Virtual Network Embedding with Collocation

Benefits and Limitations of Pre-Clustering

Carlo Fürst[1] , Stefan Schmid[2], Anja Feldmann[1]

1: TU Berlin
2: TU Berlin & Telekom Innovation Laboratories

November 12, 2013

Today's Datacenters...

- Multi-tenant virtualized
- Tenants typically pay for host resources
- Connectivity is guaranteed

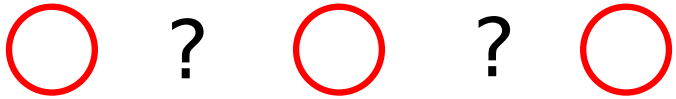
Today's Datacenters...

- Multi-tenant virtualized
- Tenants typically pay for host resources
- Connectivity is guaranteed

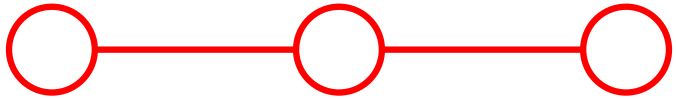
Problem [Ballani'11]:

Studies have shown that the intra-cloud bandwidth can vary by an order of magnitude.
$\Rightarrow$ Unpredictable application performance

Remove the uncertainty



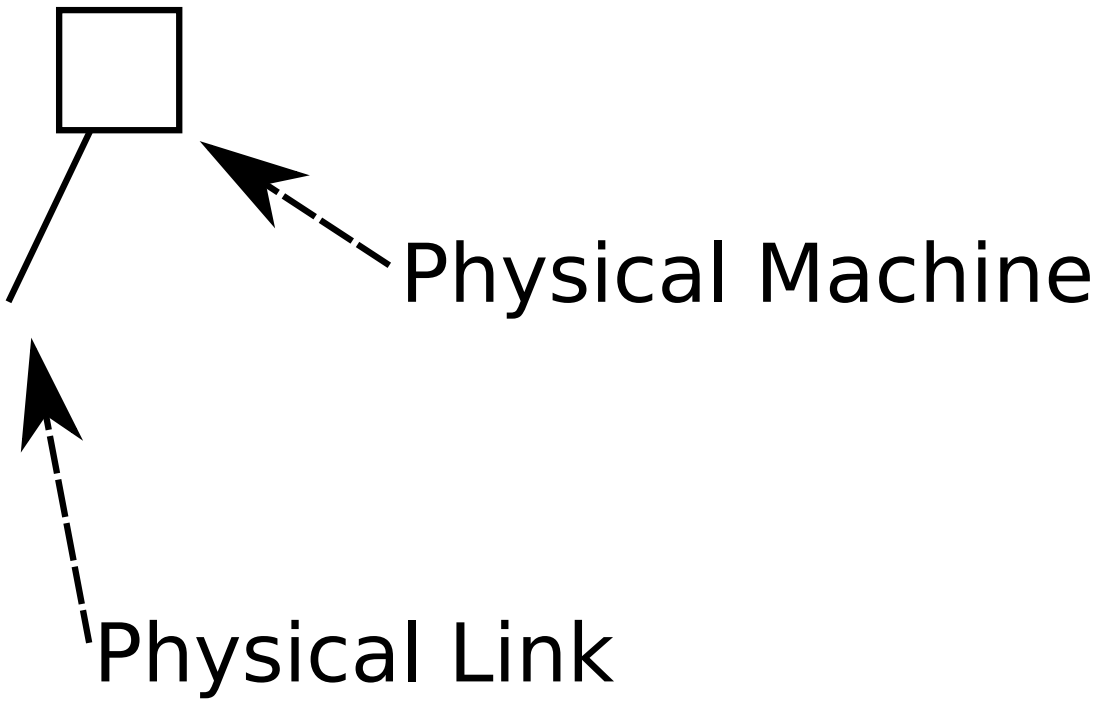
Remove the uncertainty



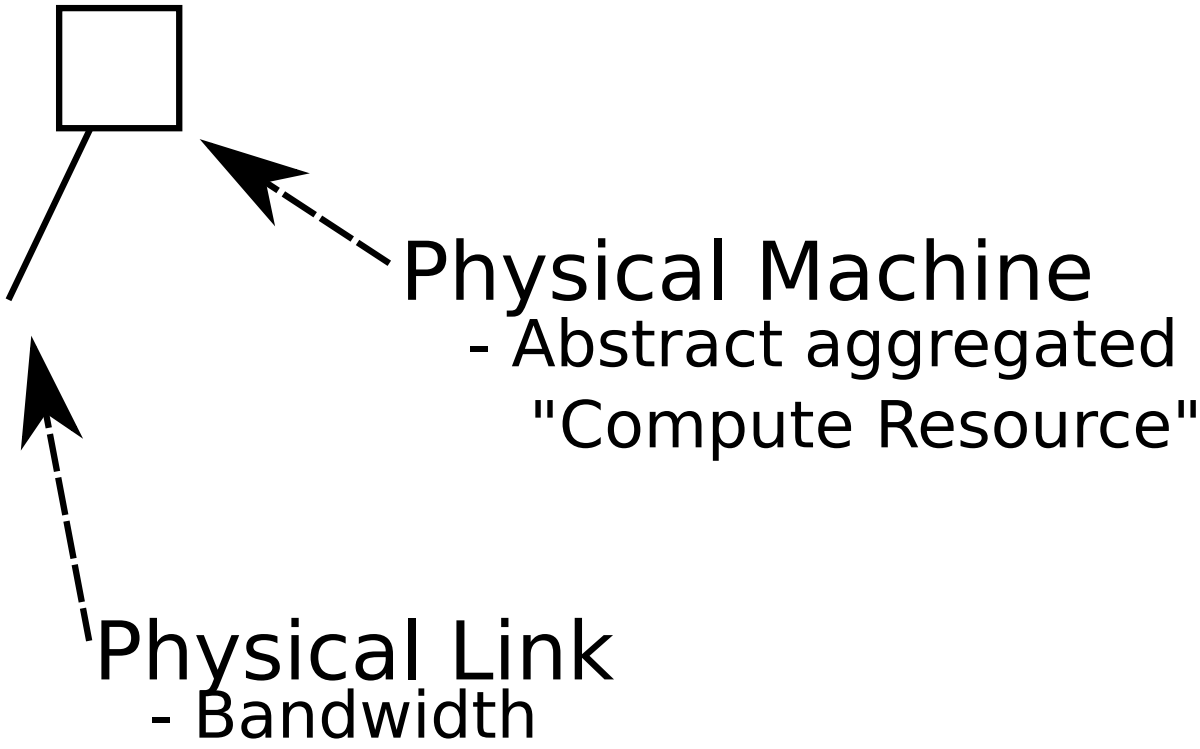
Outline

- Explain model and problem
- Identify the impact of the collocation option on embedding algorithms
- Introduce *Pre-Clustering* - a technique to enable any existing algorithm to generate collocated embeddings

