l_1, l_{12}, l_{10}\}, \\ \{l_2, l_6\}, \{l_2, l_7\}, \{l_2, l_8\}, \{l_2, l_{12}, l_9\}, \{l_2, l_{12}, l_{10}\}, \\ \{l_3, l_6\}, \{l_3, l_7\}, \{l_3, l_8\}, \{l_3, l_{12}, l_9\}, \{l_3, l_{12}, l_{10}\}, \\ \{l_4, l_{11}, l_6\}, \{l_4, l_{11}, l_7\}, \{l_4, l_{11}, l_8\}, \{l_4, l_9\}, \{l_4, l_{10}\}, \\ \{l_5, l_{11}, l_6\}, \{l_5, l_{11}, l_7\}, \{l_5, l_{11}, l_8\}, \{l_5, l_9\}, \{l_5, l_{10}\} \end{array} \right\}$$

the throughput of a single link

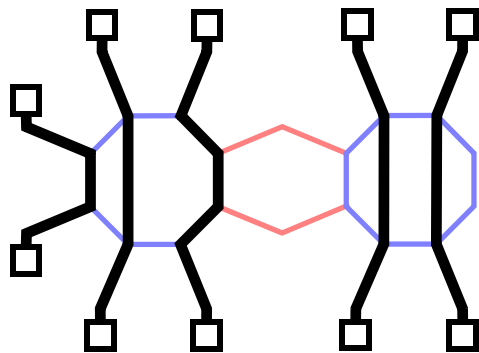
total number of transfers

$$T_{liquid} = \frac{\#(X)}{\Lambda(X)} \cdot T_{link} = \frac{25}{6} \cdot 1Gbps = 4.17Gbps$$

traffic's duration (the load of its bottlenecks)

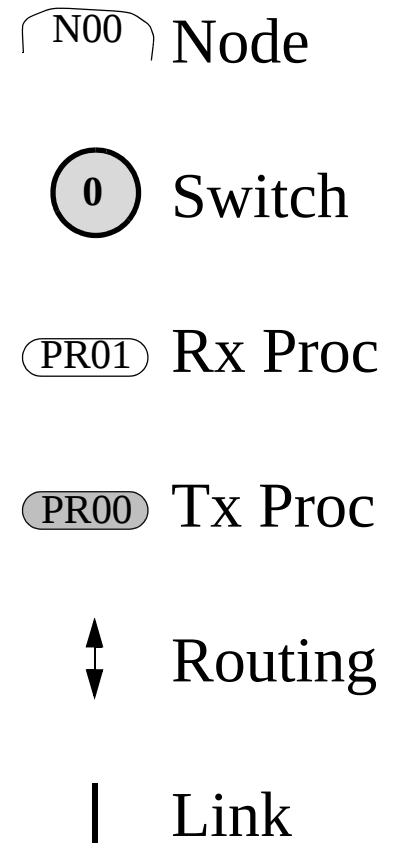
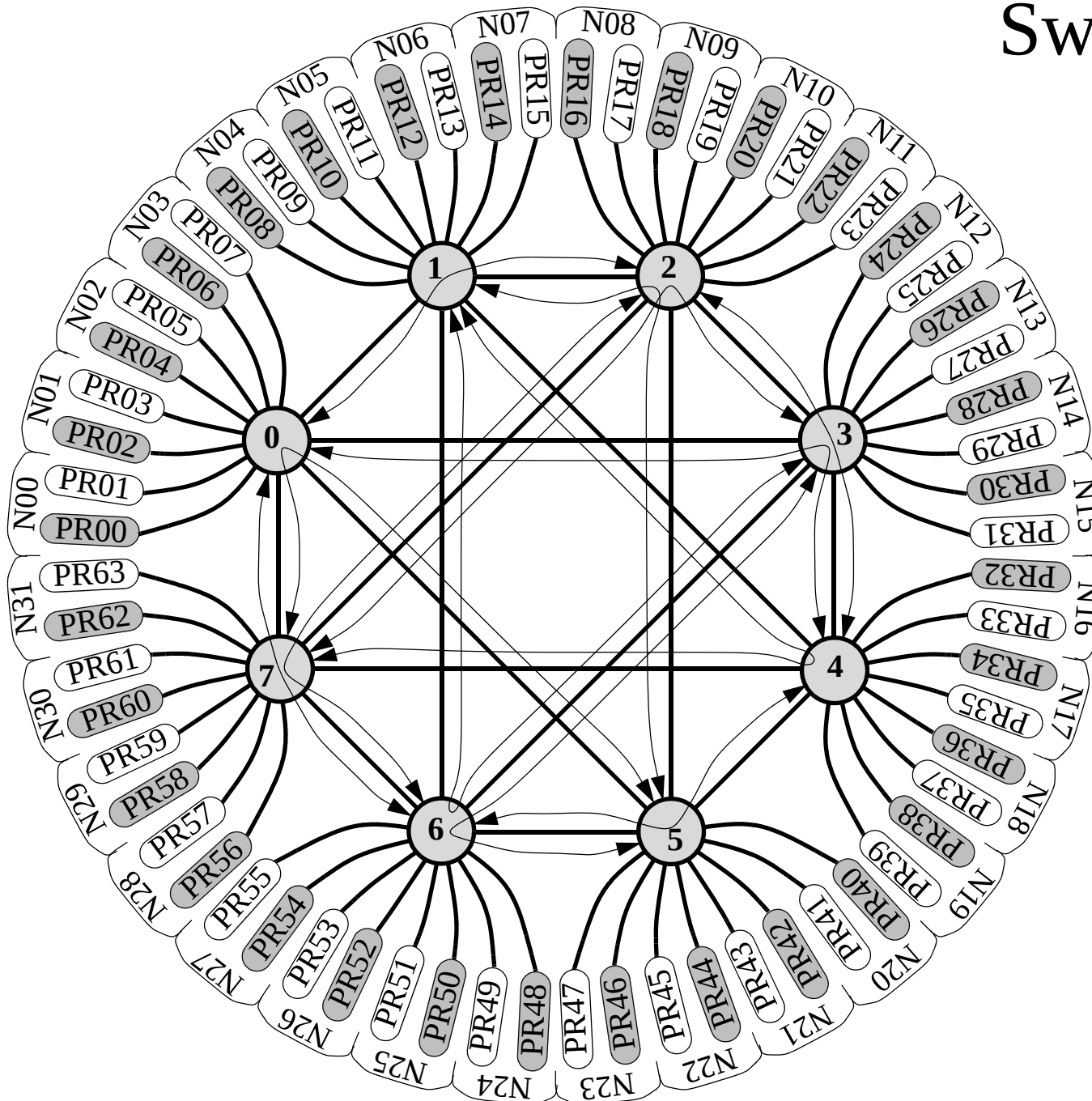
Schedules yielding the liquid throughput

