Network migration scheduling

Mauricio G. C. Resende

Algorithms & Optimization Research Dept.

AT&T Labs Research ~ Shannon Lab

Florham Park ~ New Jersey

mgcr@research.att.com

Internet & Network Systems Research Center ~ Cookie talk

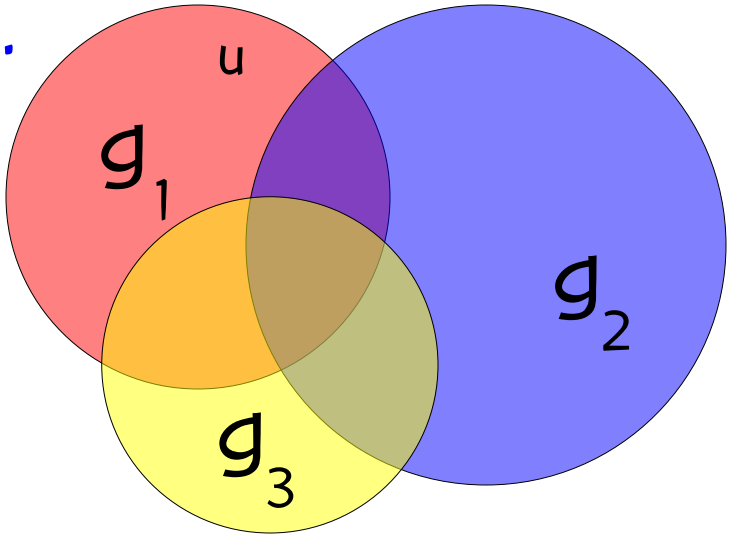
June 7, 2007 ~ Joint work with Diogo Andrade

Summary

- Batch scheduling of multi-grouped units
- Two applications
  - PBX telephone migration scheduling
  - Sequencing the 4ESS deloading process

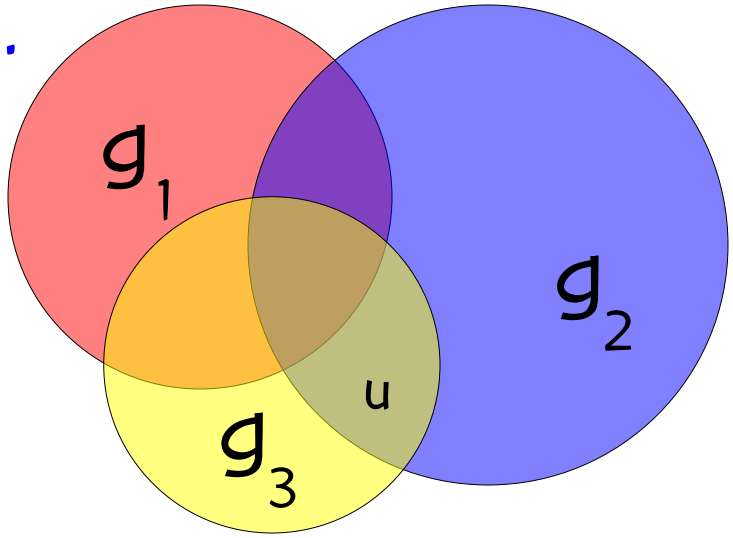
Batch scheduling of multi-grouped units

- Consider a system with
  - a set U of N units
  - a set H of M groups of units
- Each unit $u \in U$ is a member of one or more groups $g_1, g_2, \ldots, g_k \in H$.