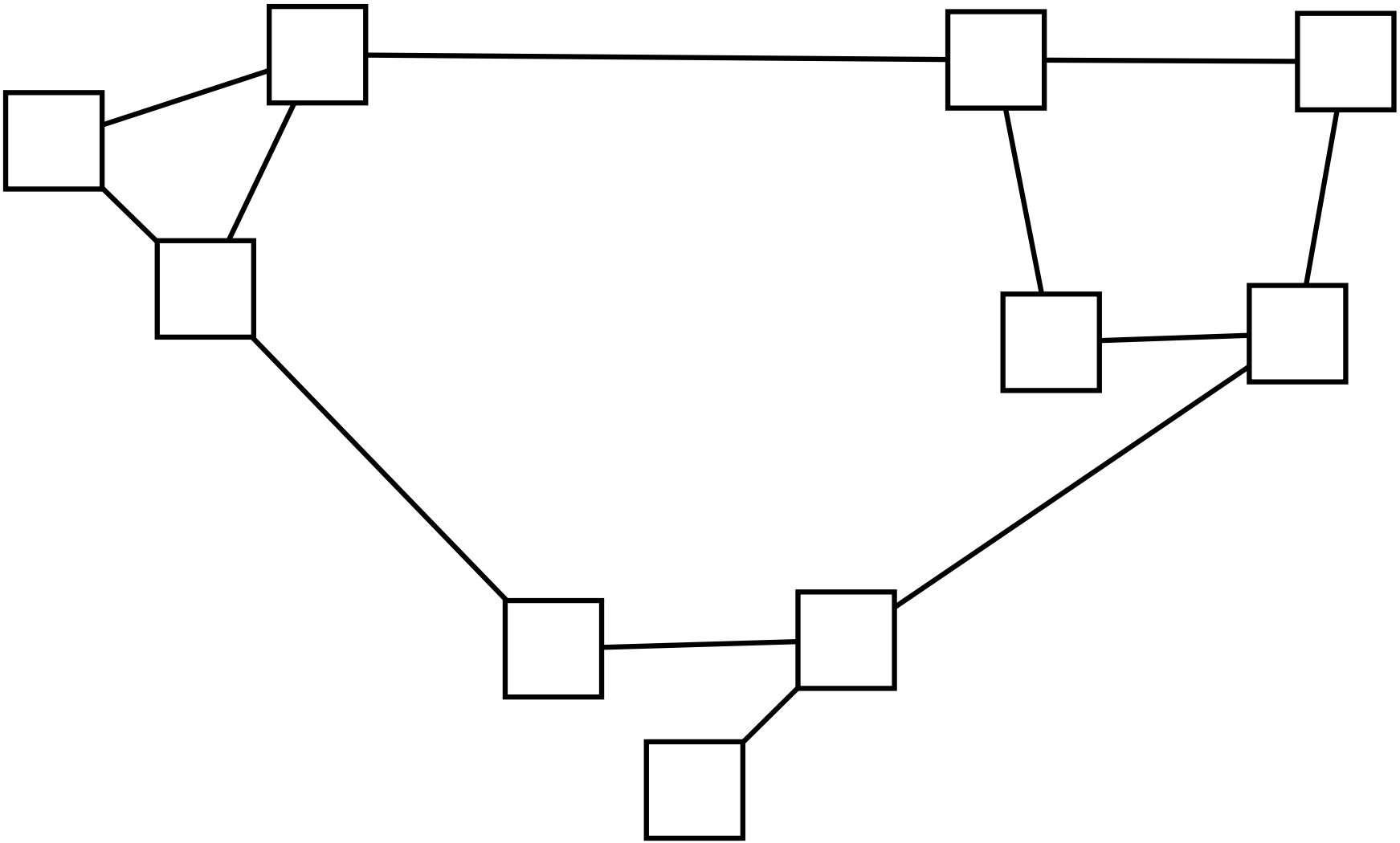
Virtual Network Embedding Problem



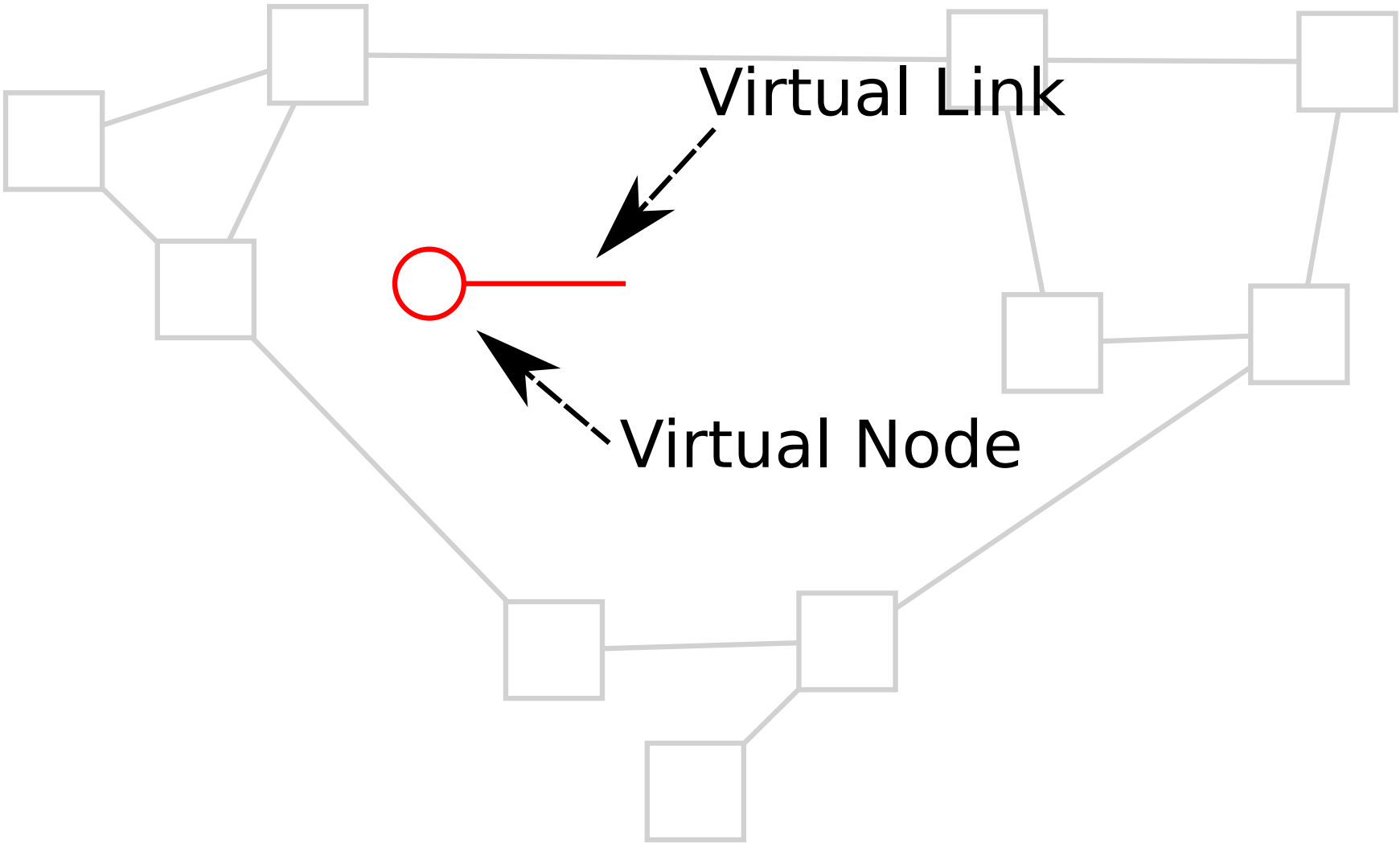
Virtual Network Embedding Problem



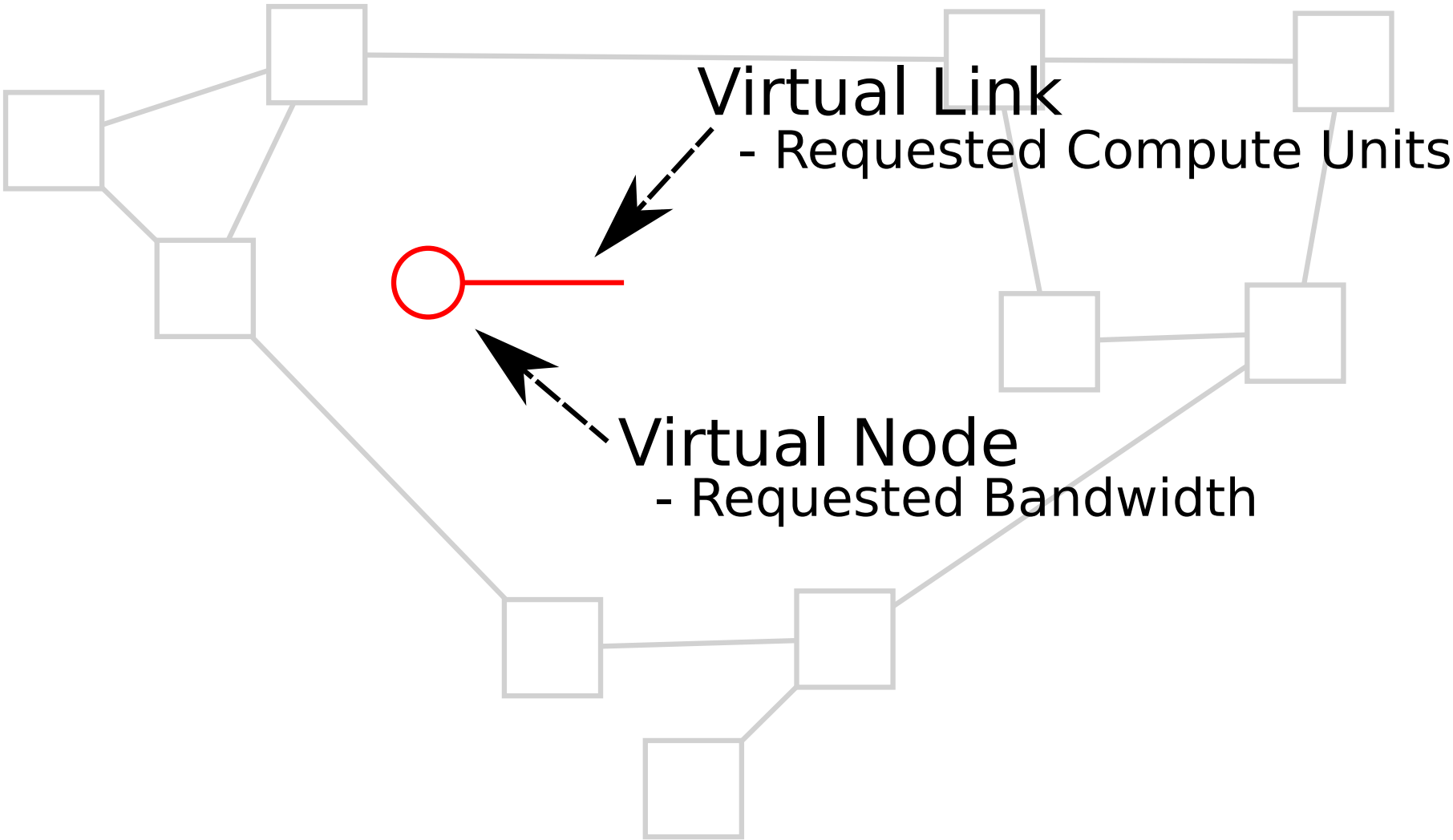
Virtual Network Embedding Problem



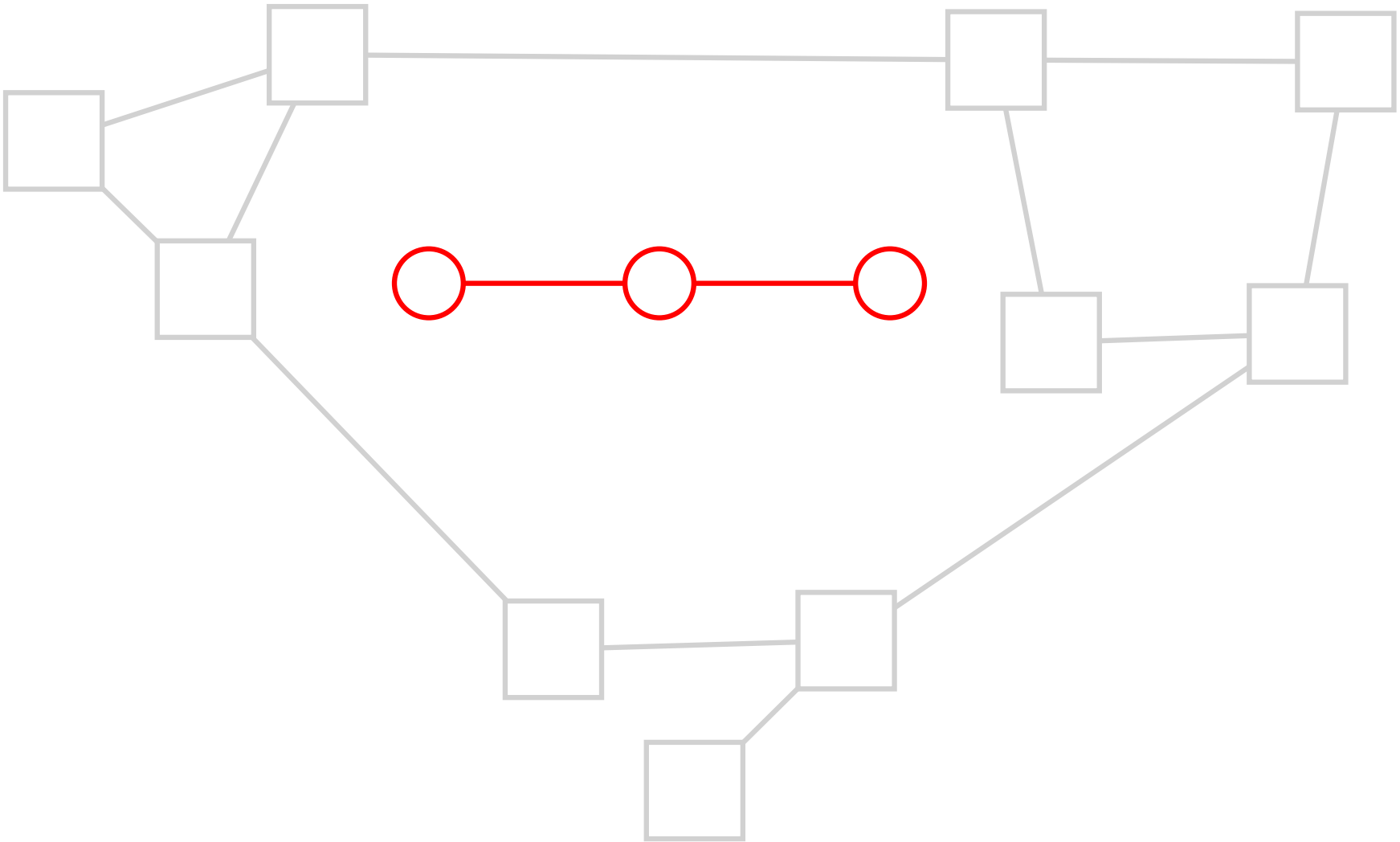
Virtual Network Embedding Problem



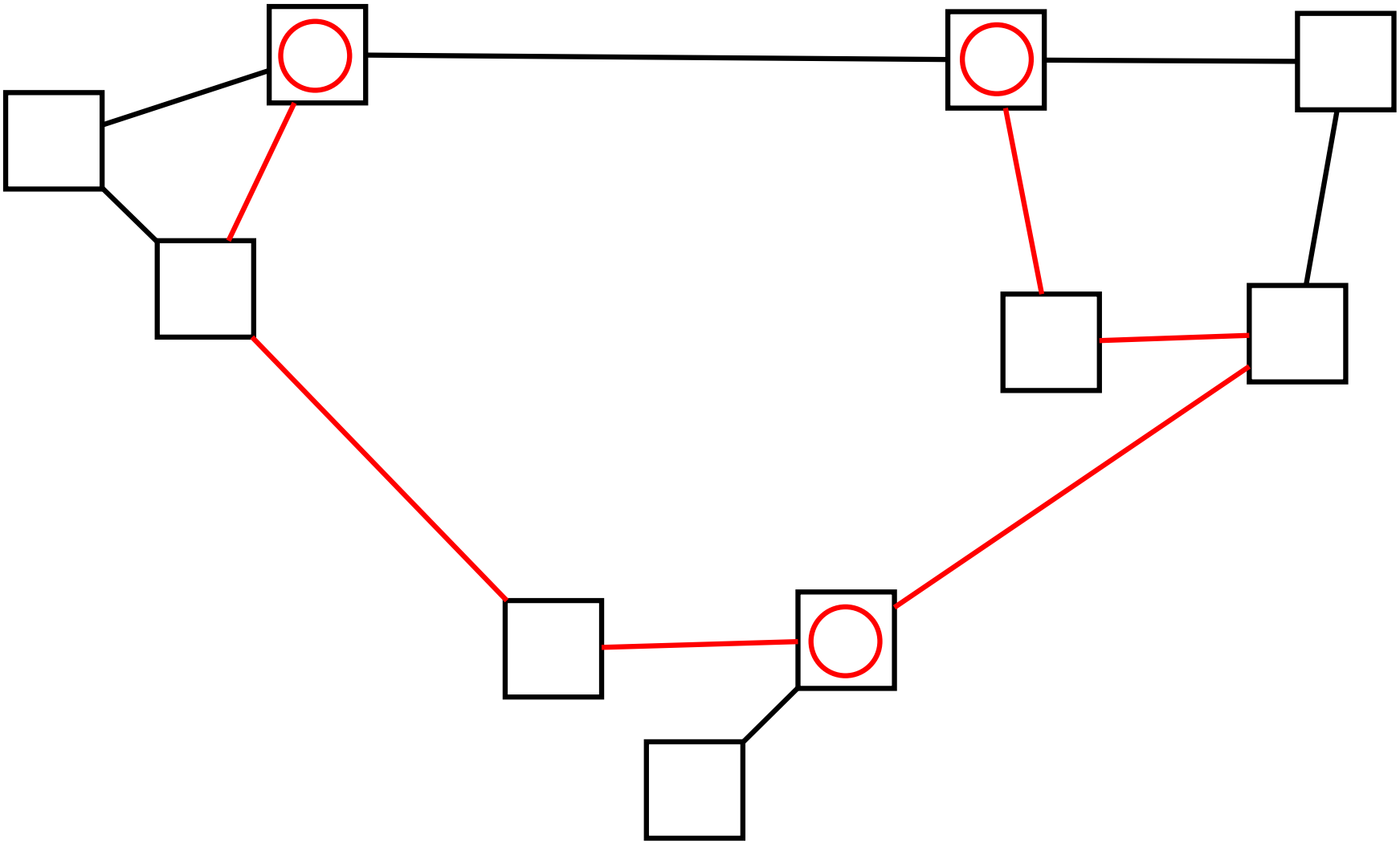
Virtual Network Embedding Problem



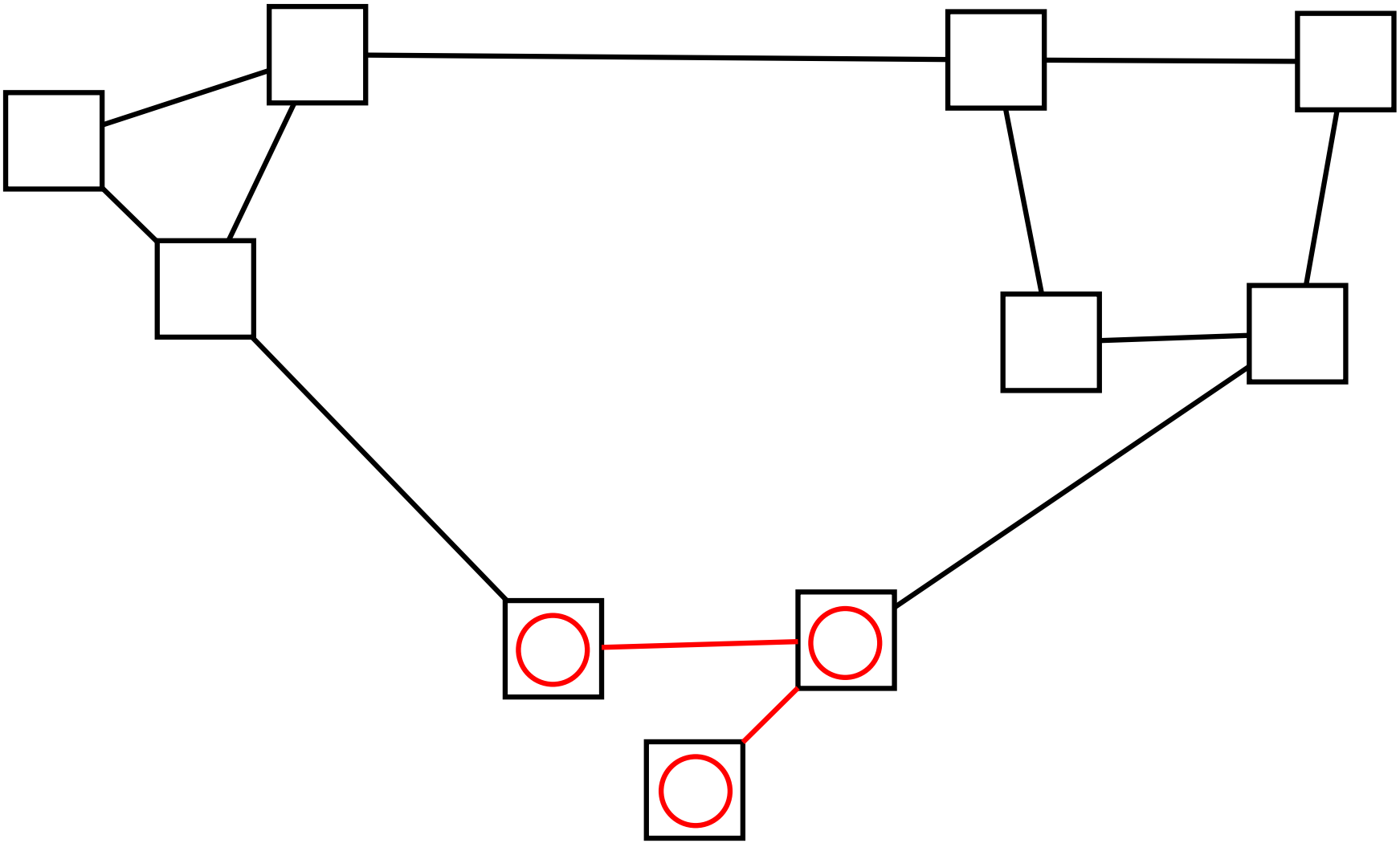
Virtual Network Embedding Problem



What is a ‘good’ mapping?



What is a ‘good’ mapping?

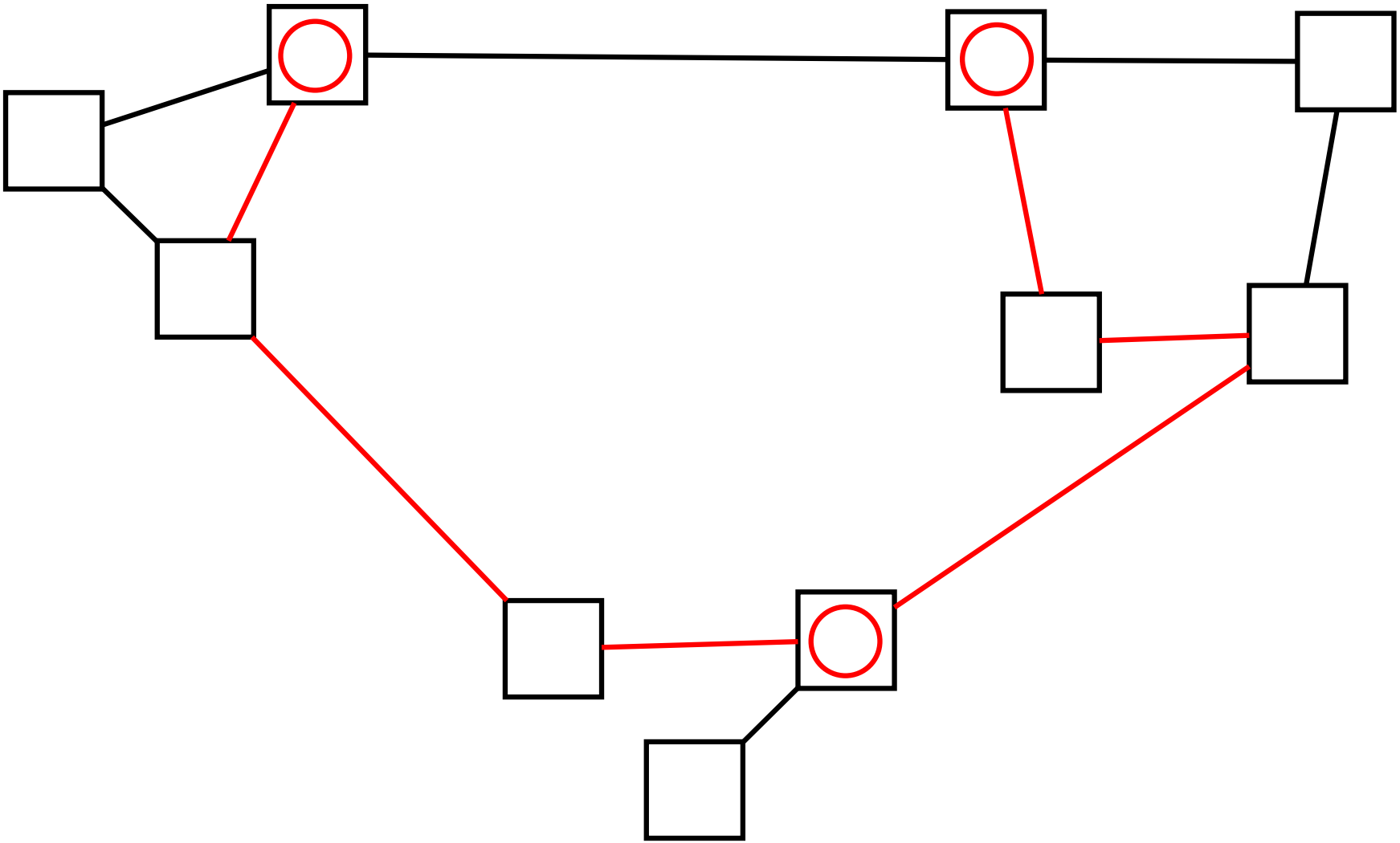


Existing Solutions

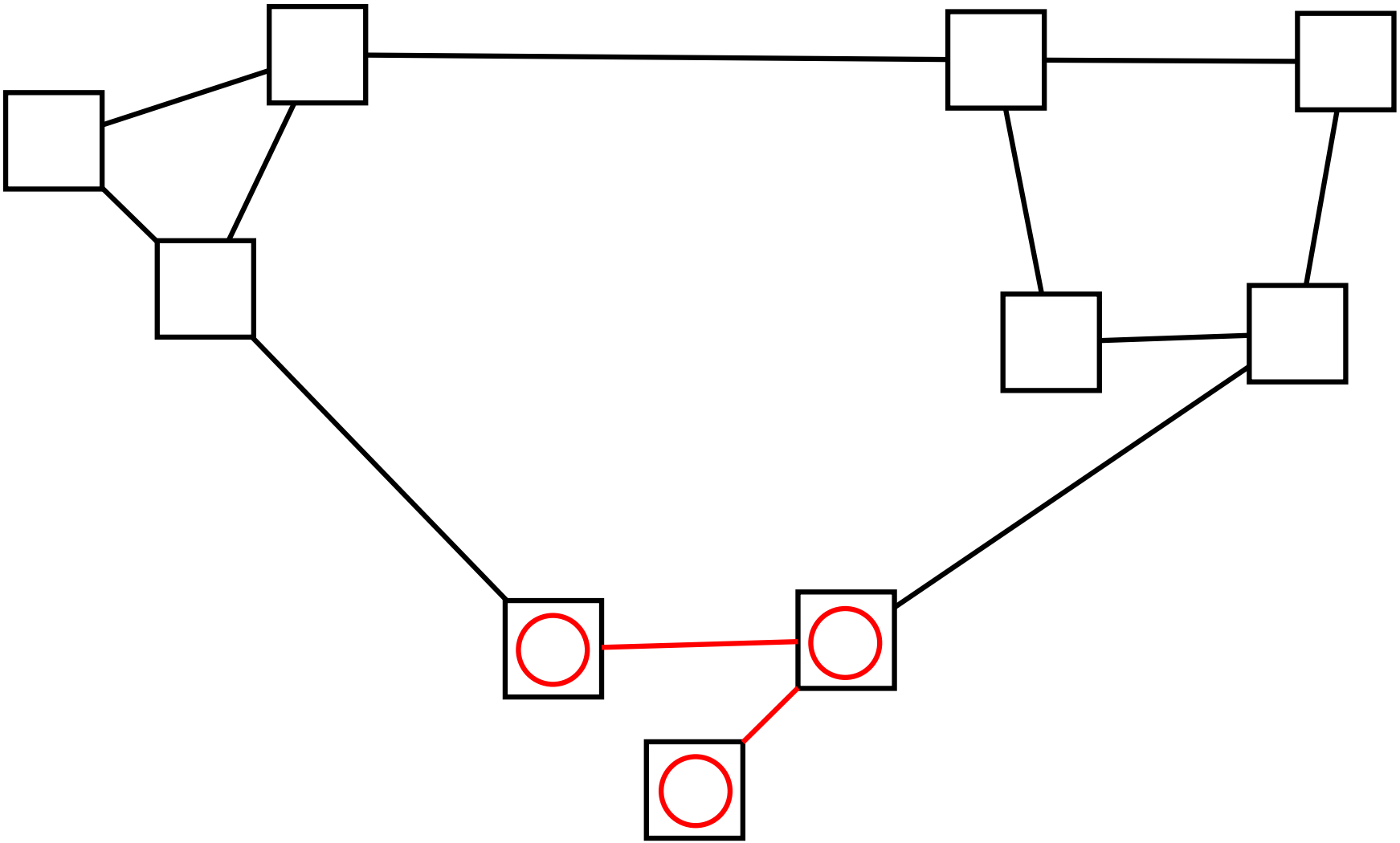
Many existing mapping algorithms

- ViNE [CHOWDHURY, Infocom 2009]
- SecondNet [GUO, Co-NEXT 2010]
- Oktopus [BALLANI, Sigcomm 2011]
- Isomorphism Detection [LISCHKA, Sigcomm 2009]
- Various Mixed-Integer-Programs
- ...