$$X = \left\{ \begin{array}{l} \{l_1, l_6\}, \{l_1, l_7\}, \{l_1, l_8\}, \{l_1, \mathbf{l_{12}}, l_9\}, \{l_1, \mathbf{l_{12}}, l_{10}\}, \\ \{l_2, l_6\}, \{l_2, l_7\}, \{l_2, l_8\}, \{l_2, \mathbf{l_{12}}, l_9\}, \{l_2, \mathbf{l_{12}}, l_{10}\}, \\ \{l_3, l_6\}, \{l_3, l_7\}, \{l_3, l_8\}, \{l_3, \mathbf{l_{12}}, l_9\}, \{l_3, \mathbf{l_{12}}, l_{10}\}, \\ \{l_4, \mathbf{l_{11}}, l_6\}, \{l_4, \mathbf{l_{11}}, l_7\}, \{l_4, \mathbf{l_{11}}, l_8\}, \{l_4, l_9\}, \{l_4, l_{10}\}, \\ \{l_5, \mathbf{l_{11}}, l_6\}, \{l_5, \mathbf{l_{11}}, l_7\}, \{l_5, \mathbf{l_{11}}, l_8\}, \{l_5, l_9\}, \{l_5, l_{10}\} \end{array} \right\}$$

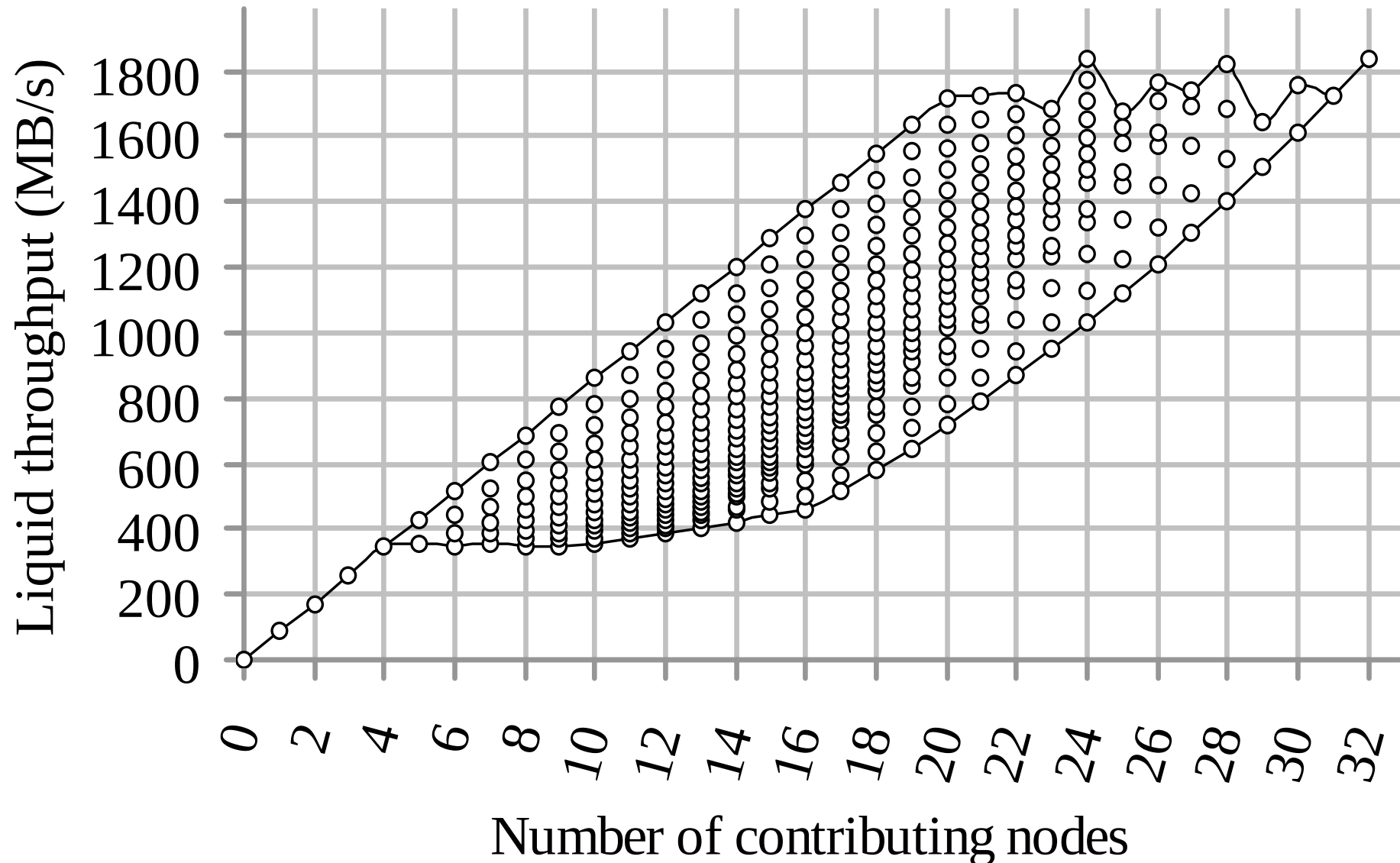


- Without a right schedule we may have intervals when the access to the bottleneck links is blocked by other transmissions.
- Our goal is to schedule the transfers such that all bottlenecks are always kept occupied ensuring that the liquid throughput is obtained.
- A schedule yielding the liquid throughput we call as a liquid schedule and our objective is to find a liquid schedule whenever it exists.