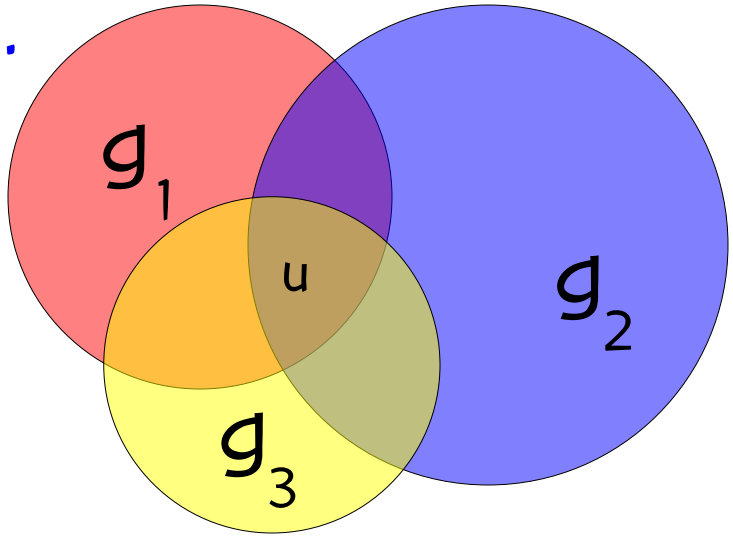
Batch scheduling of multi-grouped units

- Consider a system with
  - a set U of N units
  - a set H of M groups of units
- Each unit $u \in U$ is a member of one or more groups $g_1, g_2, \ldots, g_k \in H$.



Batch scheduling of multi-grouped units

- Consider a system with
  - a set U of N units
  - a set H of M groups of units
- Each unit $u \in U$ is a member of one or more groups $g_1, g_2, \ldots, g_k \in H$.



Batch scheduling of multi-grouped units

- Schedule: assign each unit to a time period.
- Given T time periods on which to schedule units.
- No more than C units can be assigned to a single time period.

22 units

Batch scheduling of multi-grouped units

- Schedule: assign each unit to a time period.
- Given T time periods on which to schedule units.
- No more than C units can be assigned to a single time period.

22 units

C=5 units/period

T=5 time periods

Batch scheduling of multi-grouped units

- Given T time periods on which to schedule units.
- No more than C units can be assigned to a single time period.

22 units

Schedule: assign each unit to a time period

C=5 units/period

T=5 time periods

Batch scheduling of multi-grouped units

- Schedule: assign each unit to a time period.
- Given T time periods on which to schedule units.
- No more than C units can be assigned to a single time period.

C=5 units/period T=5 time periods

Batch scheduling of multi-grouped units

- Objective: Schedule two units sharing same group as close together as possible.
- Let w(u,v,g) be the per-period penalty associated with assigning a group-g pair u and v to different periods.
- Scheduling penalty: Let $G(u,v) \subseteq H$ be the set of groups shared by units u and v. If units u and v are assigned to periods $\pi(u)$ and $\pi(v)$, respectively, then a penalty

$$p(u,v) = |\pi(u) - \pi(v)| \times \sum_{g \in G(u,v)} w(u,v,g)$$

is incurred.

Batch scheduling of multi-grouped units

- Objective: Schedule two units sharing same group as close together as possible.
- Let w(u,v,g) be the per-period penalty associated with assigning a group-g pair u and v to different periods.
- Scheduling penalty: Let $G(u,v) \subseteq H$ be the set of groups shared by units u and v. If units u and v are assigned to periods $\pi(u)$ and $\pi(v)$, respectively, then a penalty

$$p(u,v) = |\pi(u) - \pi(v)| \times \sum_{g \in G(u,v)} w(u,v,g)$$

is incurred.

Batch scheduling of multi-grouped units

- Objective: Schedule two units sharing same group as close together as possible.
- Let w(u,v,g) be the per-period penalty associated with assigning a group-g pair u and v to different periods.
- Scheduling penalty: Let $G(u,v) \subseteq H$ be the set of groups shared by units u and v. If units u and v are assigned to periods $\pi(u)$ and $\pi(v)$, respectively, then a penalty