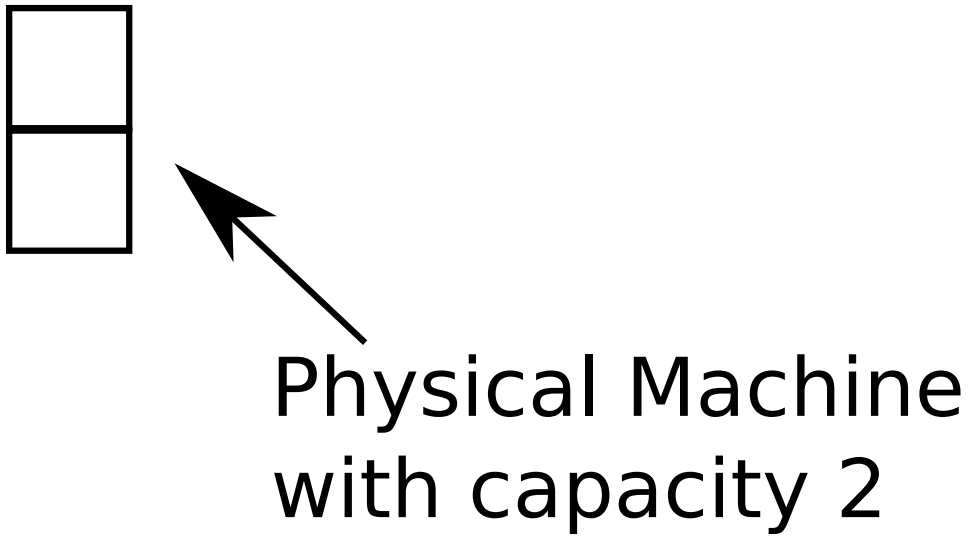
Existing Solutions



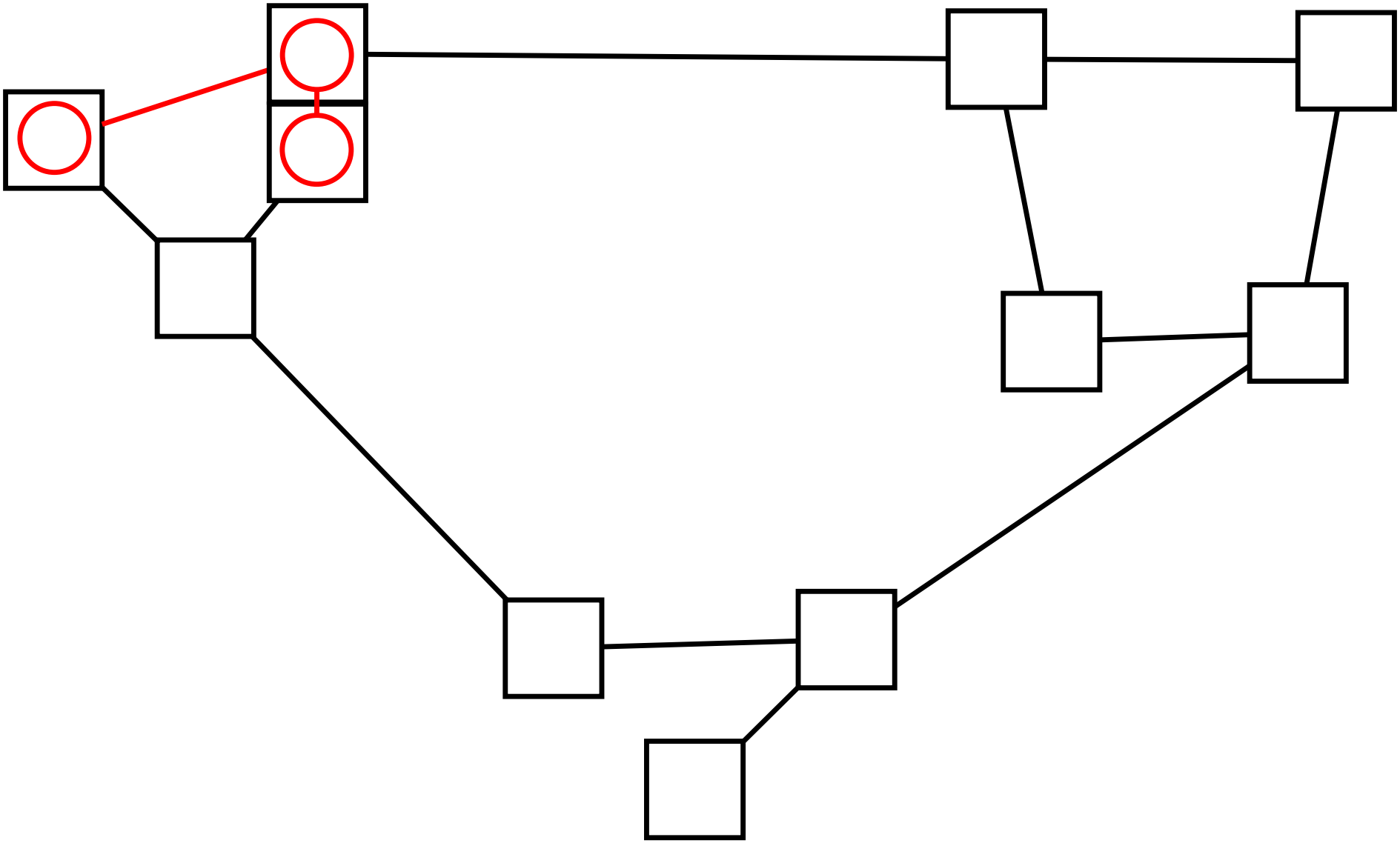
Existing Solutions



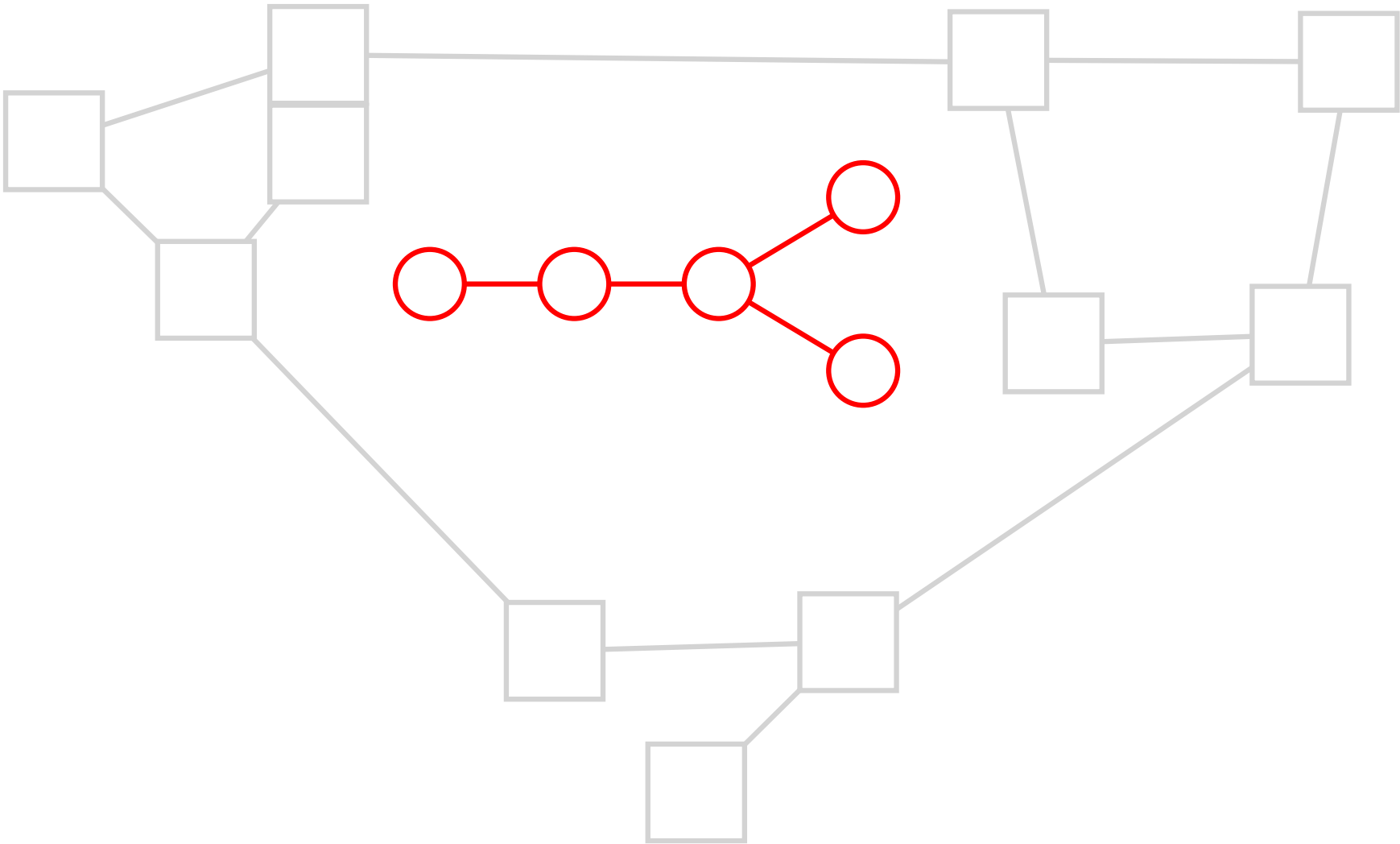
Collocated Mappings



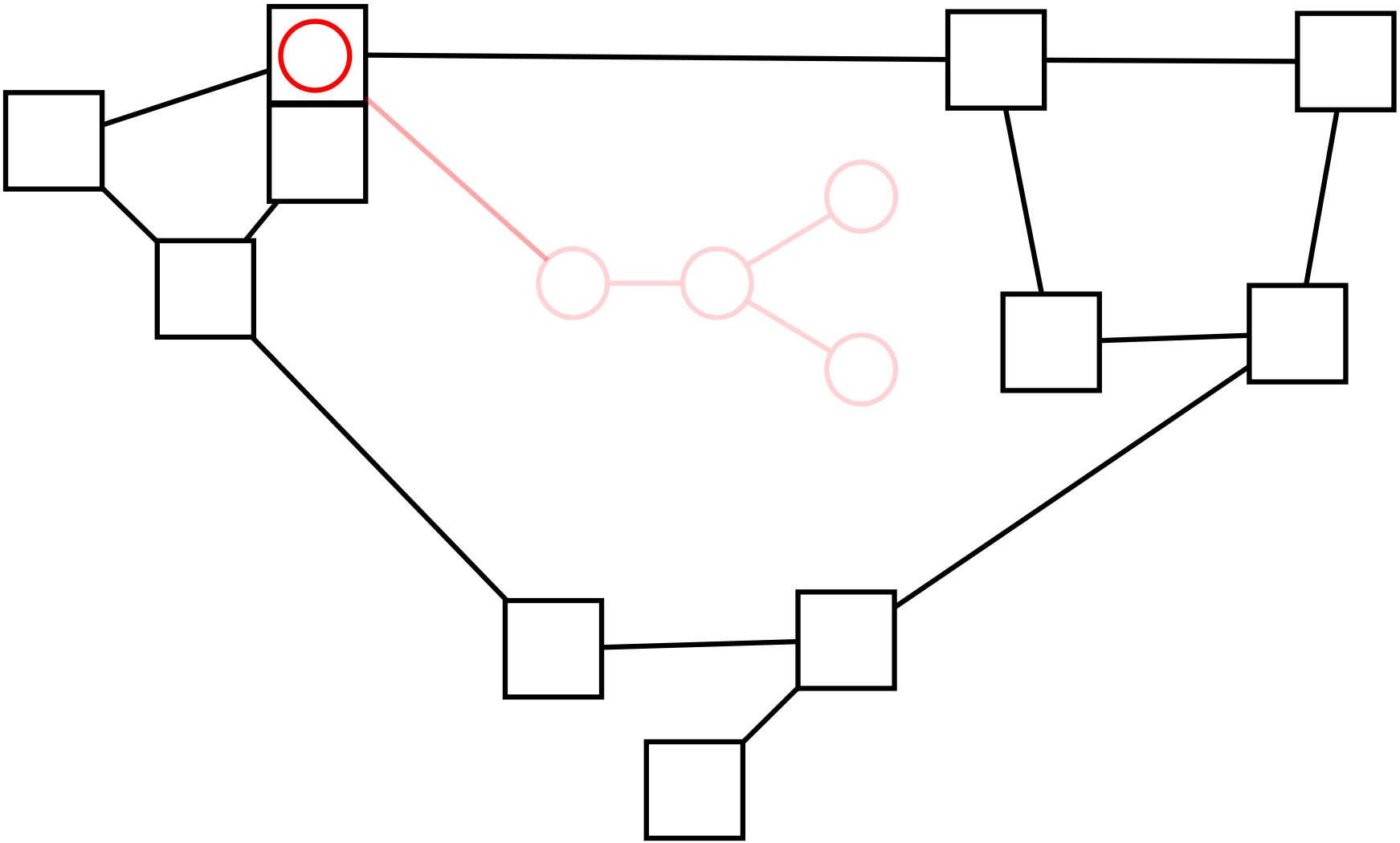
Collocated Mappings



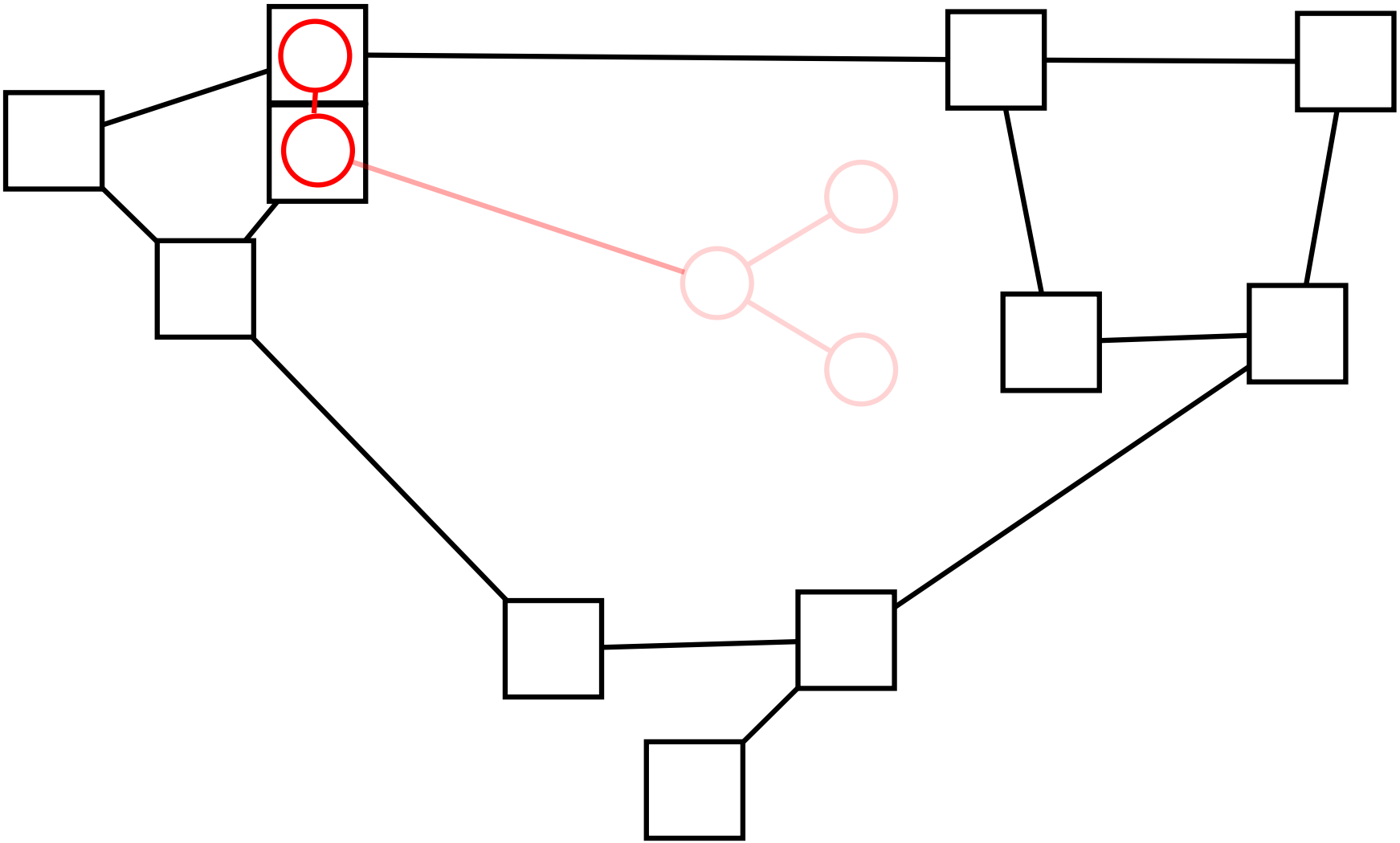
Benchmarking Algorithm: LoCo



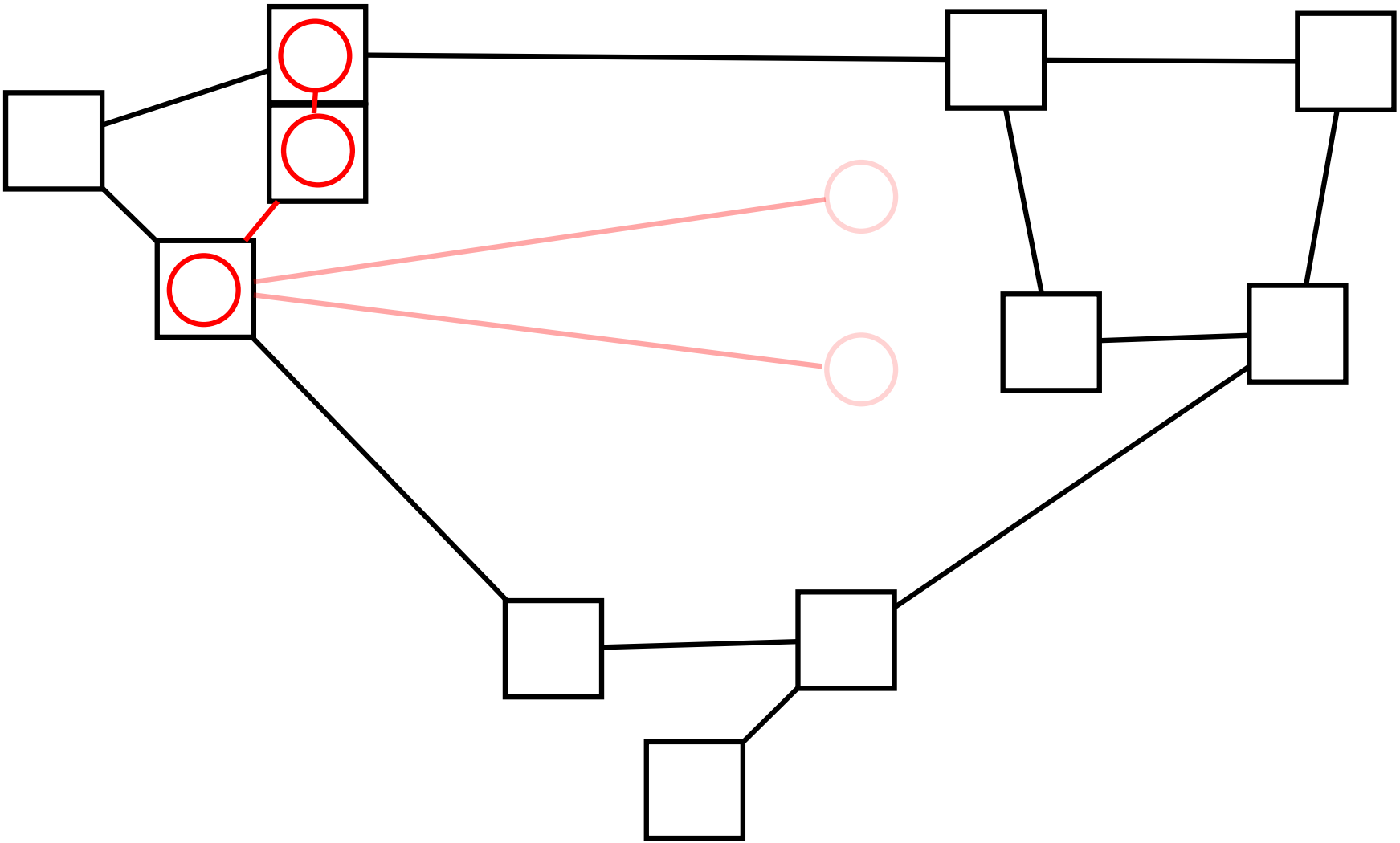
Benchmarking Algorithm: LoCo



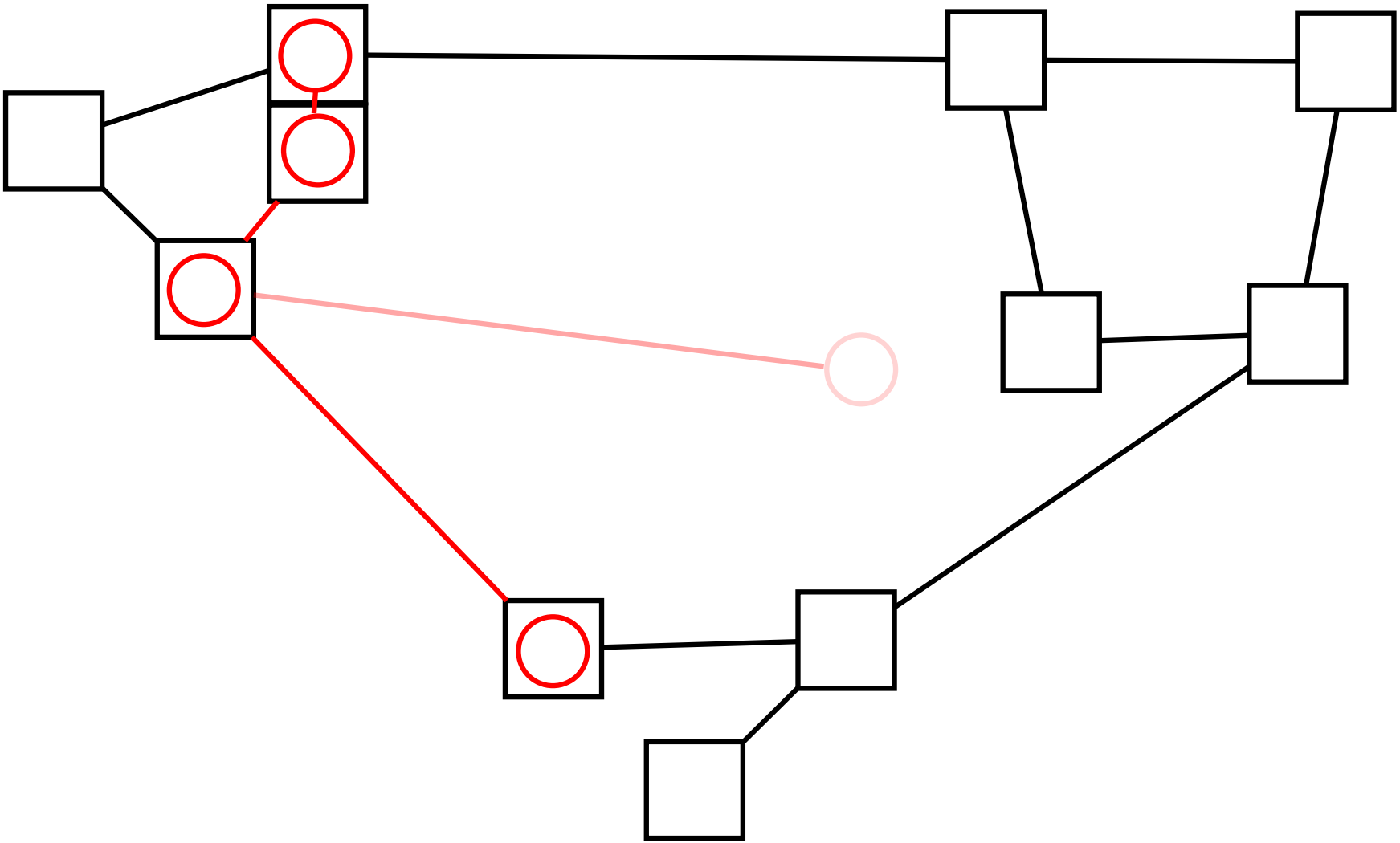
Benchmarking Algorithm: LoCo



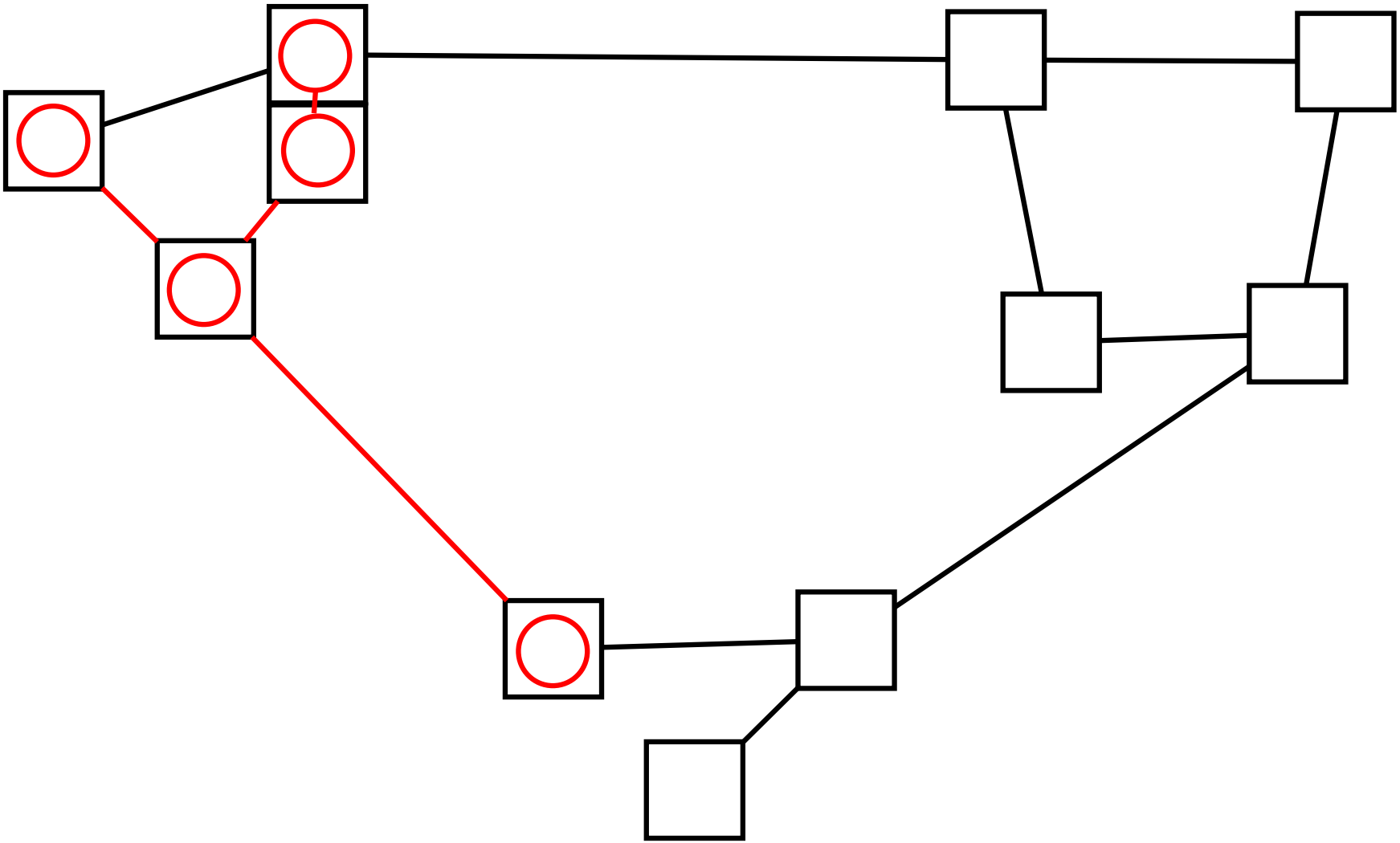
Benchmarking Algorithm: LoCo