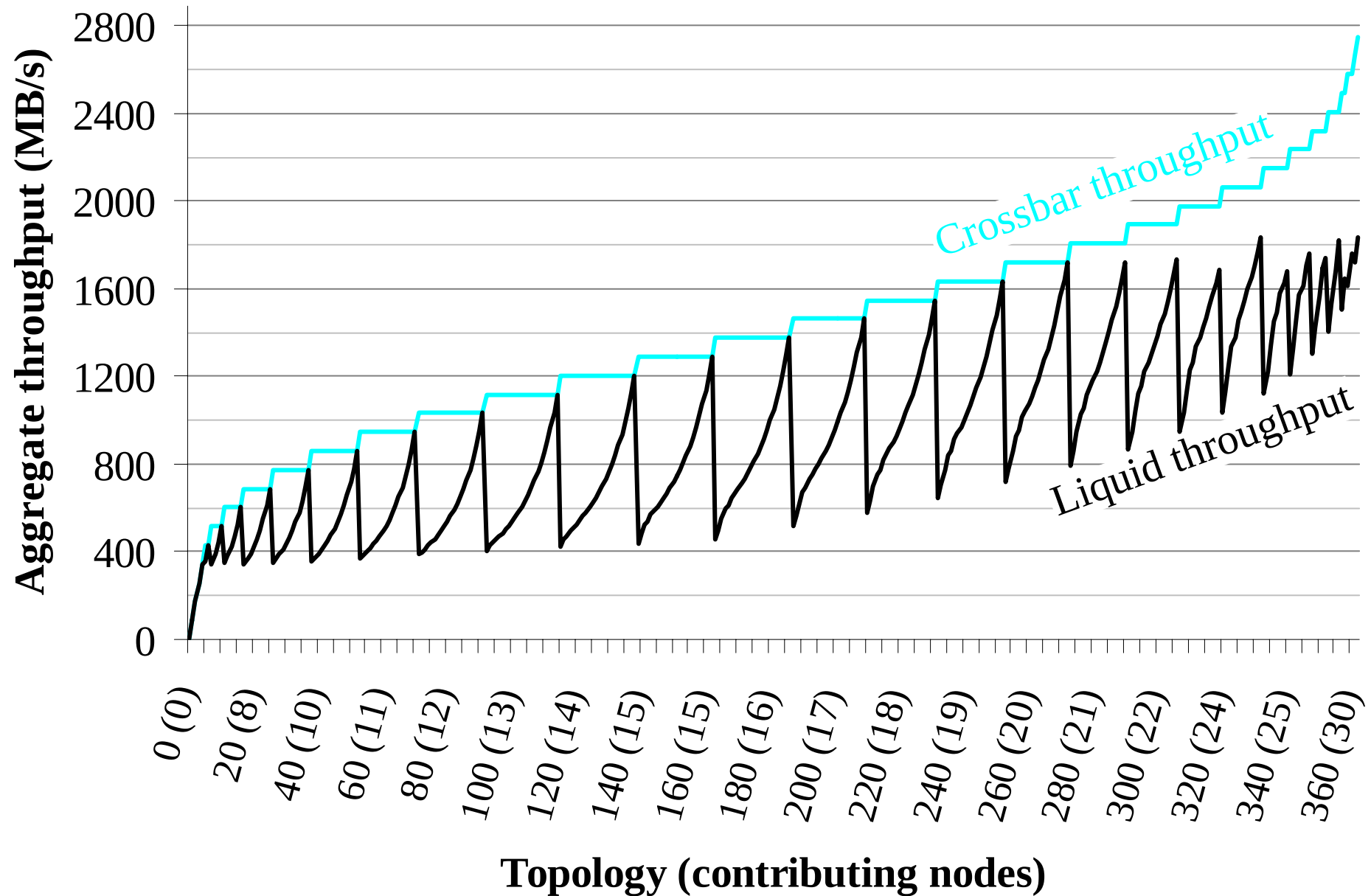
Swiss-T1 Cluster



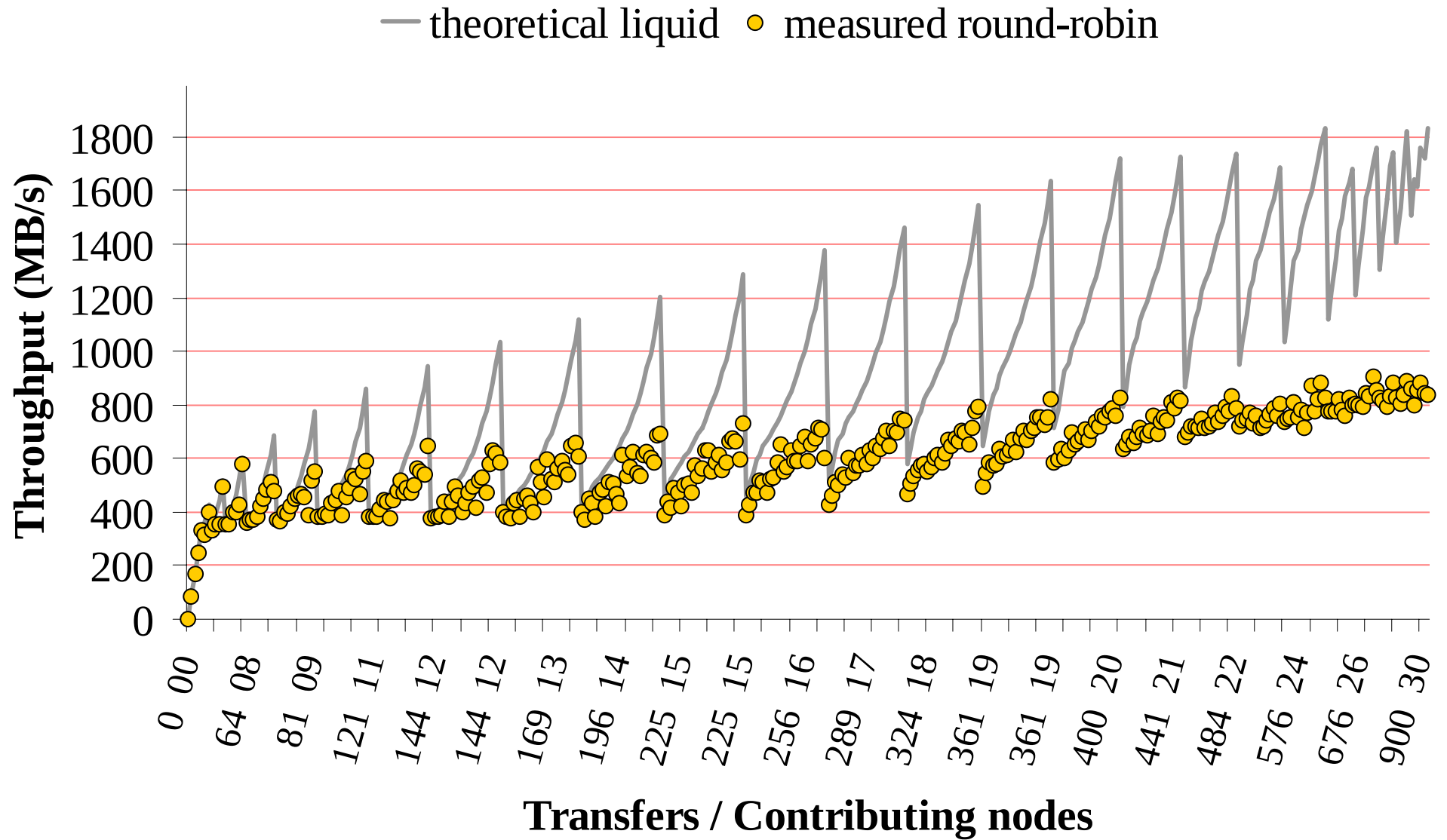
363 Communication Patterns



363 Topology Test-bed



Round-robin throughput



Team: a set of mutually non-congesting transfers using all bottlenecks

$$\begin{aligned}
 X = & \left\{ \begin{array}{l} \{l_1, l_6\}, \{l_1, l_7\}, \{l_1, l_8\}, \{l_1, \mathbf{l}_{12}, l_9\}, \{l_1, \mathbf{l}_{12}, l_{10}\}, \\ \{l_2, l_6\}, \{l_2, l_7\}, \{l_2, l_8\}, \{l_2, \mathbf{l}_{12}, l_9\}, \{l_2, \mathbf{l}_{12}, l_{10}\}, \\ \{l_3, l_6\}, \{l_3, l_7\}, \{l_3, l_8\}, \{l_3, \mathbf{l}_{12}, l_9\}, \{l_3, \mathbf{l}_{12}, l_{10}\}, \\ \{l_4, \mathbf{l}_{11}, l_6\}, \{l_4, \mathbf{l}_{11}, l_7\}, \{l_4, l_{11}, l_8\}, \{l_4, l_9\}, \{l_4, l_{10}\}, \\ \{l_5, \mathbf{l}_{11}, l_6\}, \{l_5, \mathbf{l}_{11}, l_7\}, \{l_5, l_{11}, l_8\}, \{l_5, l_9\}, \{l_5, l_{10}\} \end{array} \right\} \quad \text{schedule } \alpha \text{ is liquid} \Leftrightarrow \\