$$p(u,v) = |\pi(u) - \pi(v)| \times \sum_{g \in G(u,v)} w(u,v,g)$$

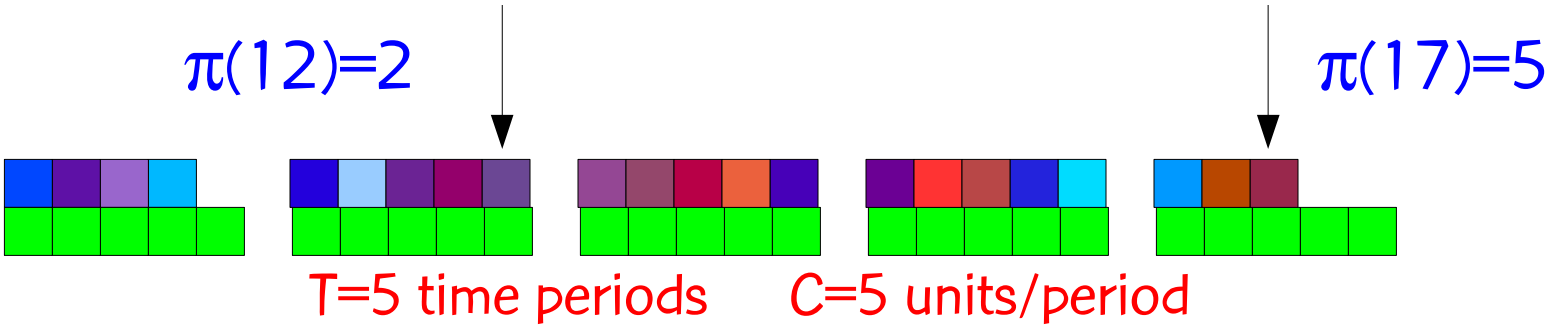
is incurred.

Batch scheduling of multi-grouped units

12

17

Units 12 and 17 share groups 2, 4, and 8.



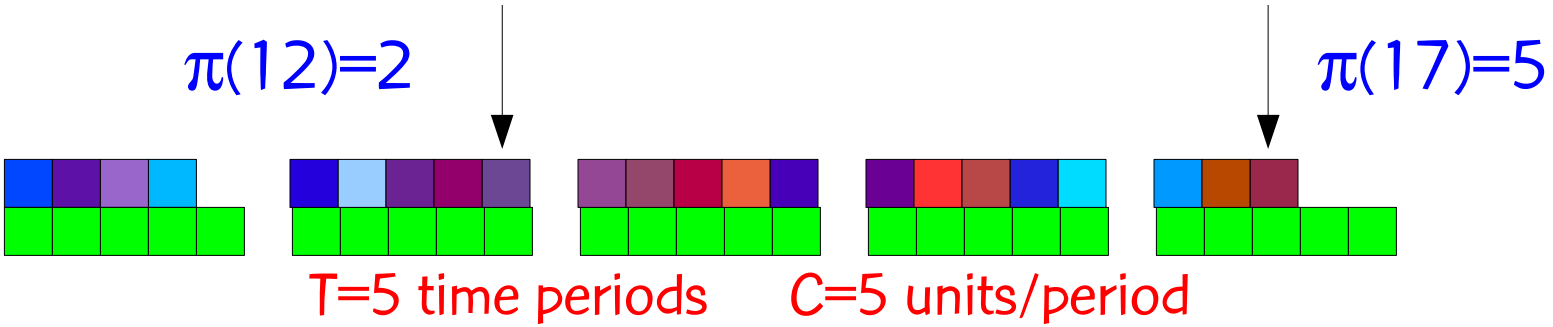
Batch scheduling of multi-grouped units

12

17

Units 12 and 17 share groups 2, 4, and 8.

Let w(12,17,2) = 10, w(12,17,4) = 20, w(12,17,8) = 5.