Benchmarking Algorithm: LoCo



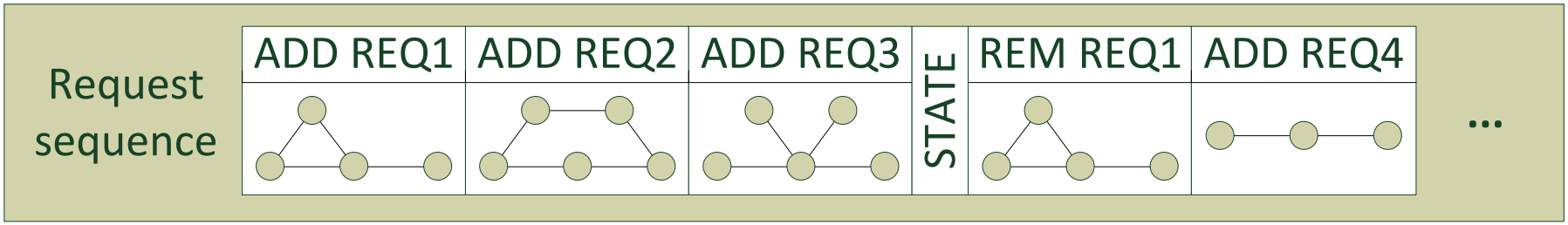
Benchmarking Algorithm: LoCo



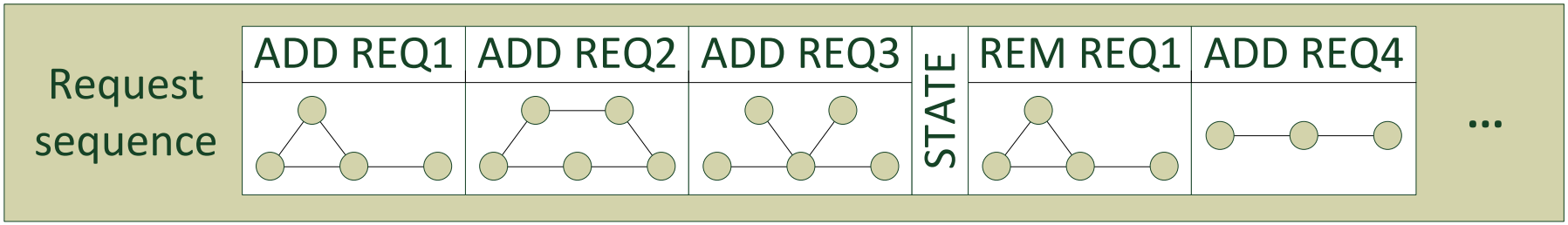
Benchmarking Algorithm: LoCo

- Backtrack on failure
- Backtrack only over possible start nodes
- Graph exploration is directed by node / link resource requests
- Avoid Backtracking by forward checking

Evaluation Setup

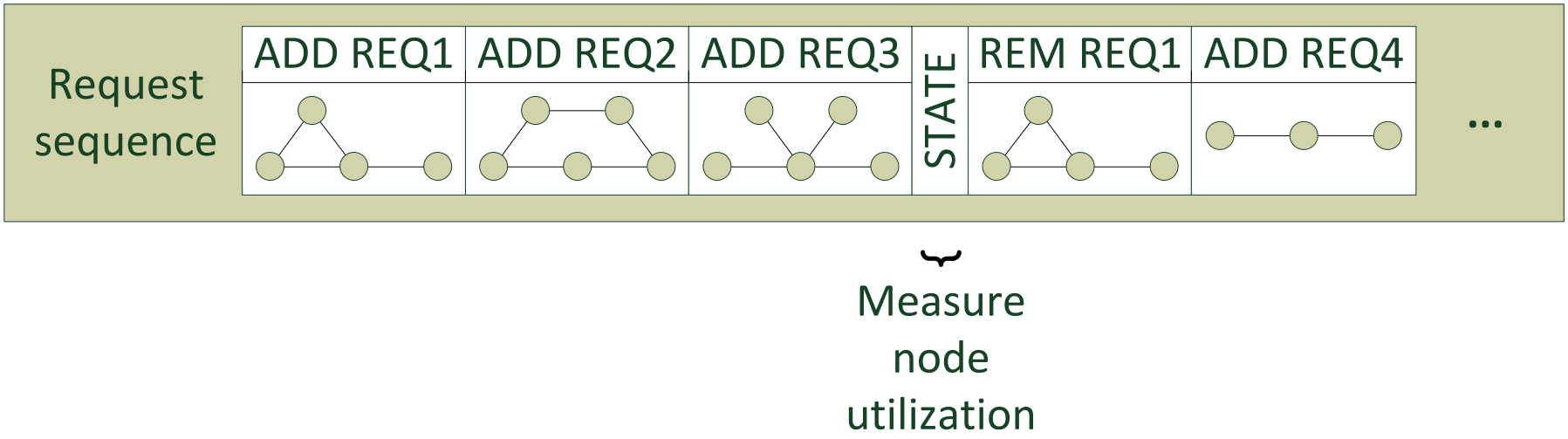


Evaluation Setup

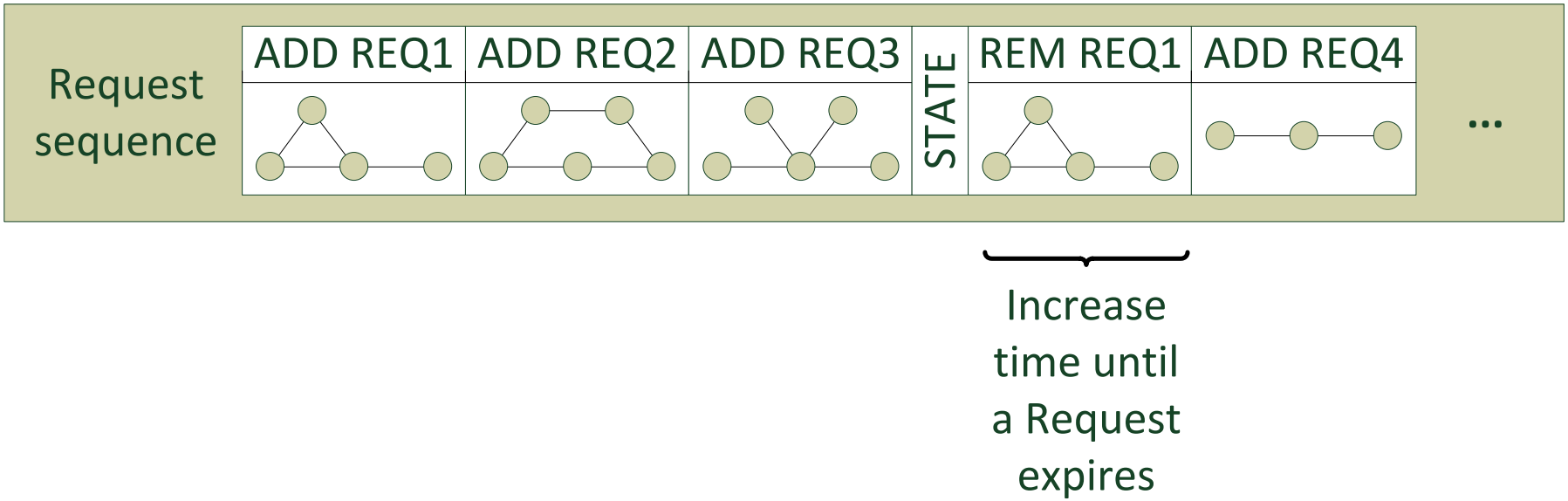


Add Requests
Until:
Sum of requested node resources = Sum of substrate node resources