 \alpha = & \left\{ \begin{array}{l} \left\{ \begin{array}{l} \{l_1, \mathbf{l}_{12}, l_9\}, \\ \{l_2, l_7\}, \\ \{l_3, l_8\}, \\ \{l_4, \mathbf{l}_{11}, l_6\}, \\ \{l_5, l_{10}\} \end{array} \right\}, \left\{ \begin{array}{l} \{l_1, \mathbf{l}_{12}, l_{10}\}, \\ \{l_2, l_6\}, \\ \{l_4, \mathbf{l}_{11}, l_7\}, \\ \{l_5, l_9\} \end{array} \right\}, \left\{ \begin{array}{l} \{l_1, l_8\}, \\ \{l_2, \mathbf{l}_{12}, l_9\}, \\ \{l_3, l_6\}, \\ \{l_4, l_{10}\}, \\ \{l_5, \mathbf{l}_{11}, l_7\} \end{array} \right\}, \\ \\ \left\{ \begin{array}{l} \{l_1, l_7\}, \\ \{l_2, l_8\}, \\ \{l_3, \mathbf{l}_{12}, l_9\}, \\ \{l_5, \mathbf{l}_{11}, l_6\} \end{array} \right\}, \left\{ \begin{array}{l} \{l_1, l_6\}, \\ \{l_2, \mathbf{l}_{12}, l_{10}\}, \\ \{l_3, l_7\}, \\ \{l_4, \mathbf{l}_{11}, l_8\} \end{array} \right\}, \left\{ \begin{array}{l} \{l_3, \mathbf{l}_{12}, l_{10}\}, \\ \{l_4, l_9\}, \\ \{l_5, \mathbf{l}_{11}, l_8\} \end{array} \right\} \end{array} \right\}
 \end{aligned}$$

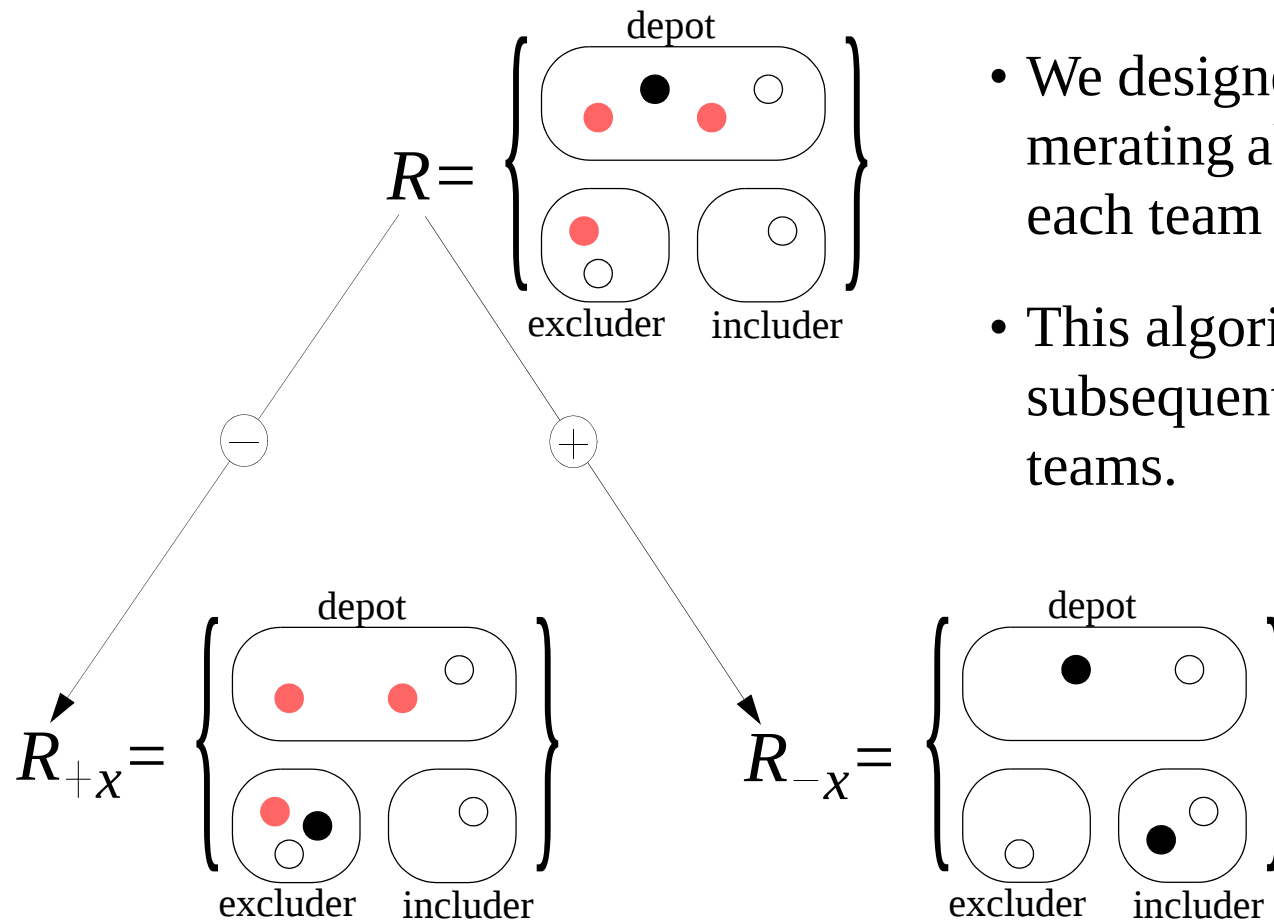
load of the bottlenecks
number of timeframes