Batch scheduling of multi-grouped units

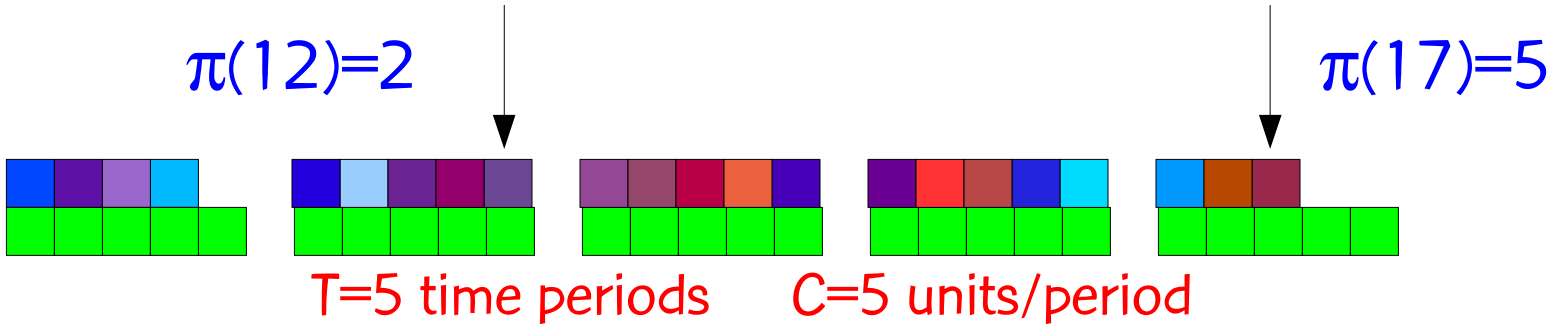
12

17

Units 12 and 17 share groups 2, 4, and 8.

Let w(12,17,2) = 10, w(12,17,4) = 20, w(12,17,8) = 5.

Then w(12,17,2) + w(12,17,4) + w(12,17,8) = 35.



Batch scheduling of multi-grouped units

12

17

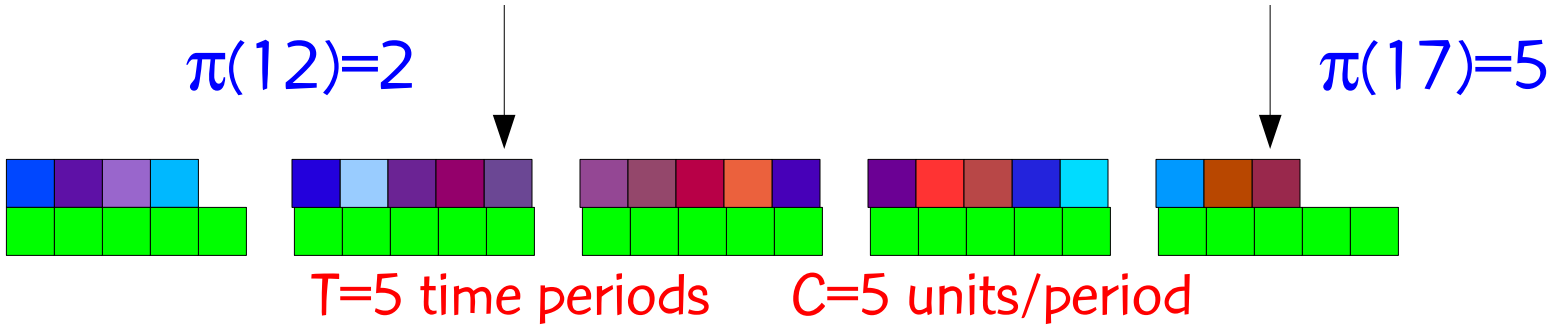
Units 12 and 17 share groups 2, 4, and 8.

Let w(12,17,2) = 10, w(12,17,4) = 20, w(12,17,8) = 5.

Then w(12,17,2) + w(12,17,4) + w(12,17,8) = 35.

Since $\pi(12)=2$ and $\pi(17)=5$, then

$p(12,17) = |5-2| \times 35 = 105$.



Batch scheduling of multi-grouped units

- Problem: Find assignment π of units to periods that

  minimizes $\sum_{u,v \in U \times U \atop (u>v)} |\pi(u)-\pi(v)| \times \sum_{g \in G(u,v)} w(u,v,g)$

  such that

  no more than C units are assigned to any time period.

- Problem is NP-hard. It generalizes the minimum linear arrangement problem: Given a graph $G(V,E)$, find $\pi: V \rightarrow \{1,\ldots,|V|\}$ that

  minimizes $\sum_{(u,v) \in E} |\pi(u)-\pi(v)|$

Batch scheduling of multi-grouped units

- Problem: Find assignment π of units to periods that

  minimizes $\sum_{u,v \in U \times U \atop (u>v)} |\pi(u)-\pi(v)| \times \sum_{g \in G(u,v)} w(u,v,g)$

  such that

  no more than C units are assigned to any time period.