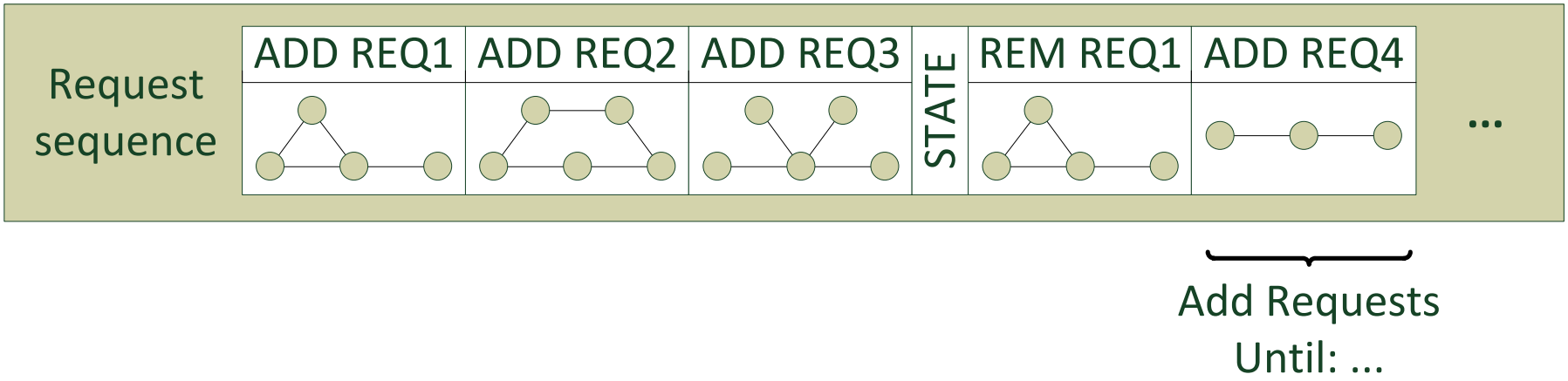
Evaluation Setup



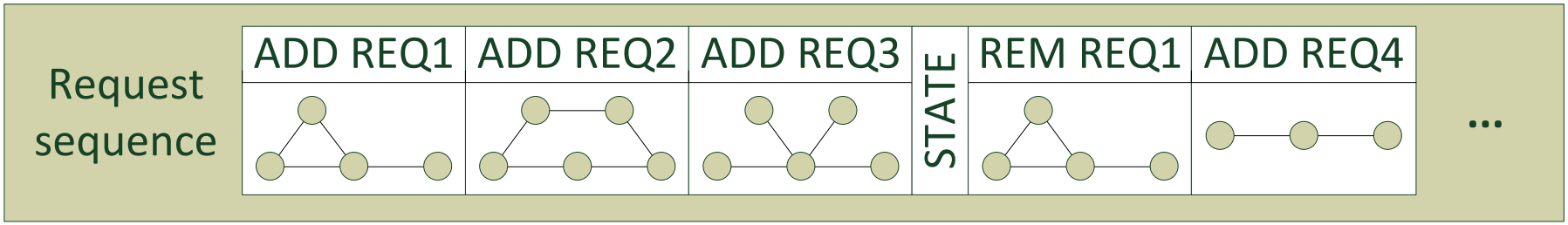
Evaluation Setup



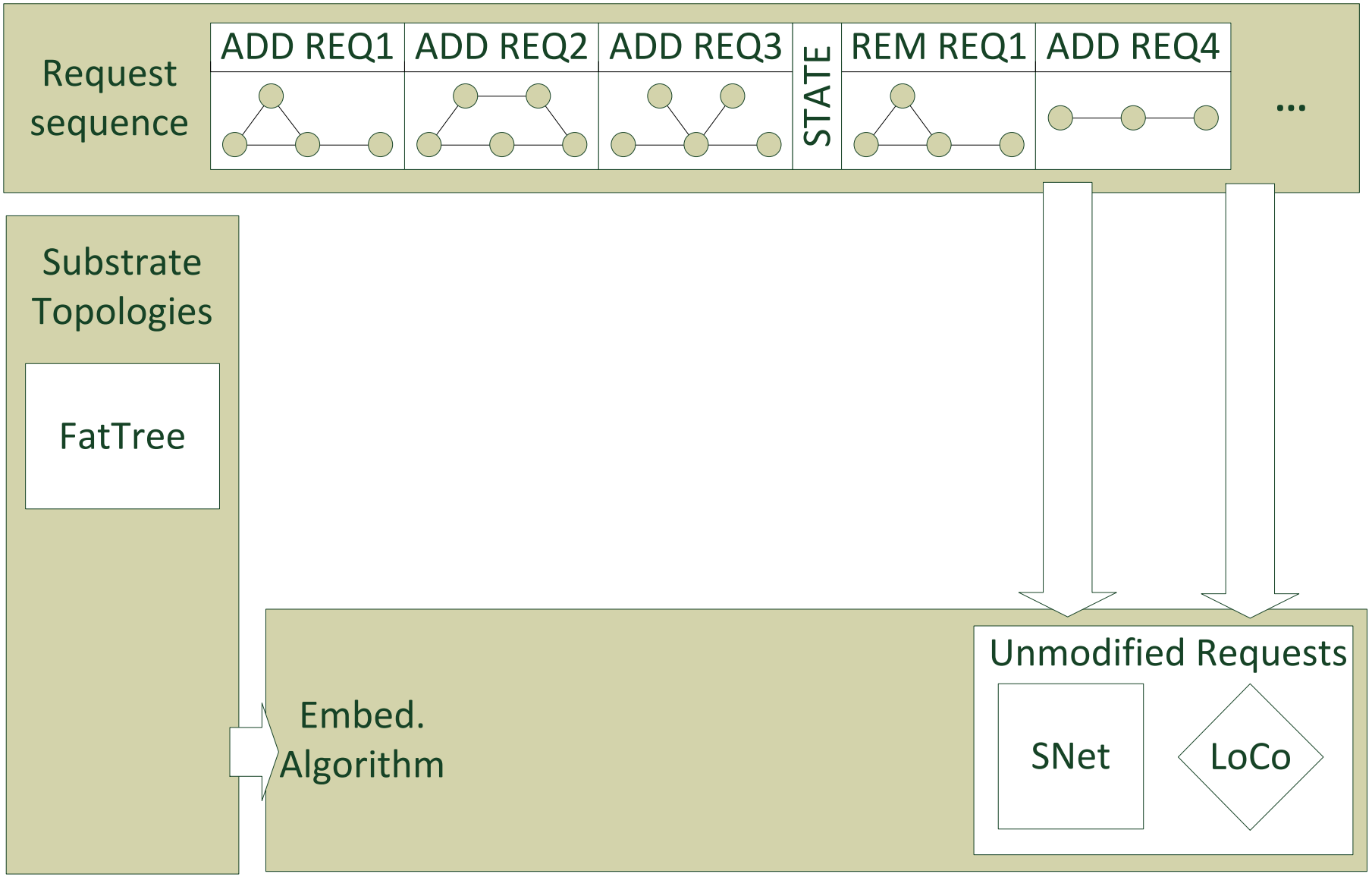
Evaluation Setup



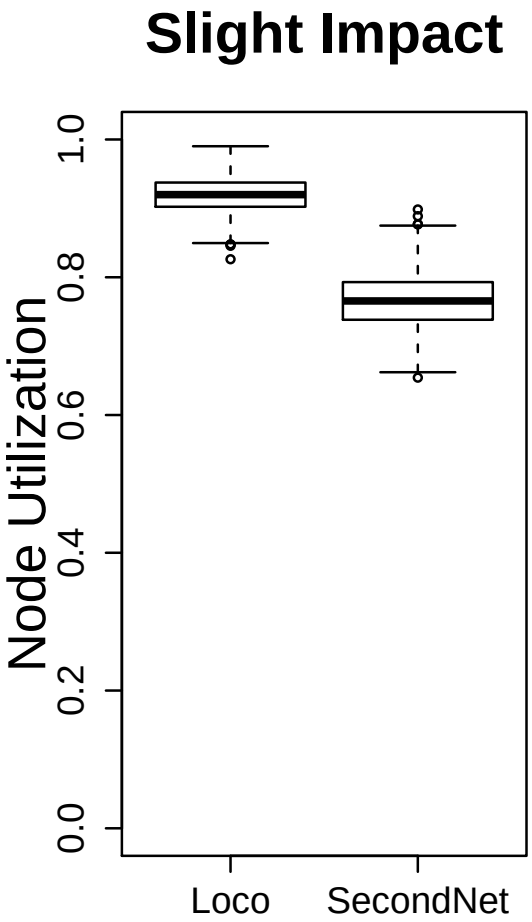
Evaluation Setup



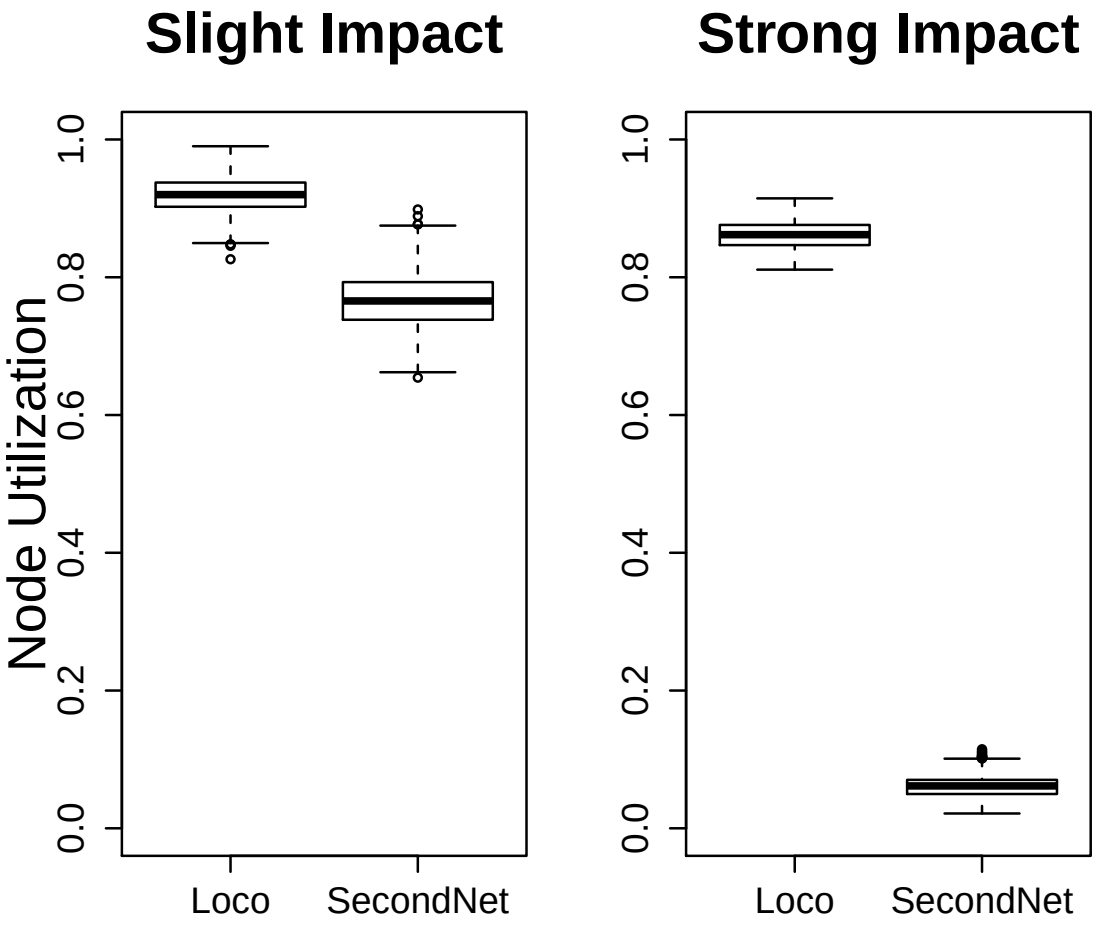
Evaluation Setup



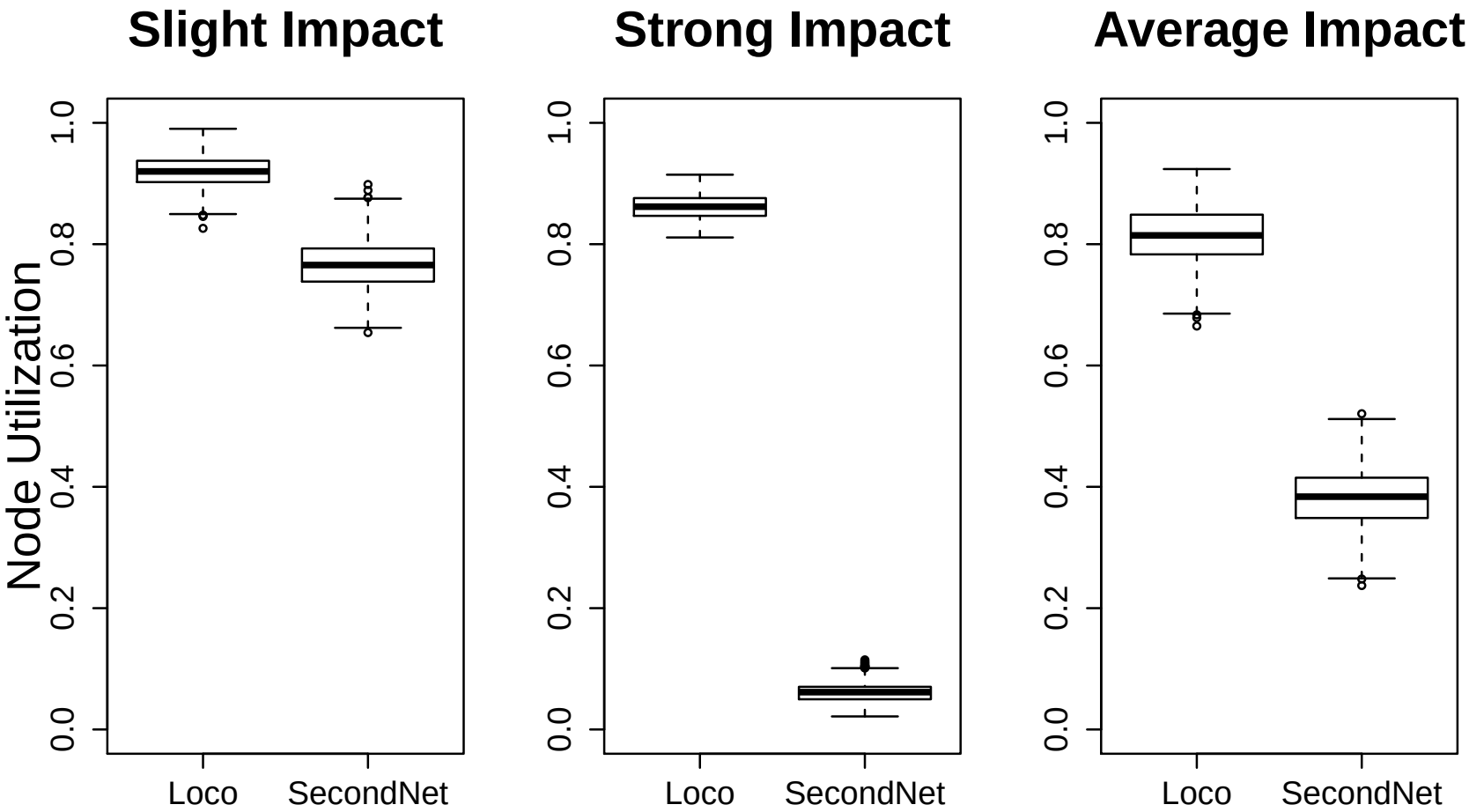
Impact of the collocation option



Impact of the collocation option

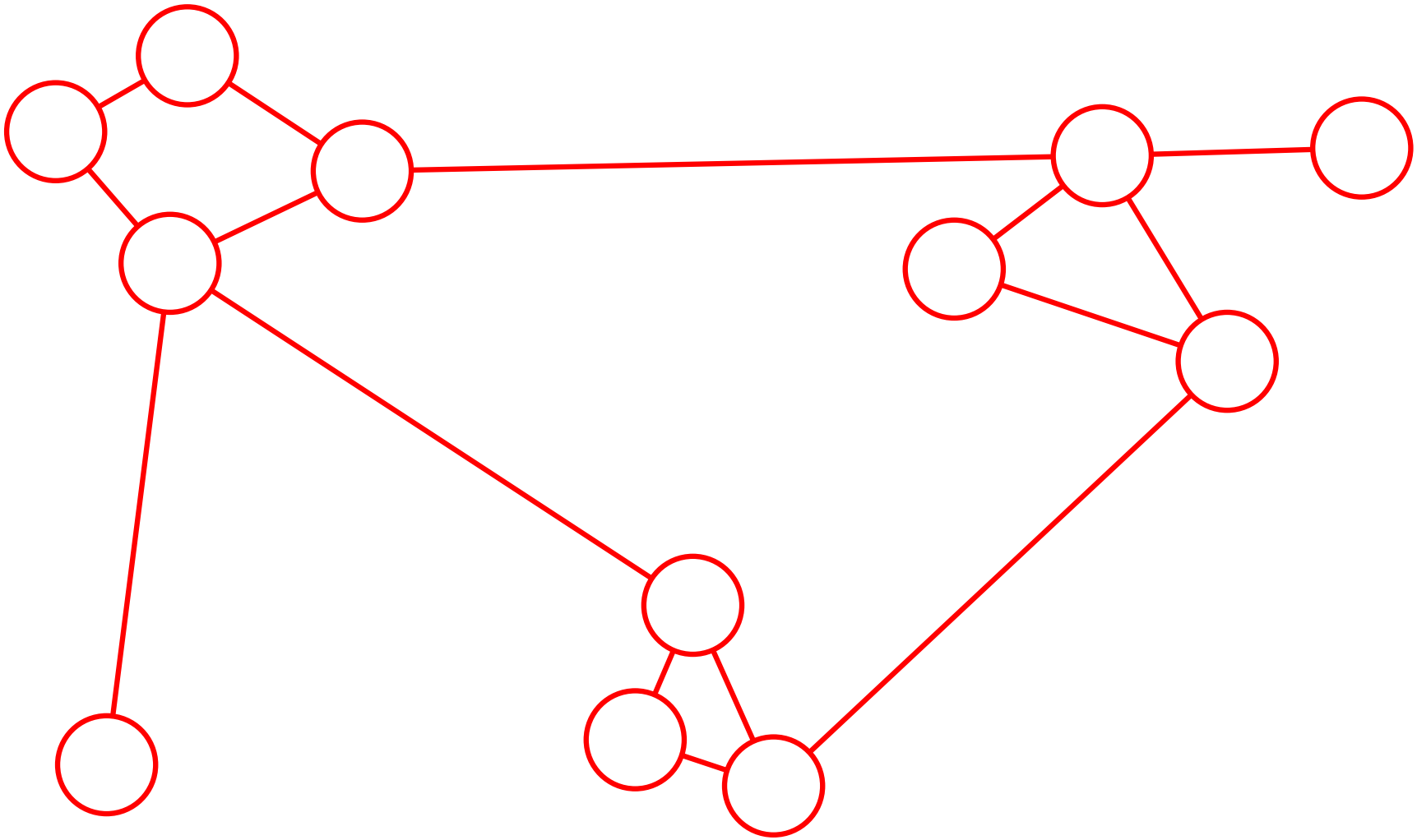


Impact of the collocation option

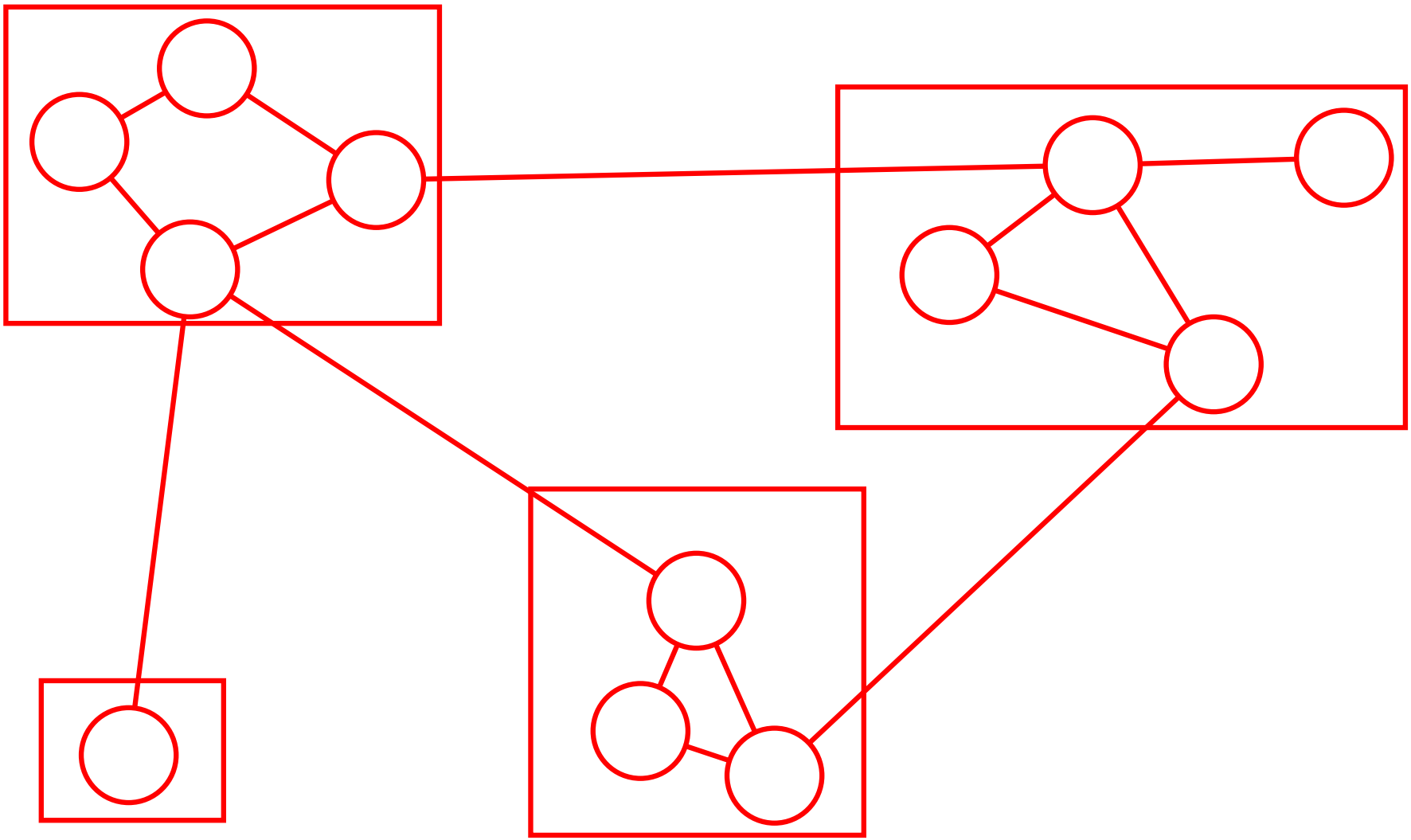


Can we leverage the benefits of collocation with the existing algorithms?