$\Leftrightarrow \#(\alpha) = \Lambda(X) \Leftrightarrow$

$\Leftrightarrow \forall (A \in \alpha)$
 $A \text{ is a team of } X$

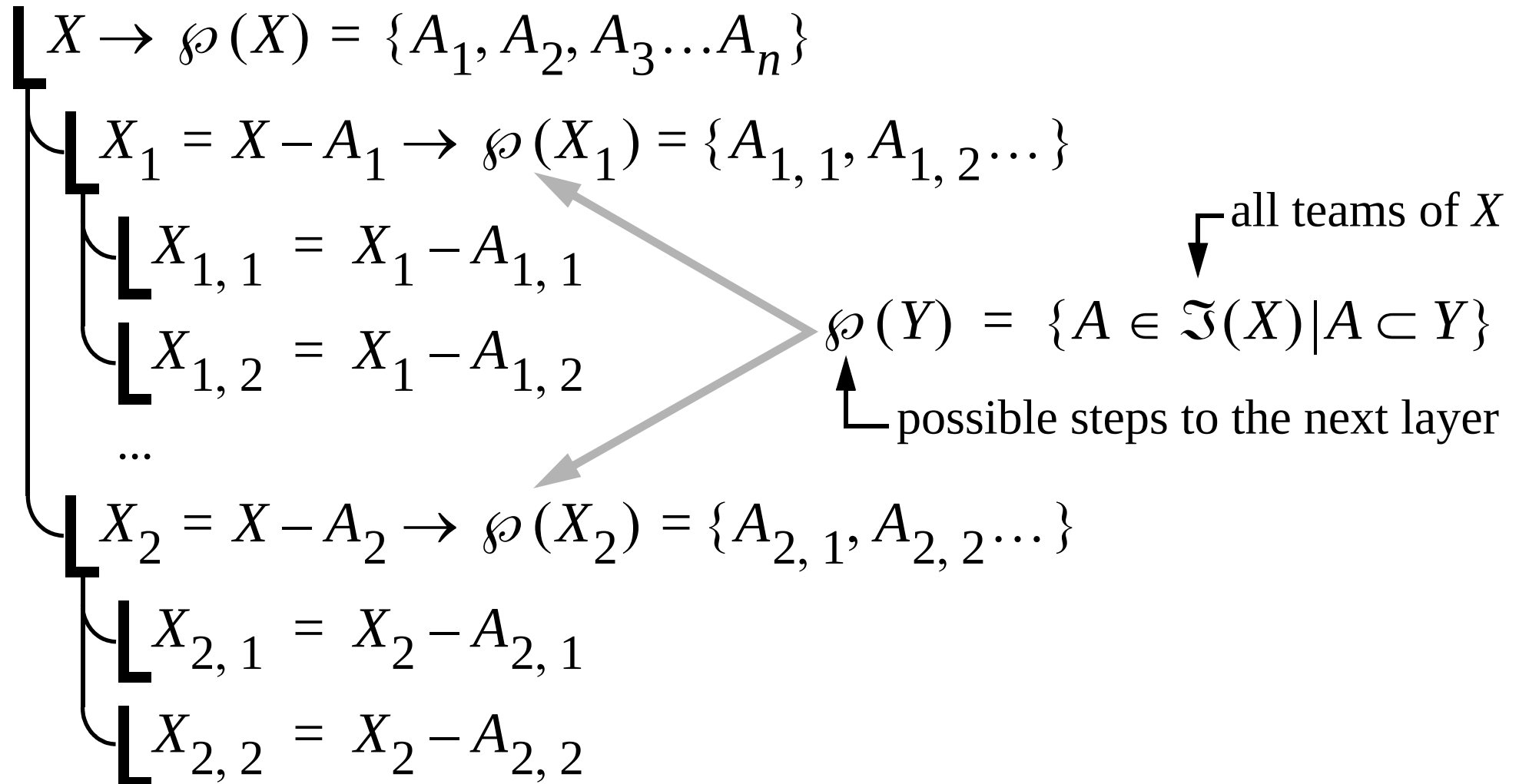
$\mathfrak{T}(X)$, all teams of the traffic X

- - transfer x
- - transfers congesting with x
- - transfers non-congesting with x