- Problem is NP-hard. It generalizes the minimum linear arrangement problem: Given a graph $G(V,E)$, find $\pi: V \rightarrow \{1,\ldots,|V|\}$ that minimizes $\sum_{(u,v) \in E} |\pi(u)-\pi(v)|$

Batch scheduling of multi-grouped units

- Problem: Find assignment π of units to periods that

  minimizes $\sum_{u,v \in U \times U \atop (u>v)} |\pi(u)-\pi(v)| \times \sum_{g \in G(u,v)} w(u,v,g)$

  such that

  no more than C units are assigned to any time period.

- Problem is NP-hard. It generalizes the minimum linear arrangement problem: Given a graph $G(V,E)$,

  find $\pi: V \rightarrow \{1,\ldots,|V|\}$ that

  minimizes $\sum_{(u,v) \in E} |\pi(u)-\pi(v)|$

Our solution

- Metaheuristics are high level procedures that coordinate simple heuristics, such as local search, to find solutions that are of better quality than those found by the simple heuristics alone.
- One such metaheuristic is GRASP (Feo & R., 1995)
  - repeat ...
    - construct solution using randomized greedy algorithm
    - do local search starting from constructed solution
  - return best local minimum found

Our solution

- Hybridization of metaheuristics
- GRASP with evolutionary path-relinking (PR)
  - repeat … maintaining pool of elite solutions
    - construct solution using randomized greedy algorithm
    - do local search starting from constructed solution
    - PR: explore path connecting local min and some elite solution
    - once in while do evolutionary PR, i.e. improve pool by exploring paths connecting elite solutions
  - return best elite solution found

PBX telephone migration scheduling

PBX telephone migration scheduling

- Phone migration occurs when an organization upgrades to a newer phone switch (PBX).
  - All phones using the old PBX must be moved to the new PBX.
  - Each phone belong to one or more groups of phones that interact and should to be moved together in same time period.
  - Given penalties for not moving a pair of phones together and a maximum number of phones that can be moved in a time period, find assignment of phones to periods such that total penalty is minimized.

PBX telephone migration scheduling

- Phone migration occurs when an organization upgrades to a newer phone switch (PBX).
- All phones using the old PBX must be moved to the new PBX.
  - Each phone belong to one or more groups of phones that interact and should to be moved together in same time period.
  - Given penalties for not moving a pair of phones together and a maximum number of phones that can be moved in a time period, find assignment of phones to periods such that total penalty is minimized.

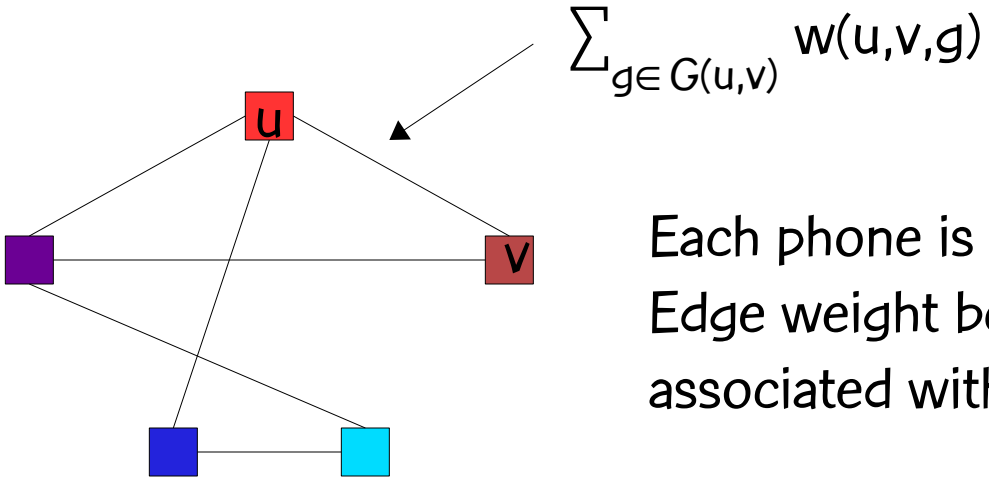
PBX telephone migration scheduling

- Phone migration occurs when an organization upgrades to a newer phone switch (PBX).
- All phones using the old PBX must be moved to the new PBX.
- Each phone belongs to one or more groups of phones that interact and should to be moved together in same time period.
- Given penalties for not moving a pair of phones together and a maximum number of phones that can be moved in a time period, find assignment of phones to periods such that total penalty is minimized.