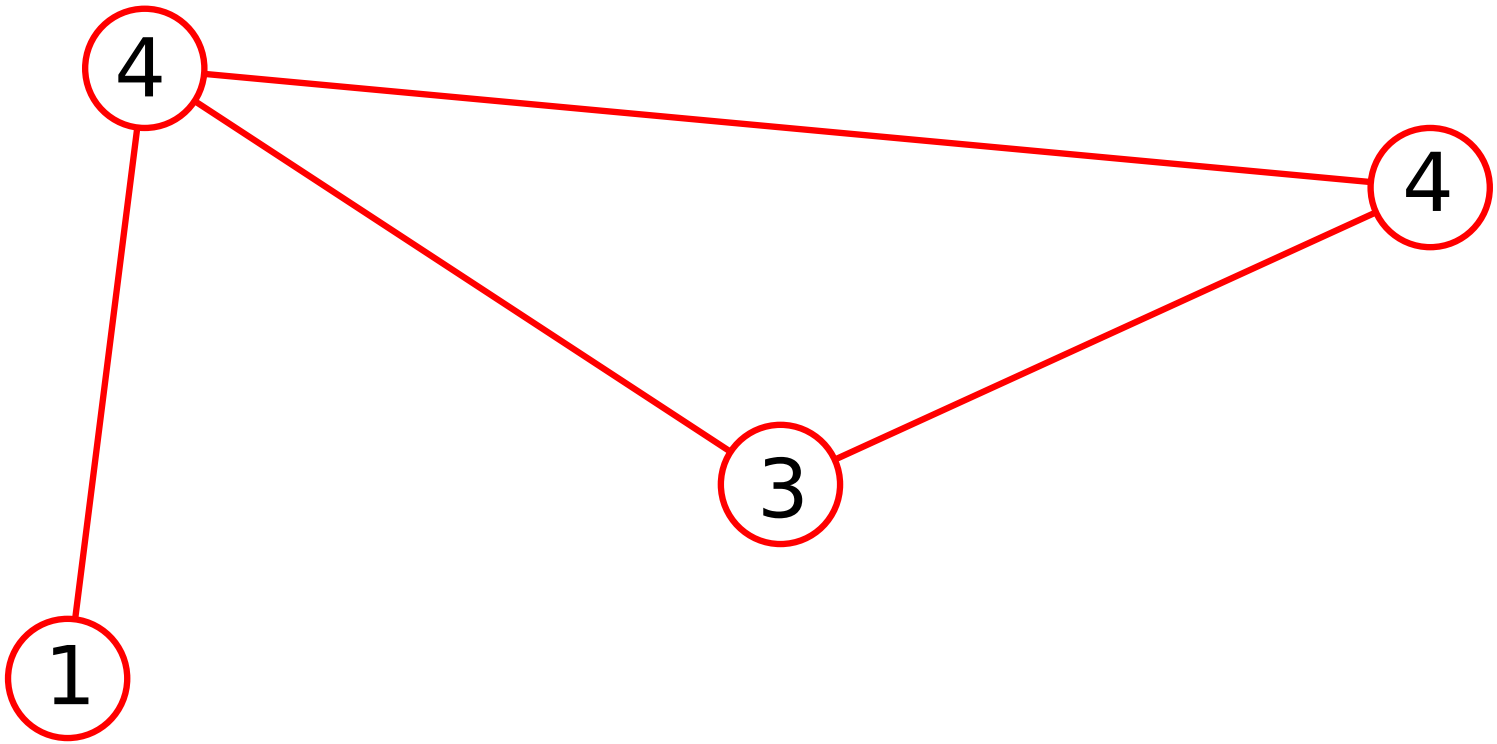
Pre-Clustering



Pre-Clustering



Pre-Clustering

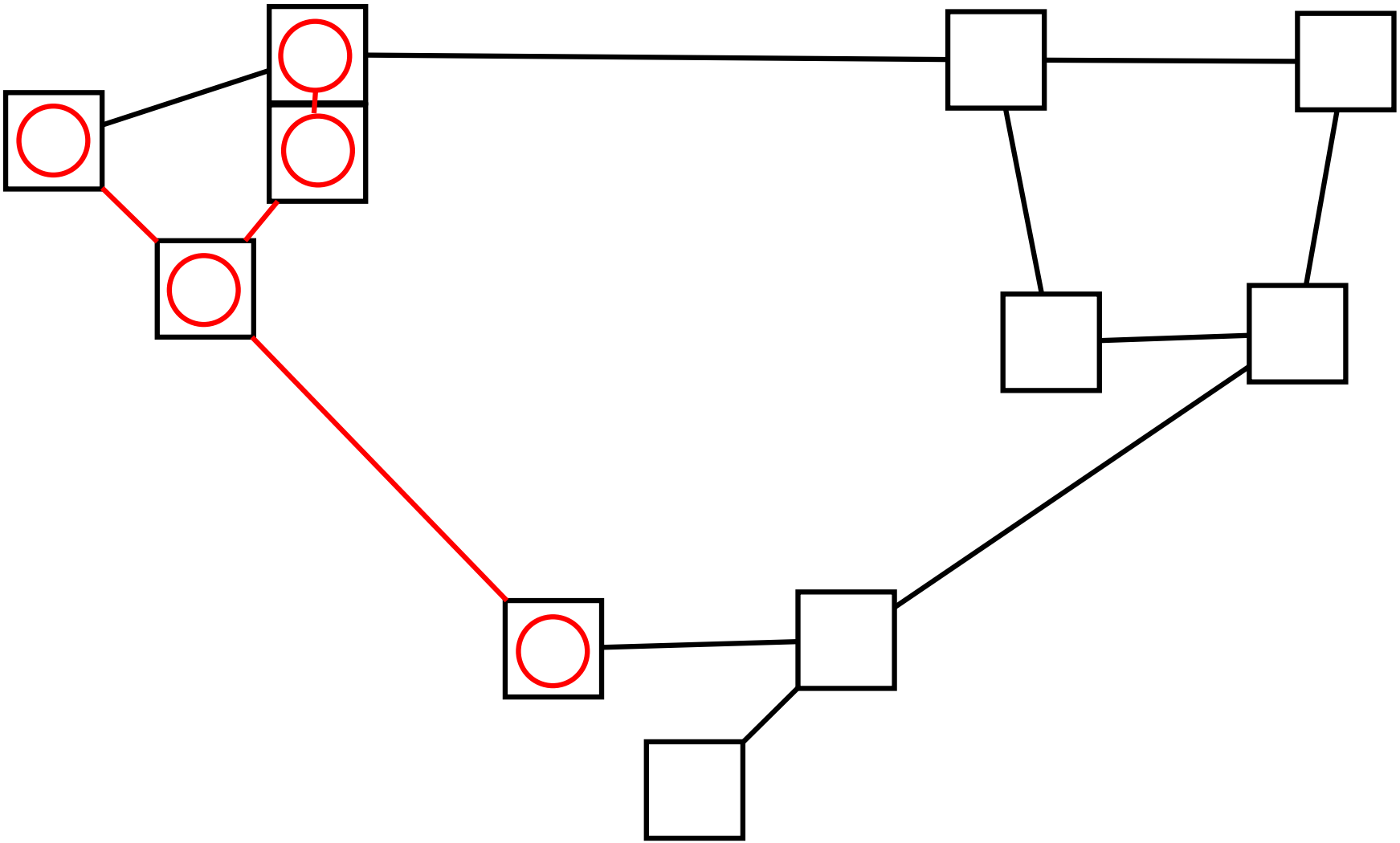


Pre-Clustering

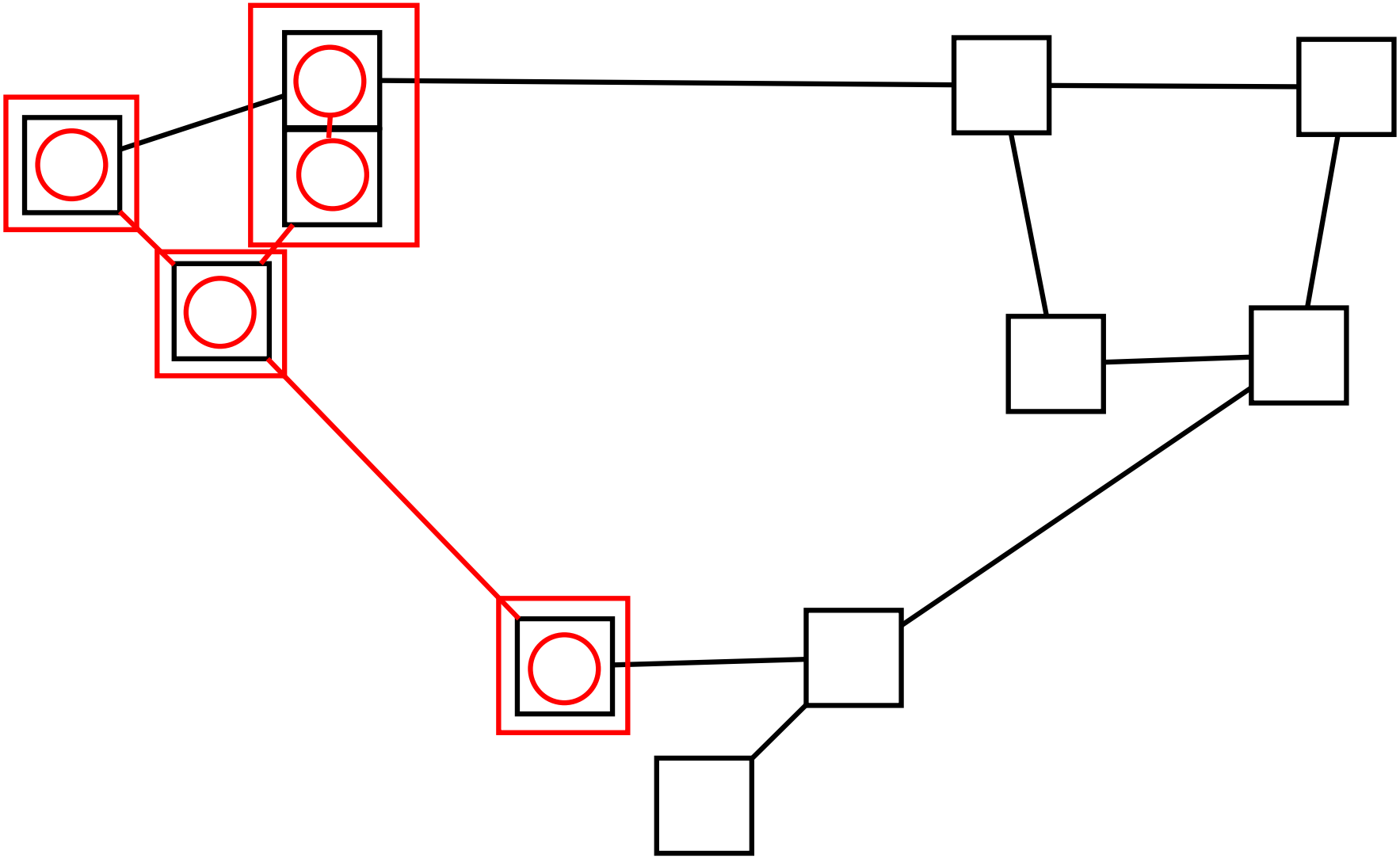
We use:

- Farhat
- LoCo
- OptCut (runtime optimized MIP)

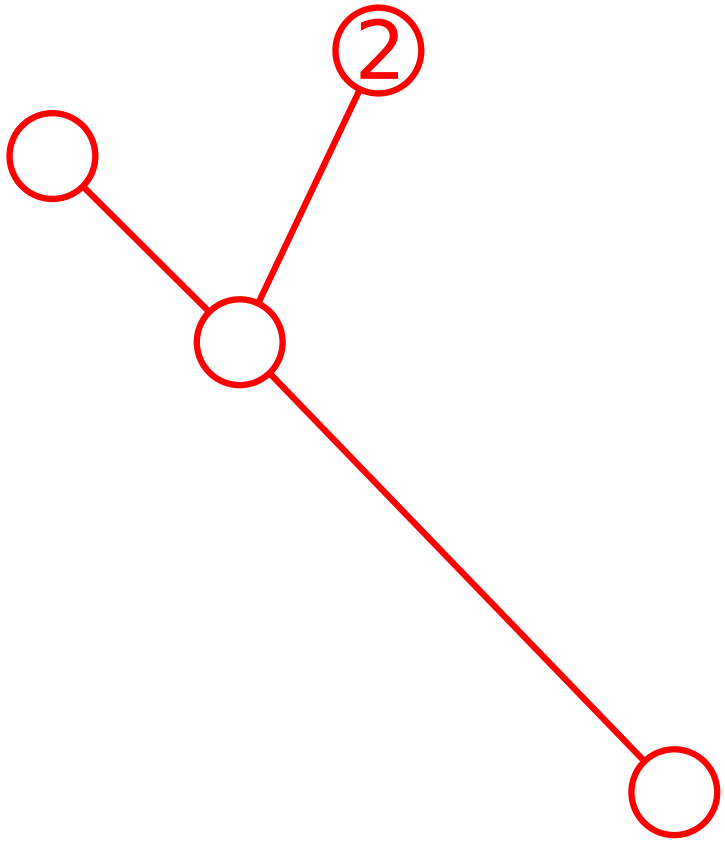
LoCo Preclustering



LoCo Preclustering



LoCo Preclustering

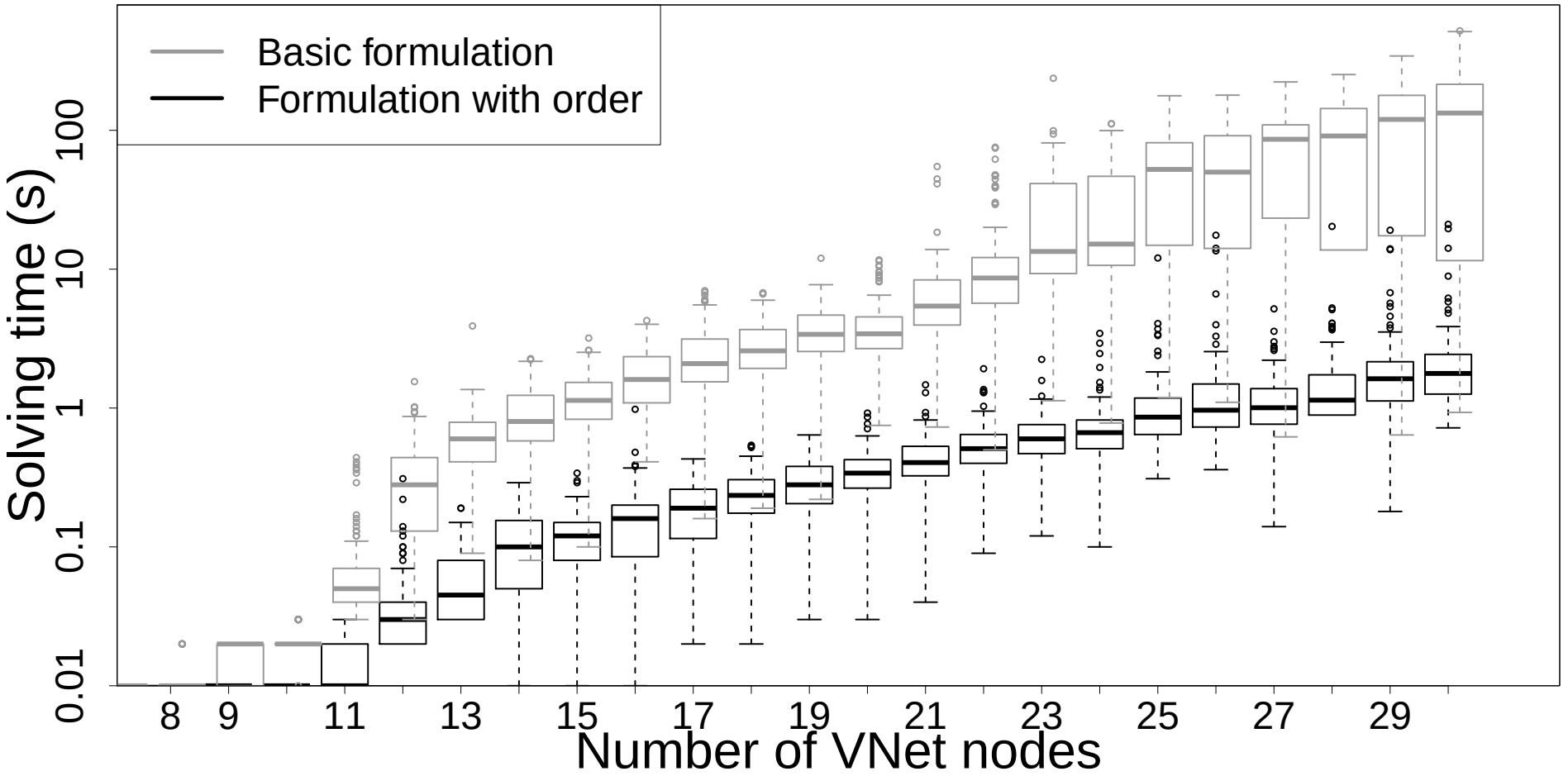


OptCut

- Generates an optimal (w.r.t. the amount of link resources between the merged nodes) Pre-Clustering
- Substrate is represented by two numbers:
  - MAX_V: The estimated host resources of a node
  - MAX_E: The estimated link resources attached to a node

  $\Rightarrow$ run time independent of substrate size and topology
- Removes symmetry from the problem to speed up the solution process

OptCut


Basic formulation
Formulation with order
Solving time (s)
0.01
0.1
1
10
100
8
9
11
13
15
17
19
21
23
25
27
29
Number of VNet nodes

Evaluation Parameters

Objective: Embed as many virtual resources as possible

Substrate

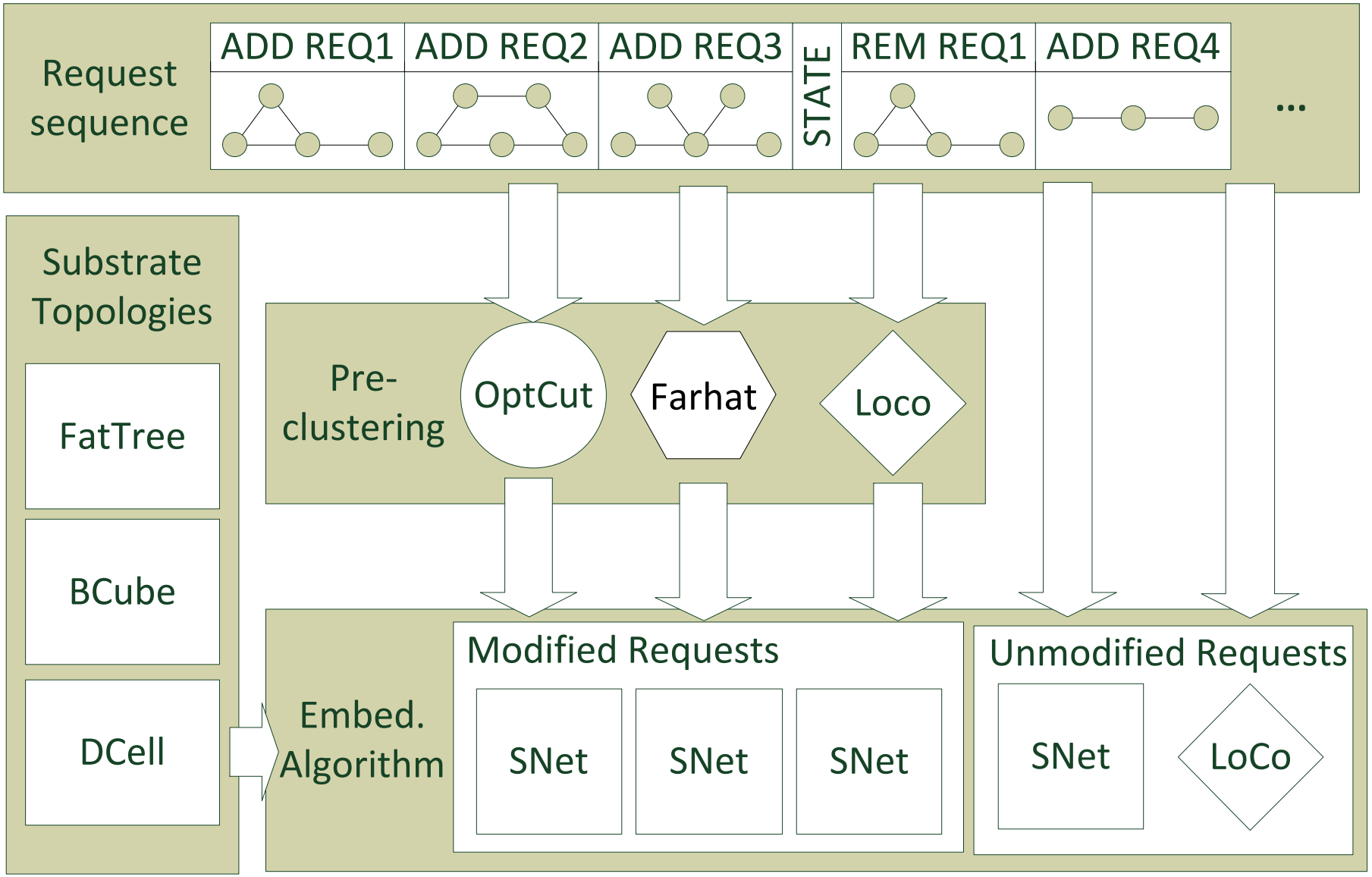
DC topologies (default FatTree with 432 hosts)
Each physical element has 4 resource units