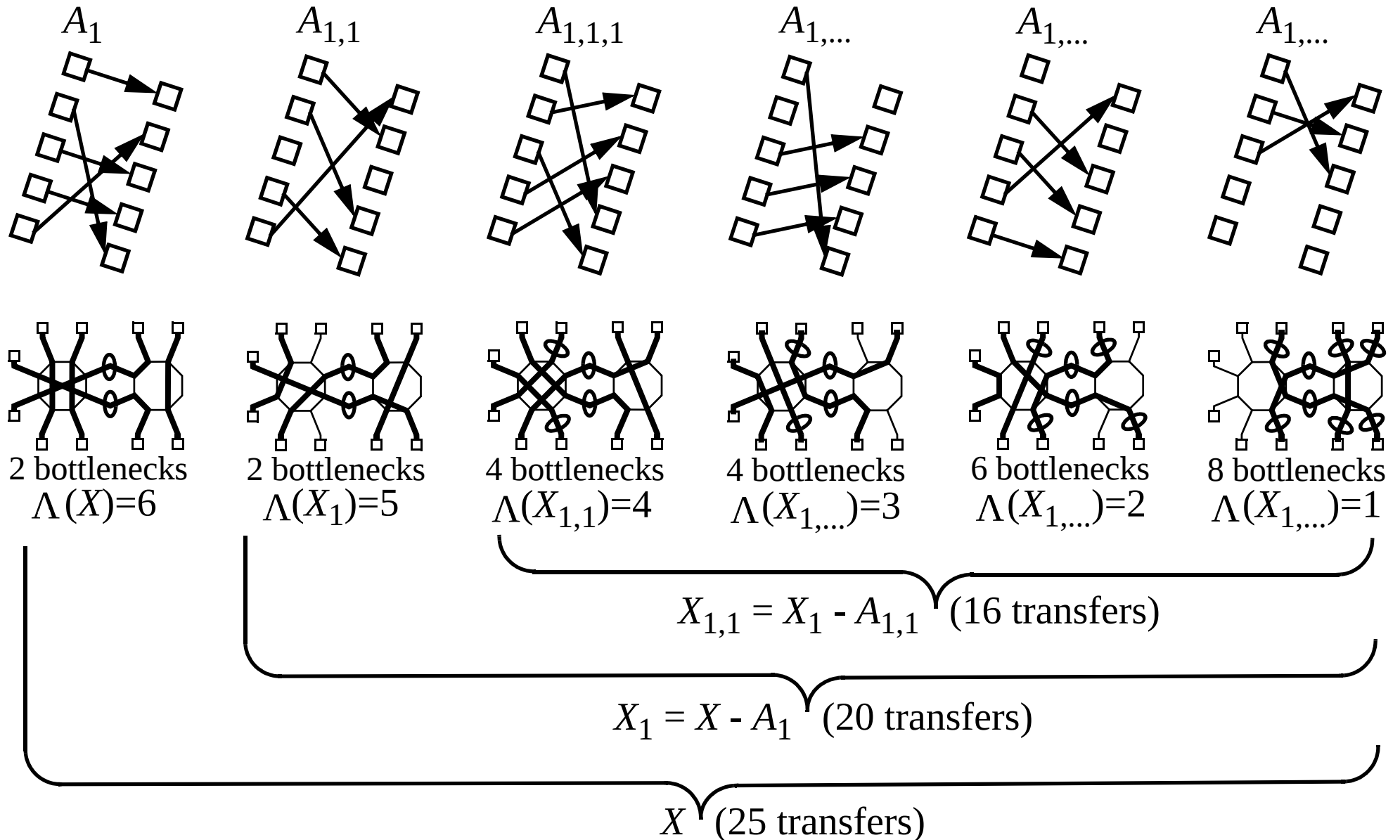


- To cover the full solution space when constructing a liquid schedule an efficient technique obtaining the whole set of possible teams of a traffic is required.
- We designed an efficient algorithm enumerating all teams of a traffic traversing each team once and only once.
- This algorithm obtains each team by subsequent partitioning of the set of all teams.
- We introduced triplets consisting of subsets of the traffic, representing one-by-one partitions of the set of all teams.

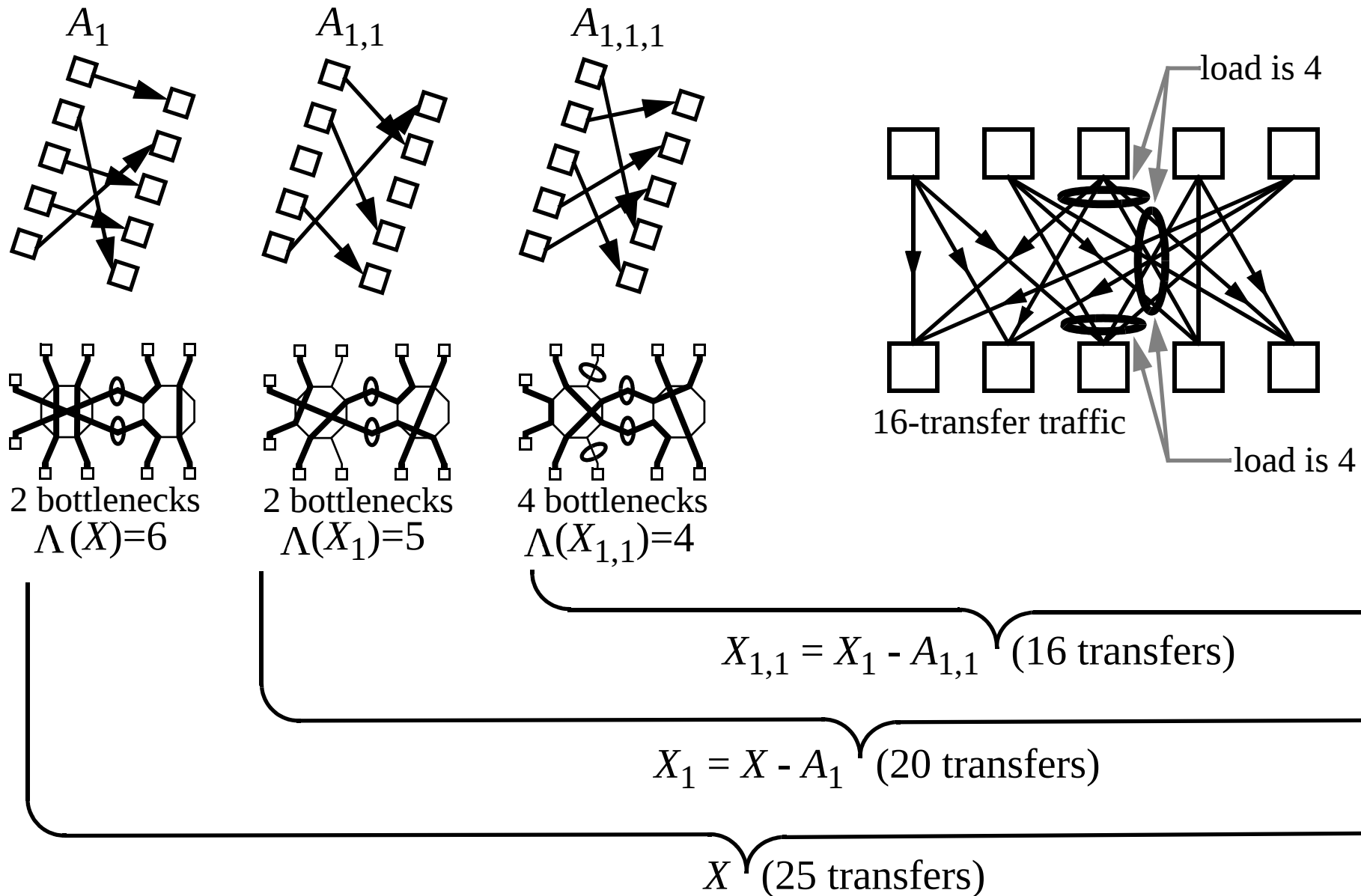
Liquid schedule search tree



Additional bottlenecks



Prediction of dead-ends



Liquid schedule search optimization

teams of the
reduced traffic

$$\mathfrak{T}(Y) \subset \{A \in \mathfrak{T}(X) | A \subset Y\}$$

original traffic's teams formed
from the reduced traffic

$$\begin{aligned}
 & \mathbf{L} \quad X \rightarrow \wp(X) = \{A_1, A_2, A_3 \dots A_n\} \\
 & \quad \mathbf{L} \quad X_1 = X - A_1 \rightarrow \wp(X_1) = \{A_{1,1}, A_{1,2} \dots\} \\
 & \quad \quad \mathbf{L} \quad X_{1,1} = X_1 - A_{1,1} \\
 & \quad \quad \mathbf{L} \quad X_{1,2} = X_1 - A_{1,2} \\
 & \quad \quad \dots \\
 & \quad \mathbf{L} \quad X_2 = X - A_2 \rightarrow \wp(X_2) = \{A_{2,1}, A_{2,2} \dots\}
 \end{aligned}$$

decreasing the search space without affecting the solution space

$$\wp(Y) = \{A \in \mathfrak{T}(X) | A \subset Y\} \rightarrow \wp(Y) = \mathfrak{T}(Y)$$