PBX telephone migration scheduling

- Phone migration occurs when an organization upgrades to a newer phone switch (PBX).
- All phones using the old PBX must be moved to the new PBX.
- Each phone belong to one or more groups of phones that interact and should to be moved together in same time period.
- Given penalties for not moving a pair of phones together and a maximum number of phones that can be moved in a time period, find assignment of phones to periods such that total penalty is minimized.

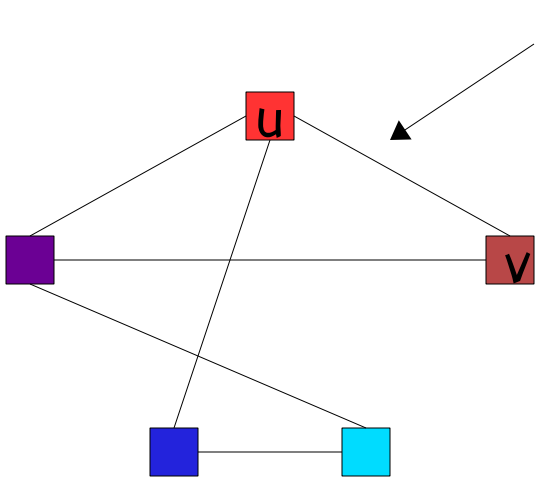
PBX telephone migration scheduling



Each phone is a unit.

Edge weight between phones is penalty associated with phone pair.