Requests

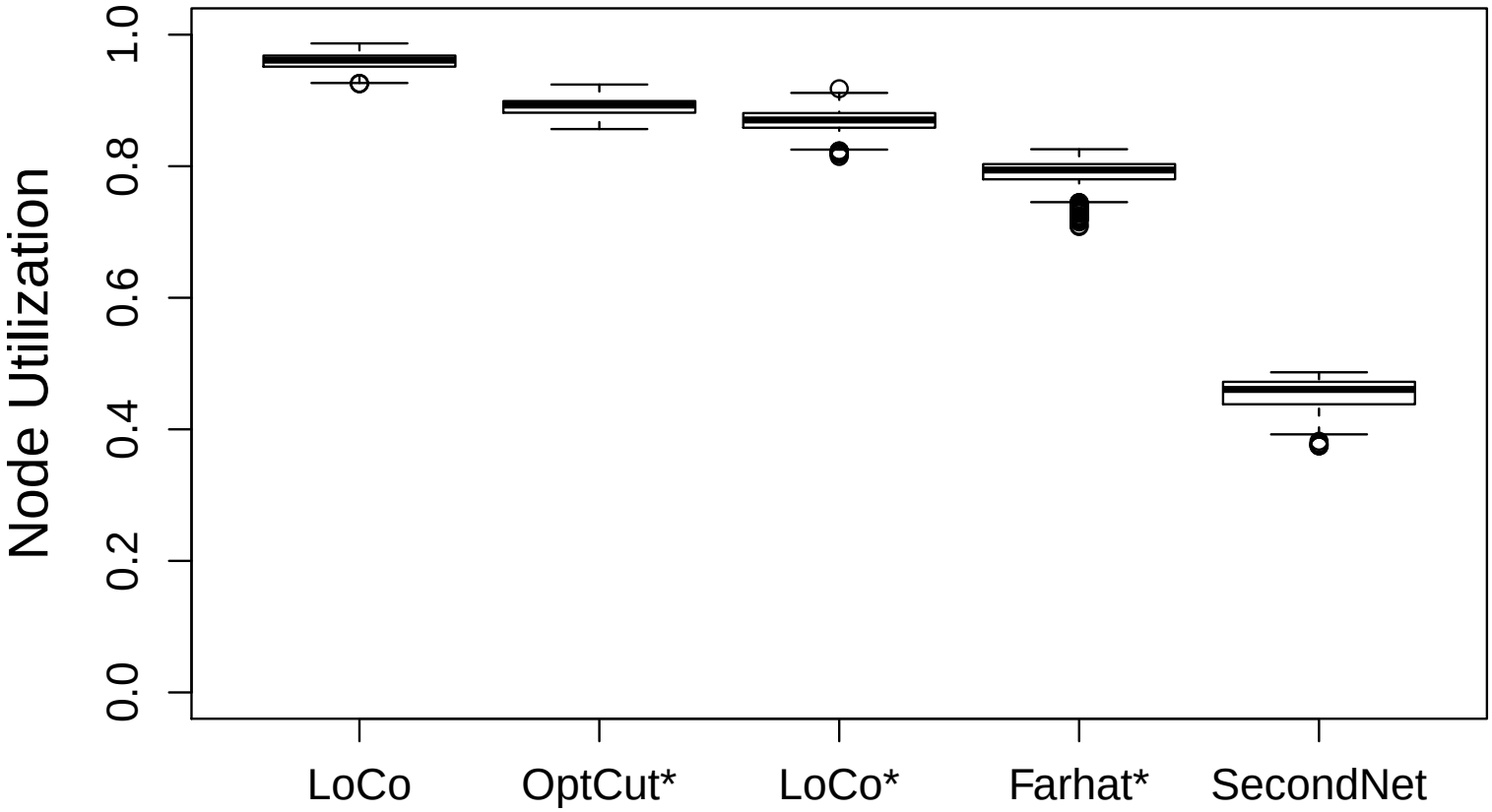
Randomized topologies (2-10 nodes, connection probability 0.15)
Exponentially distributed duration with mean 10
Resource sum of all requests $\approx$ available substrate resources

All Per-Clustering approaches are combined with SecondNet

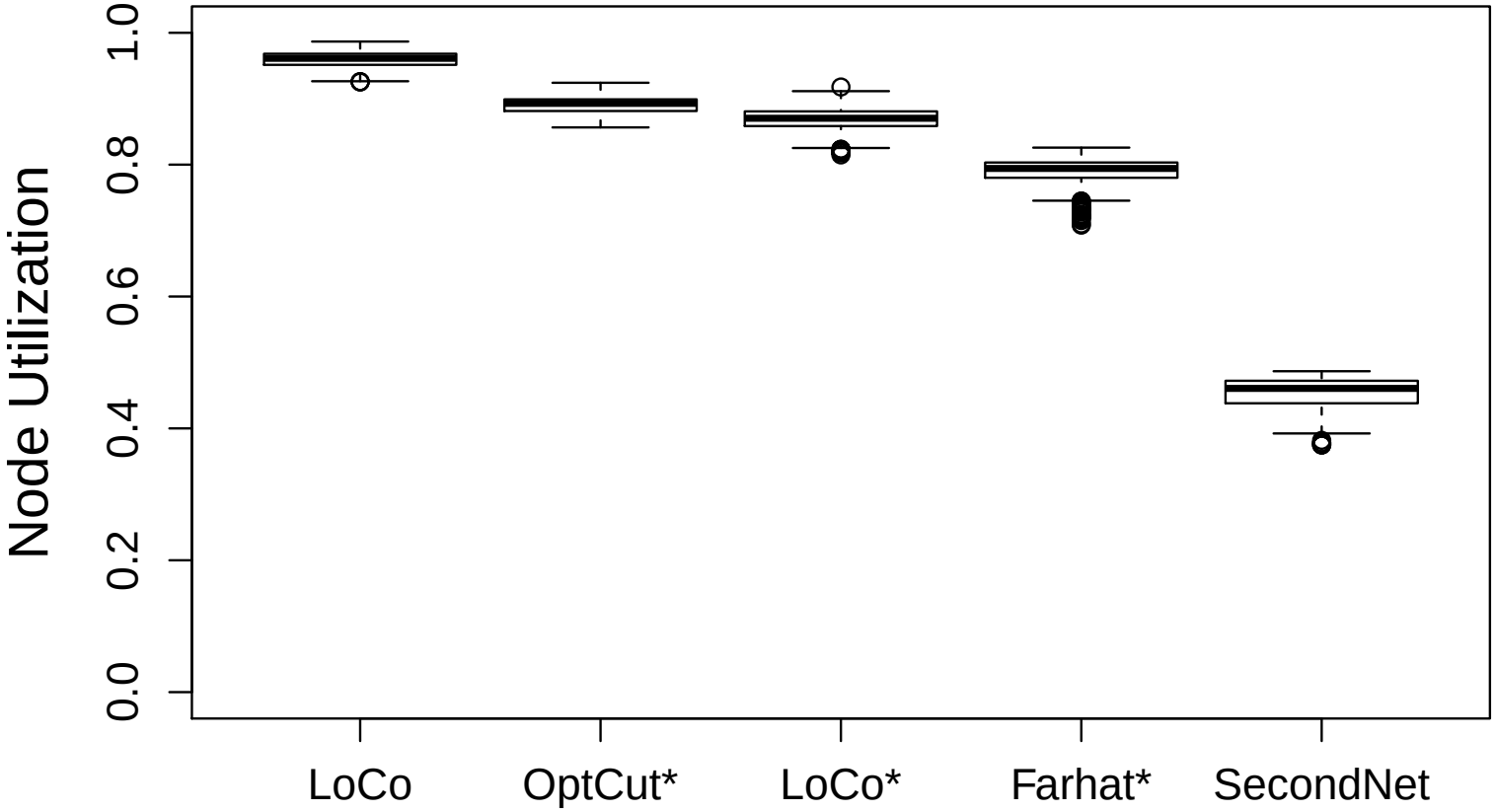
Experimental Pipeline



Performance Analysis



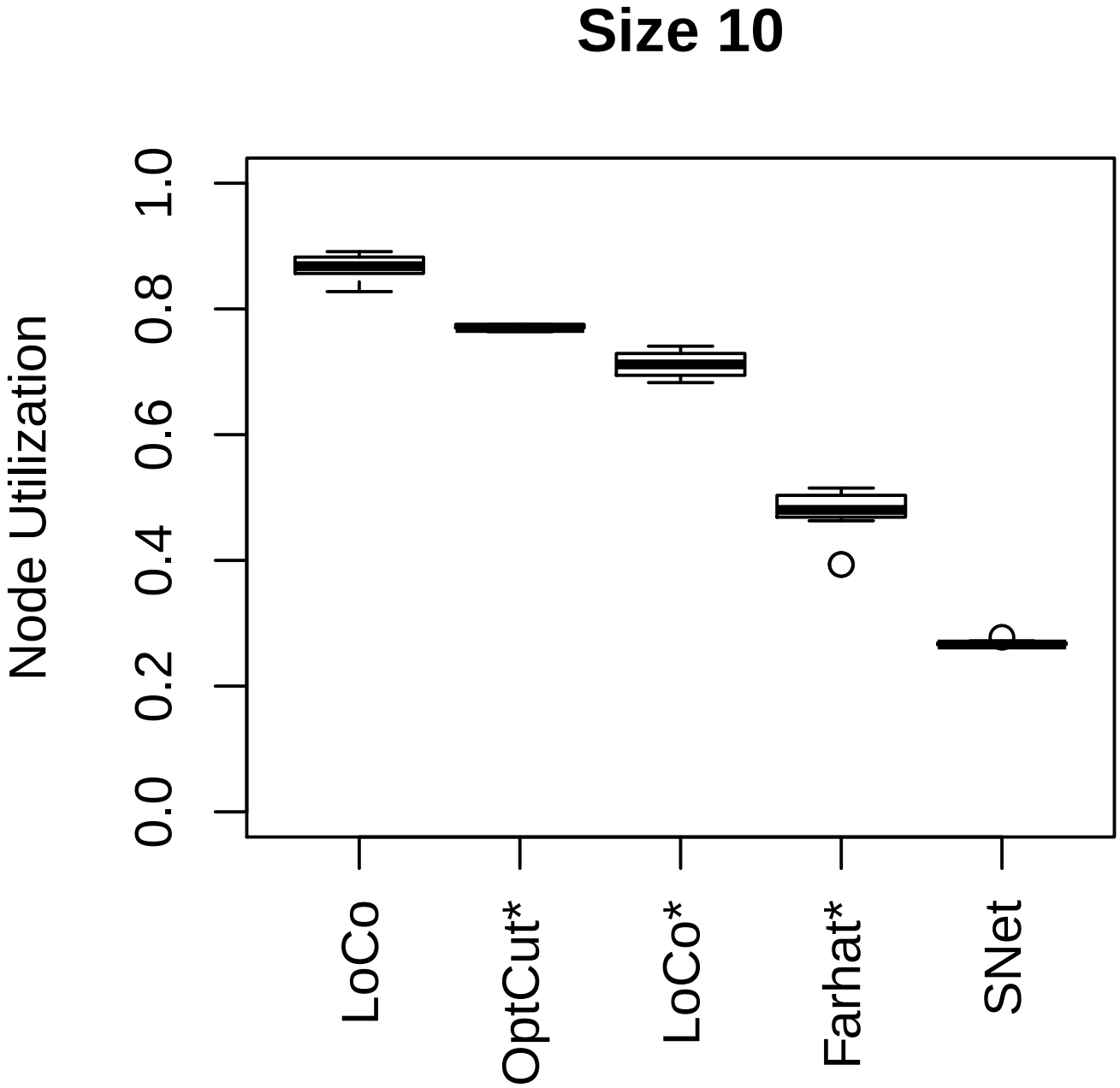
Performance Analysis



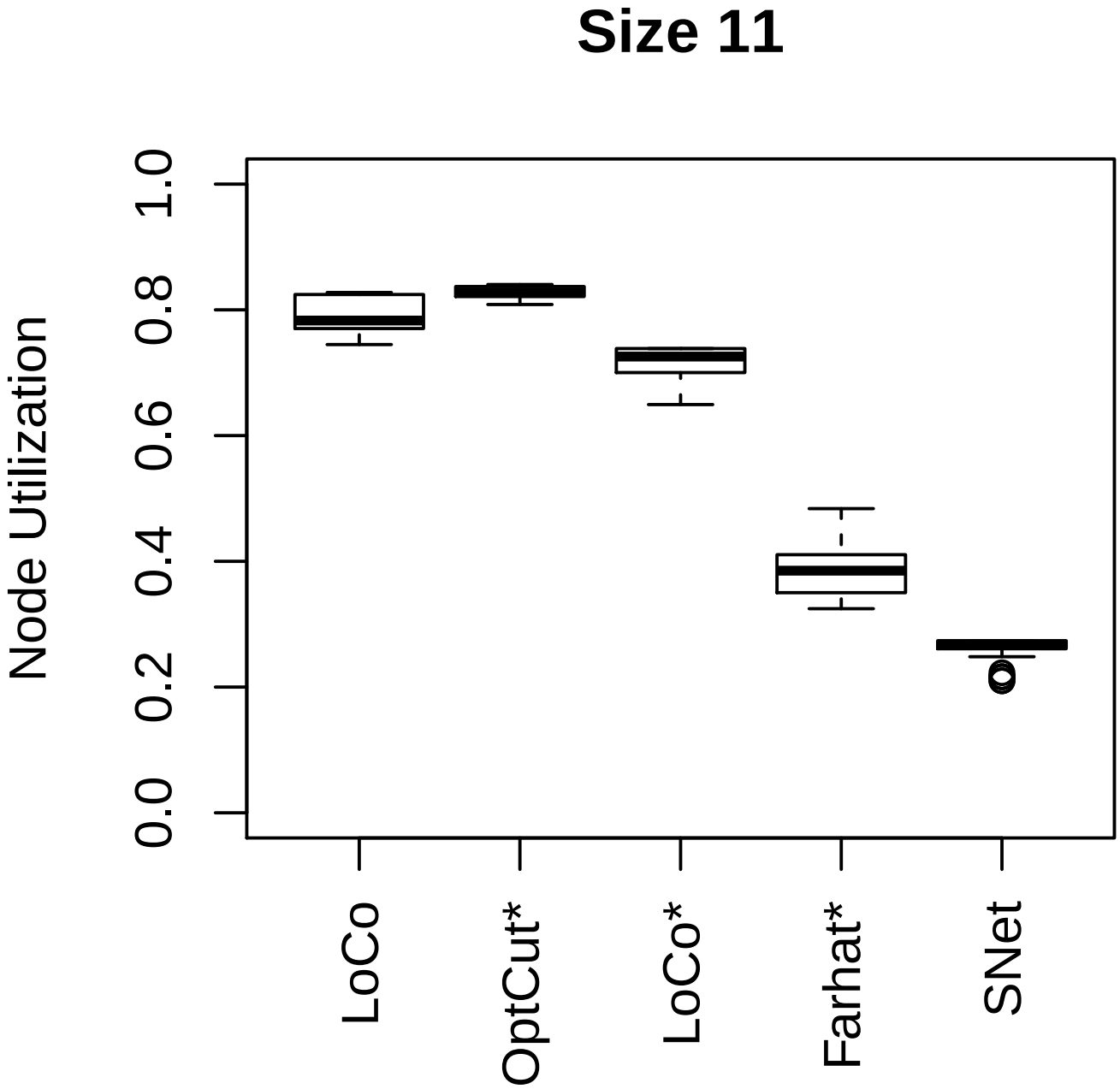
All Pre-Clustering approaches improve the performance of Secondnet by factors > 1.5

But why is standalone LoCo in this scenario more preformant?