Liquid schedules construction

$$\underbrace{\mathfrak{S}^{full}(Y)}_{\text{full teams of the reduced traffic}} \subset \mathfrak{S}(Y)$$

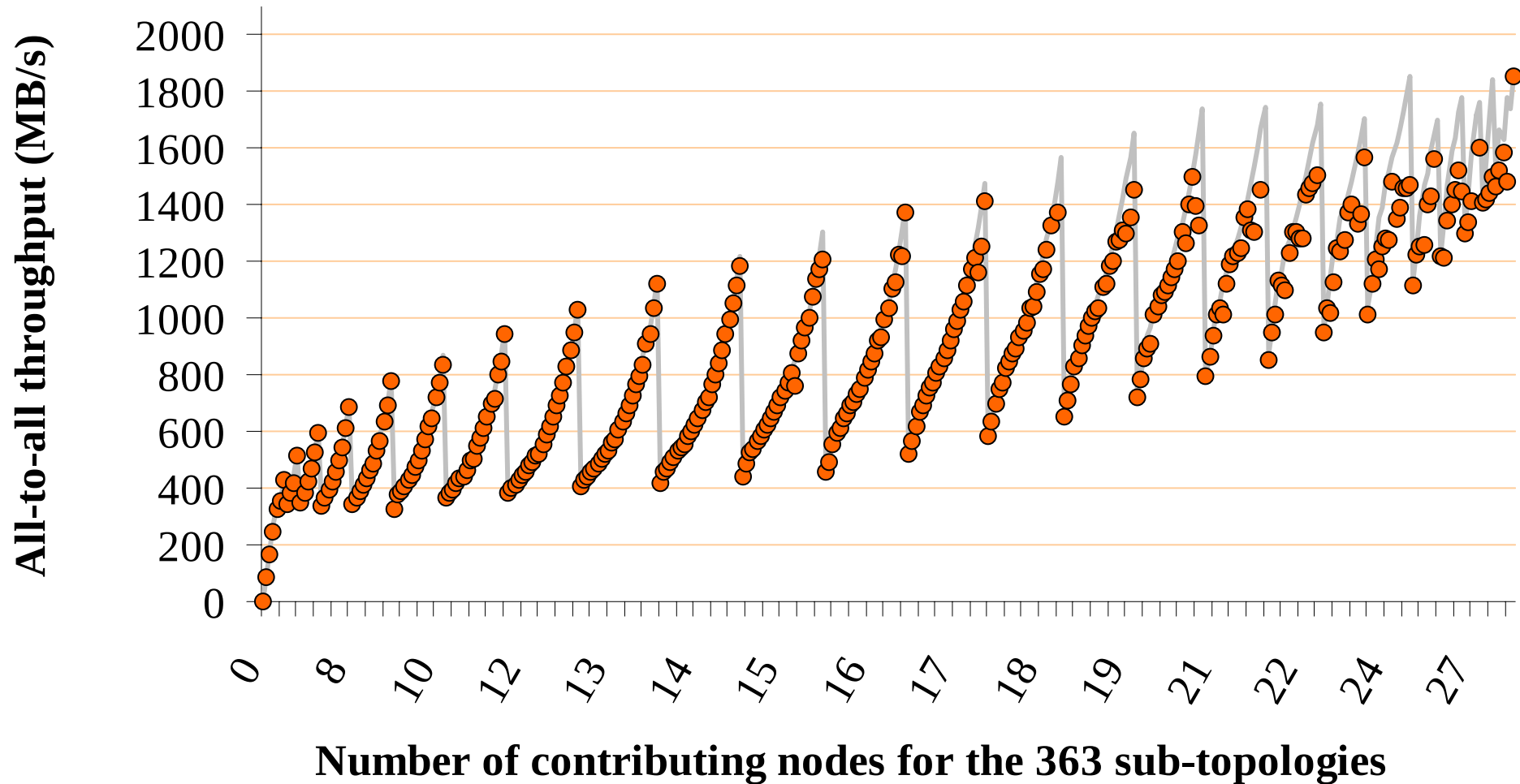
$$Choice = \wp(Y) = \mathfrak{S}(Y)$$

$$Choice = \downarrow \wp(Y) = \mathfrak{S}^{full}(Y)$$

additionally decreasing the search space without affecting the solution space

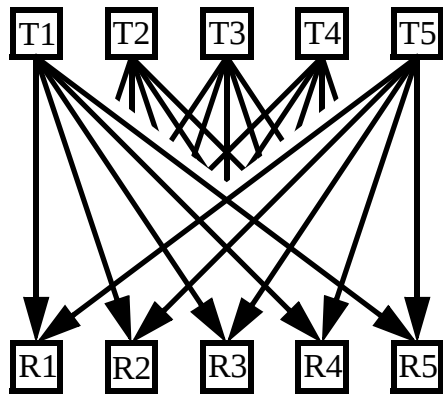
For more than 90% of the test-bed topologies construction of a global liquid schedule is completed in a fraction of a second (less than 0.1s).