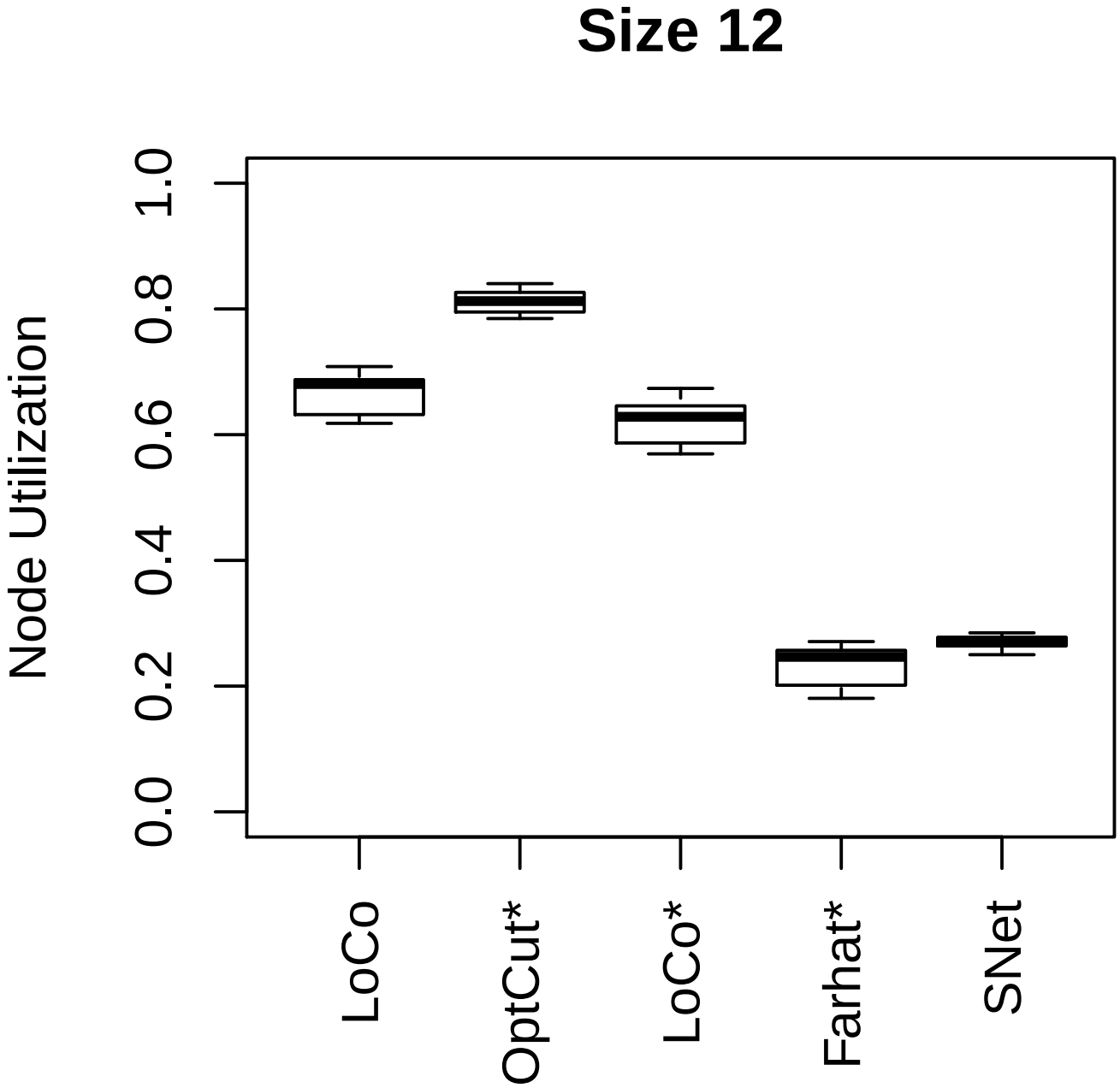
Reason I: Good Scenario for LoCo



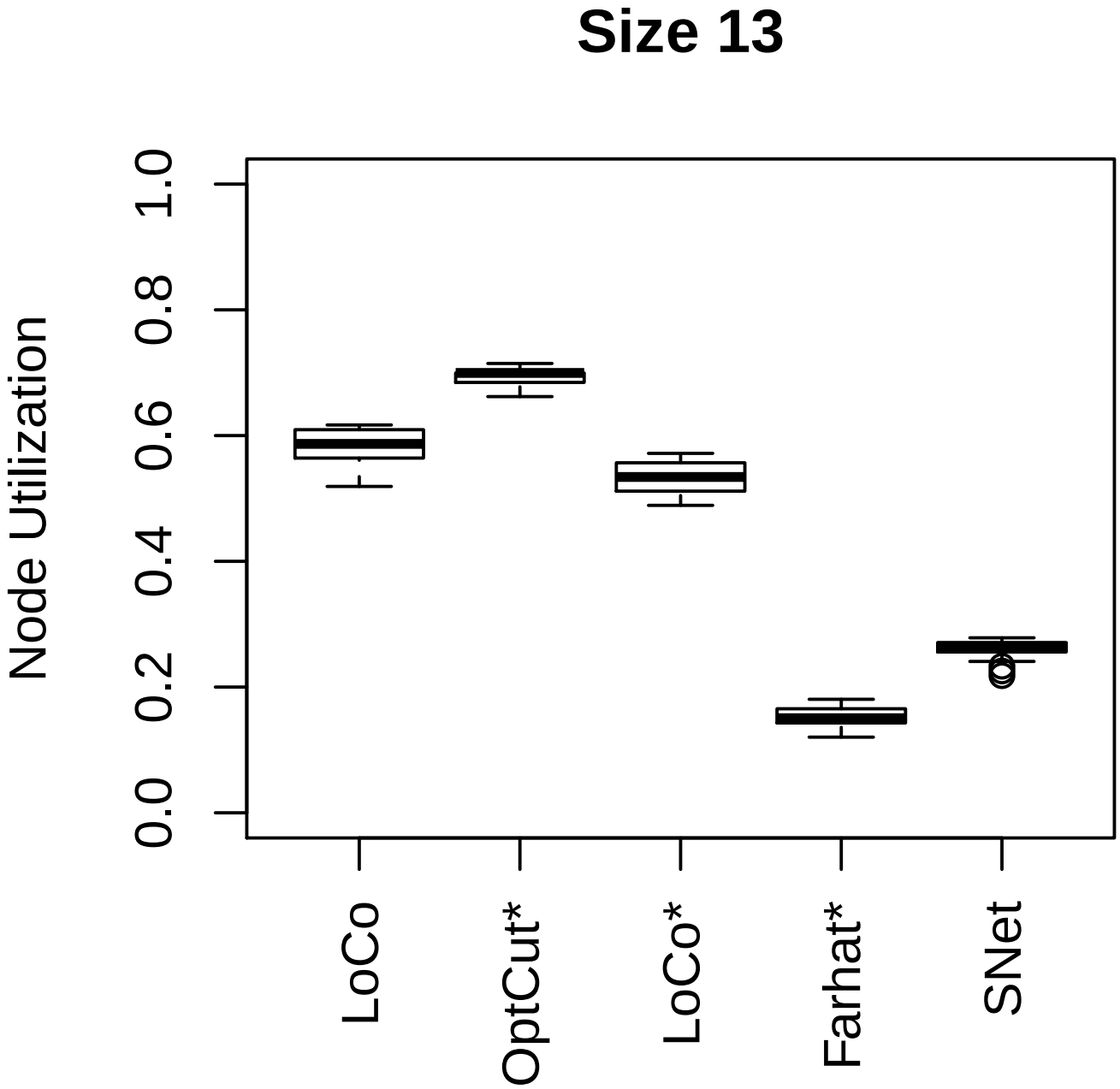
Reason I: Good Scenario for LoCo



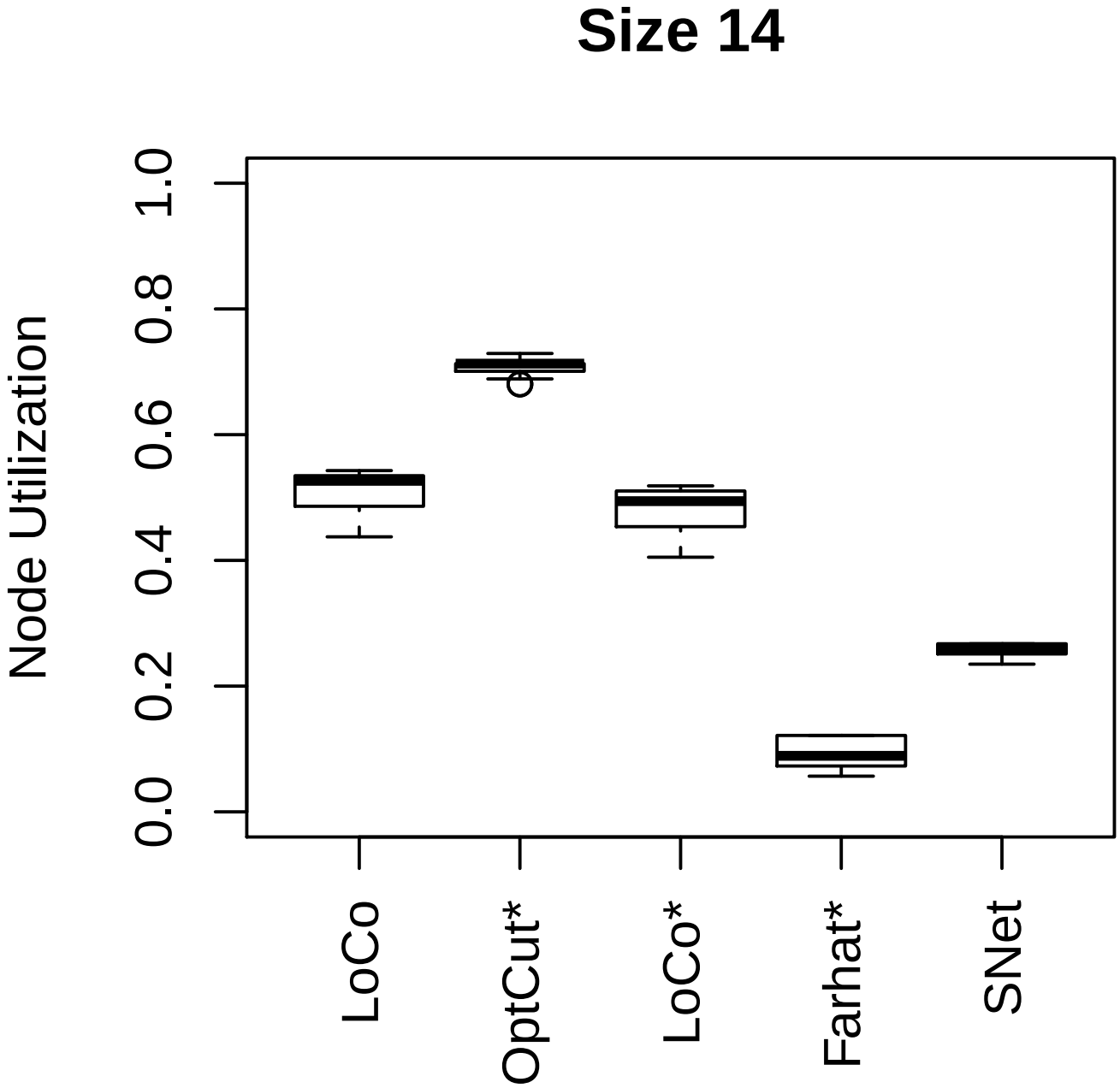
Reason I: Good Scenario for LoCo



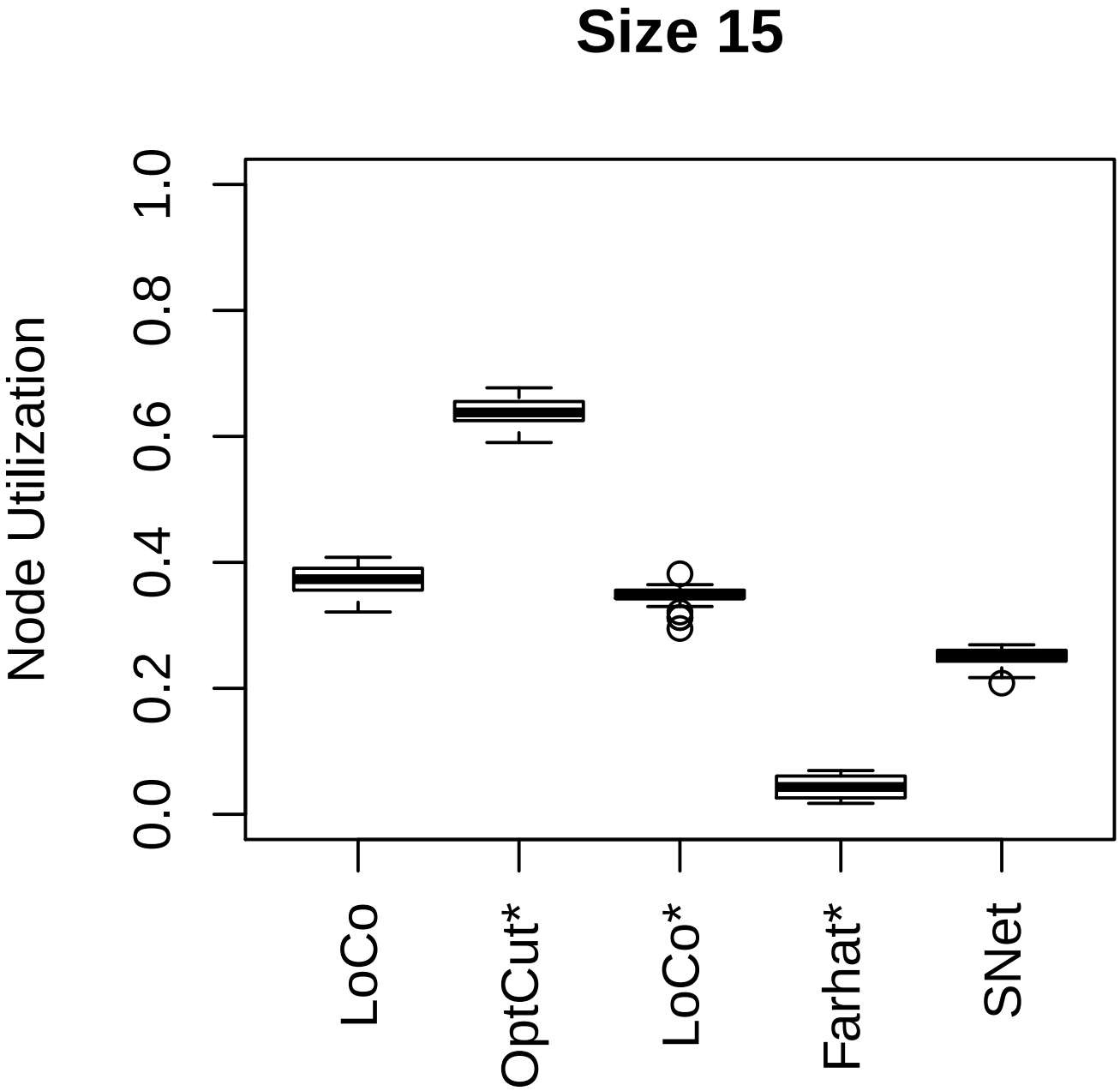
Reason I: Good Scenario for LoCo



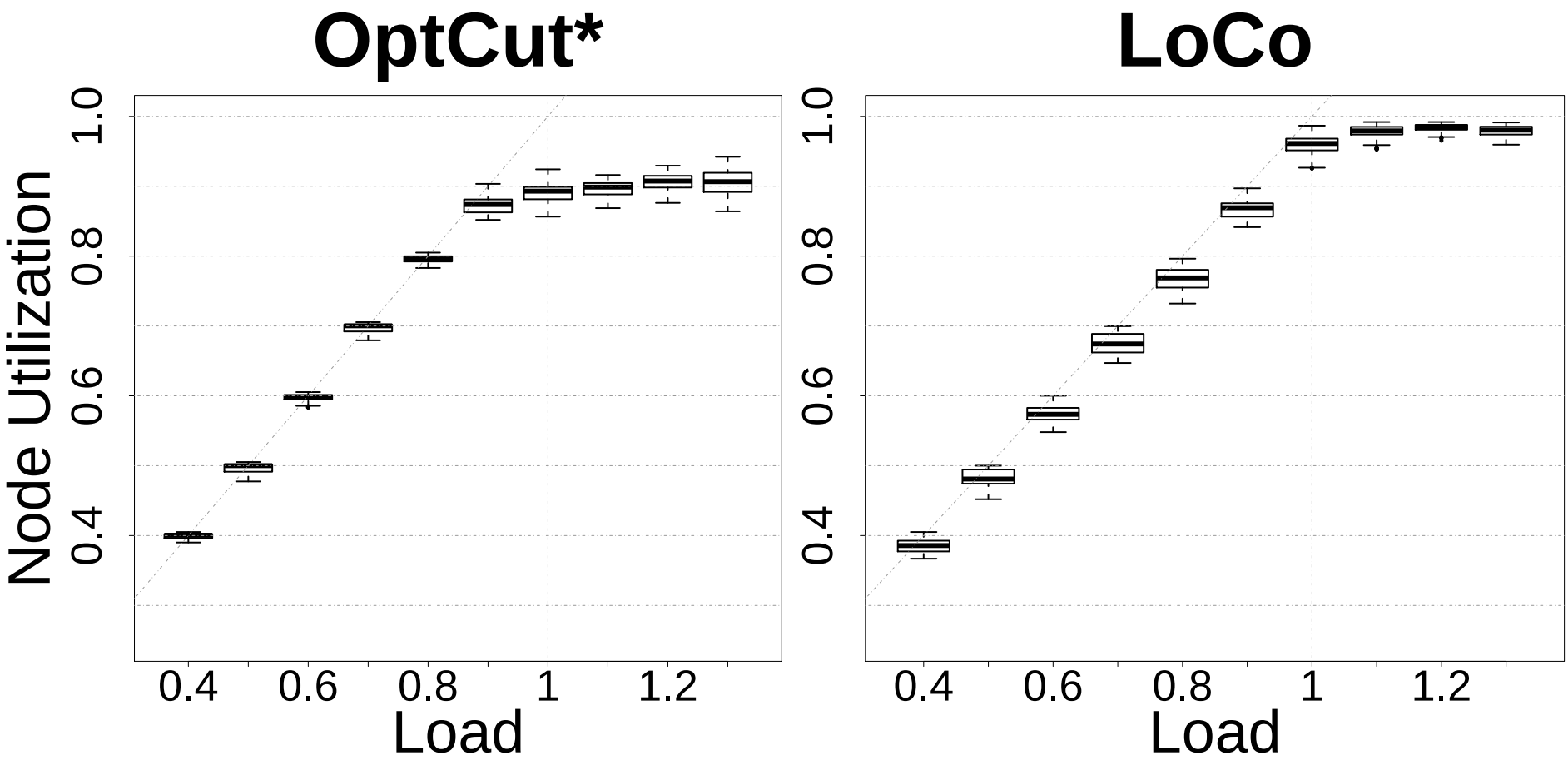
Reason I: Good Scenario for LoCo



Reason I: Good Scenario for LoCo



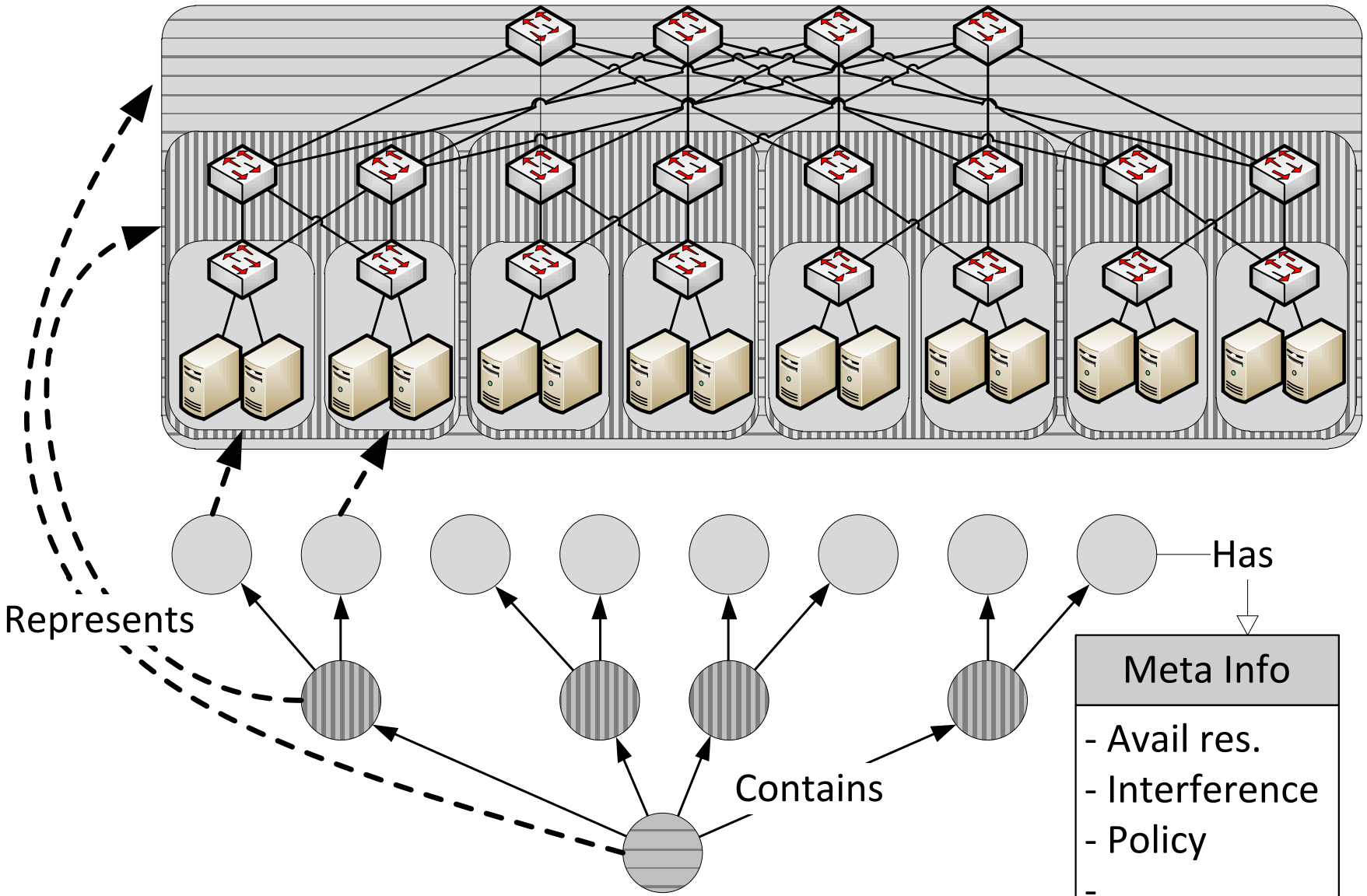
Reason II: Fragmented Residual Resources



What else is in the paper?

- Description of the MetaTree Framework

What else is in the paper?



What else is in the paper?

- Description of the MetaTree Framework
- Detailed description of LoCo

What else is in the paper?

Require: VNet $G = (V, E)$, $M = \{s\}$ for some $s \in V(G)$, $P = (\Gamma(s))$
  while $|P| > 0$ **do**
    sort P (* decreasing link capacities *)
    choose $u = P[0]$ (* next node to map *)
    map u (* forward checking *)
    map $\{u, v\} \quad \forall \quad v \in M$**, where** $\{u, v\} \in E(G)$
    $M = M \cup \{u\}$ **and** $P = P \setminus \{u\}$
  end while
  if (embedding failed), **backtrack** on s

What else is in the paper?

- Description of the MetaTree Framework
- Detailed description of LoCo
- Concrete MIP formulations and evaluation
  - Runtime comparison
  - Impact of MAX_E and MAX_V

What else is in the paper?

Constants:

Set of nodes:	V	(1)
Set of edges:	$E \subset V \times V$	(2)
Weights:	$W : V \cup E \rightarrow \mathbb{R}^{\geq 0}$	(3)
Maximal node resources:	MAX_V	(4)
Maximal link resources:	MAX_E	(5)
Larger nodes:	$\rho : V \rightarrow 2^V$	(6)

Variables:

PC

Node mapping:	$\text{alloc}_V : V \times V \rightarrow \{0, 1\}$	(7)
Auxiliary variable:	$x : E \times V \rightarrow \mathbb{R}^{\geq 0}$	(8)

...

What else is in the paper?

- Description of the MetaTree Framework
- Detailed description of LoCo
- Concrete MIP formulations and evaluation
  - Runtime comparison
  - Impact of MAX_E and MAX_V