Results

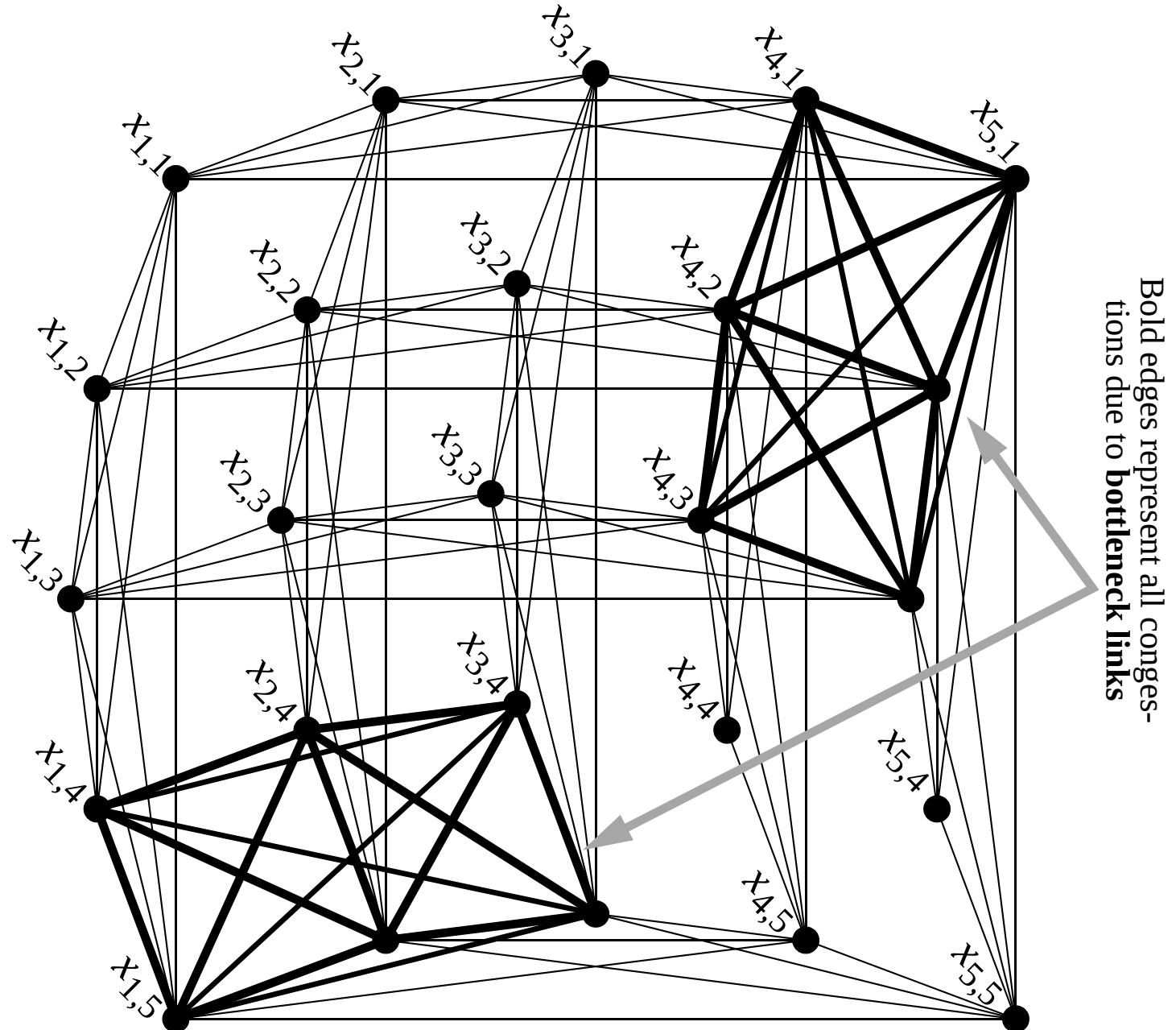
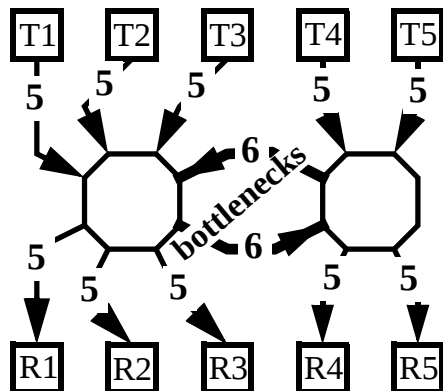


— liquid throughput ● carried out according to the liquid schedules

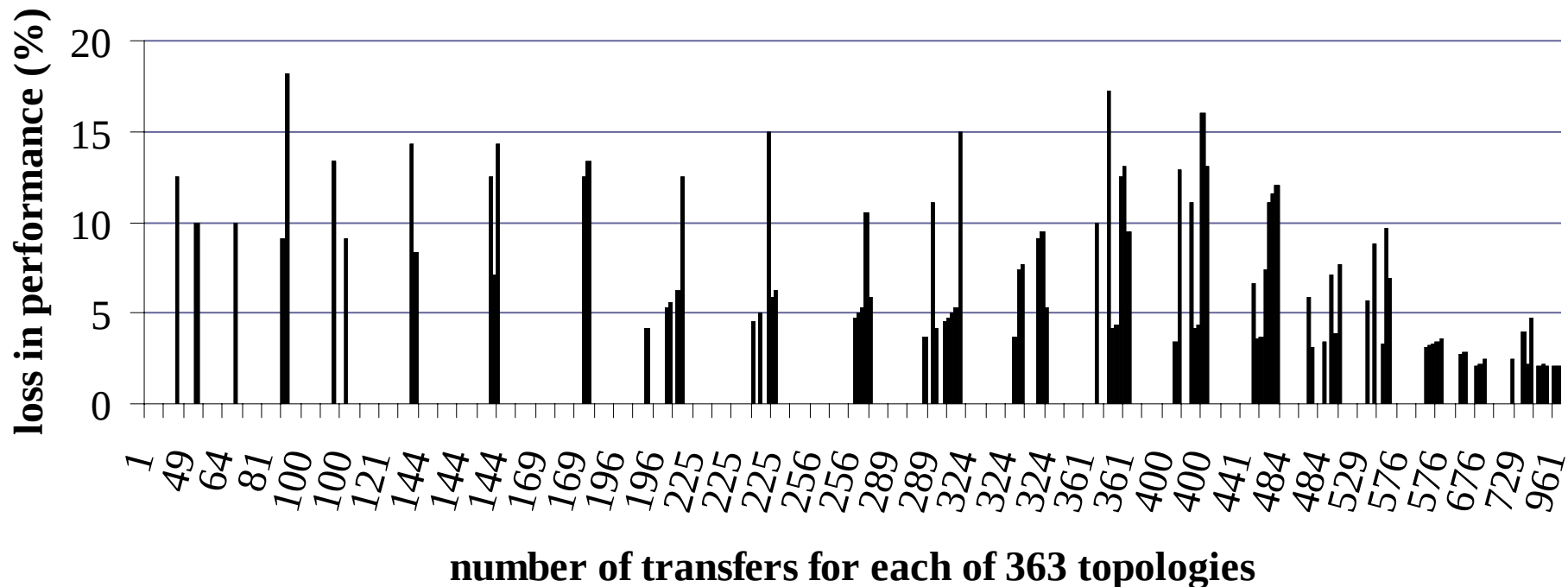
Congestion Graph



The 25 vertices of the graph represent the 25 transfers. The edges represent congestion relations between transfers, i.e. each edge represents one or more communication links shared by two transfers.



Loss of performance induced by schedules computed with a graph colouring heuristic algorithm



- For 74% of the topologies Dsatur algorithm does not induce a loss of performance.
- For 18% of topologies, the performance loss is bellow 10%.
- For 8% of topologies, the loss of performance is between 10% and 20%.

Conclusion

- Data exchanges relying on the liquid schedules may be carried out several times faster compared with topology-unaware schedules.
- Thanks to introduced theoretical model we considerably reduce the liquid schedule search space without affecting the solution space.
- Our method may be applied to applications requiring efficiency in concurrent continuous transmissions, such as video and voice traffic management, high energy physics data acquisition and reassembling.
- Liquid scheduling is applicable in wormhole, cut-through networks and can be useful in wavelength assignment problem in WDM optical networks.

Thank You!

Contact: *Emin.Gabrielyan@epfl.ch*