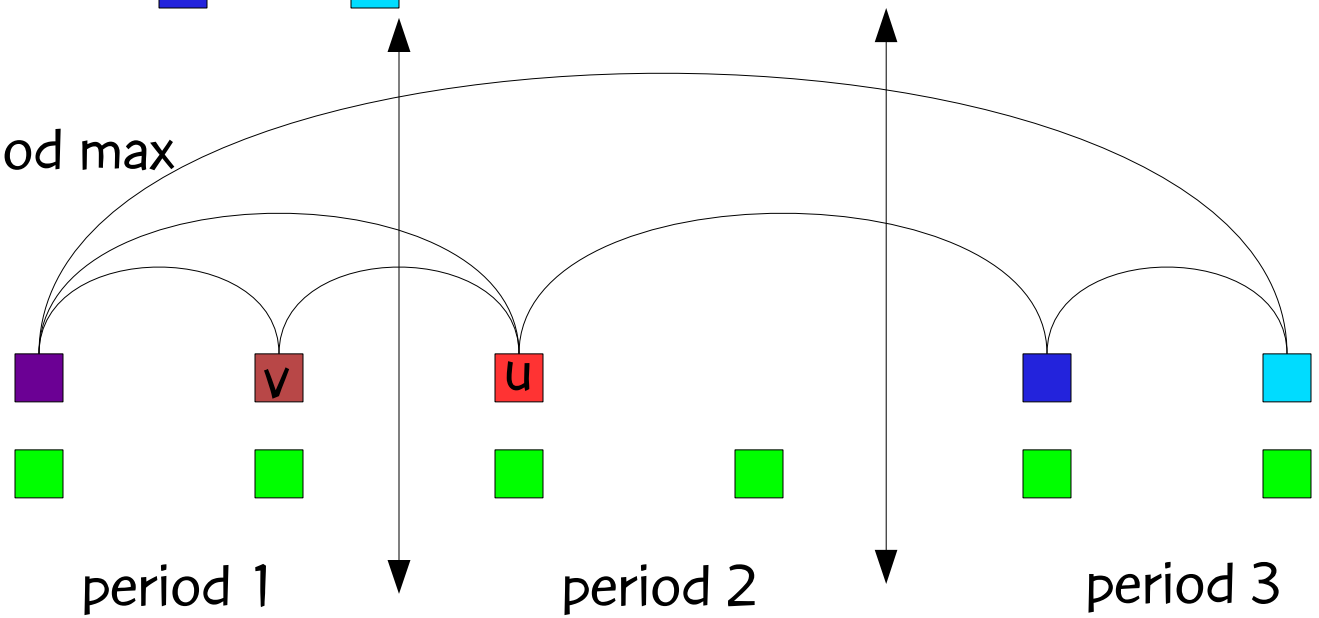
PBX telephone migration scheduling



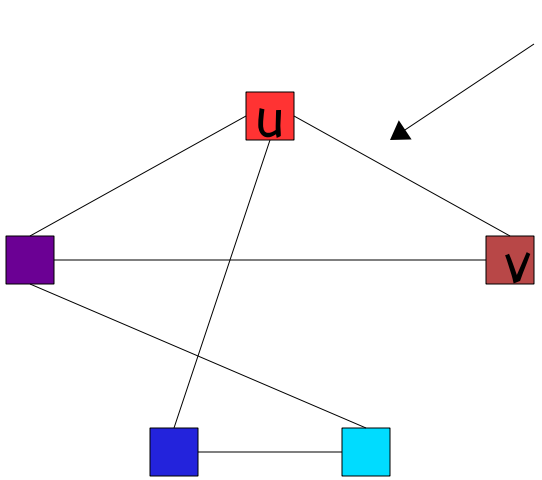
$$\sum_{g \in G(u,v)} w(u,v,g)$$

Each phone is a unit.
Edge weight between phones is penalty associated with phone pair.

3 periods
2 units/period max



PBX telephone migration scheduling

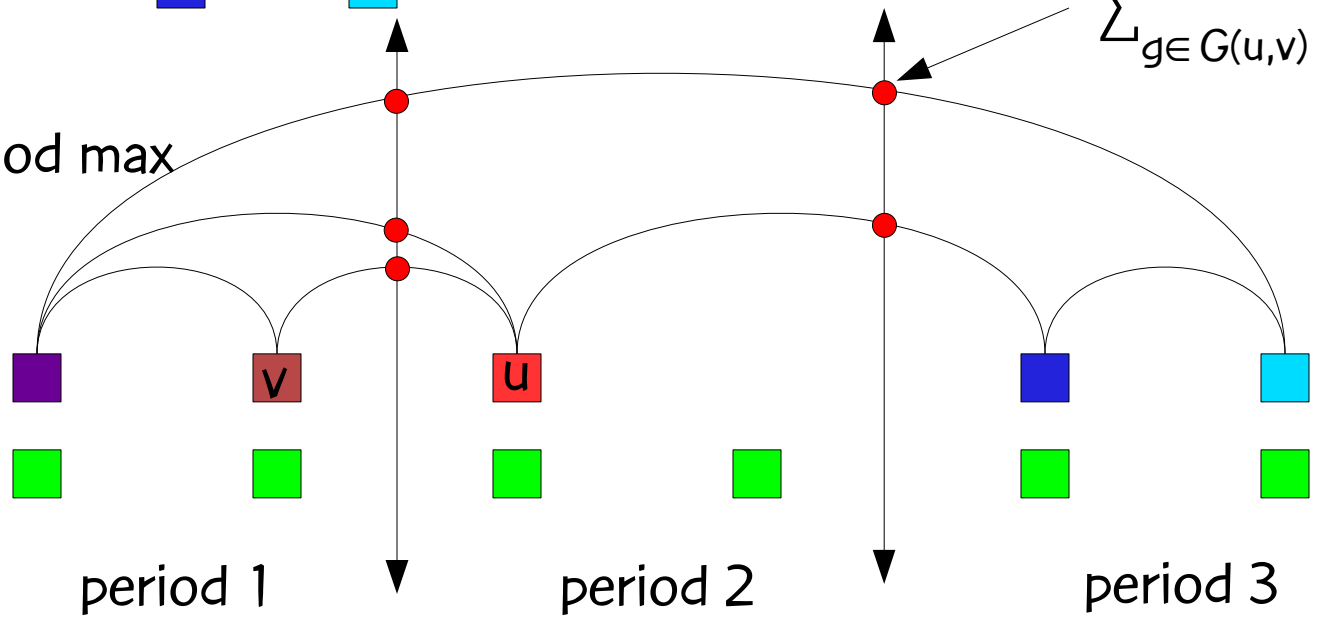


$\sum_{g \in G(u,v)} w(u,v,g)$

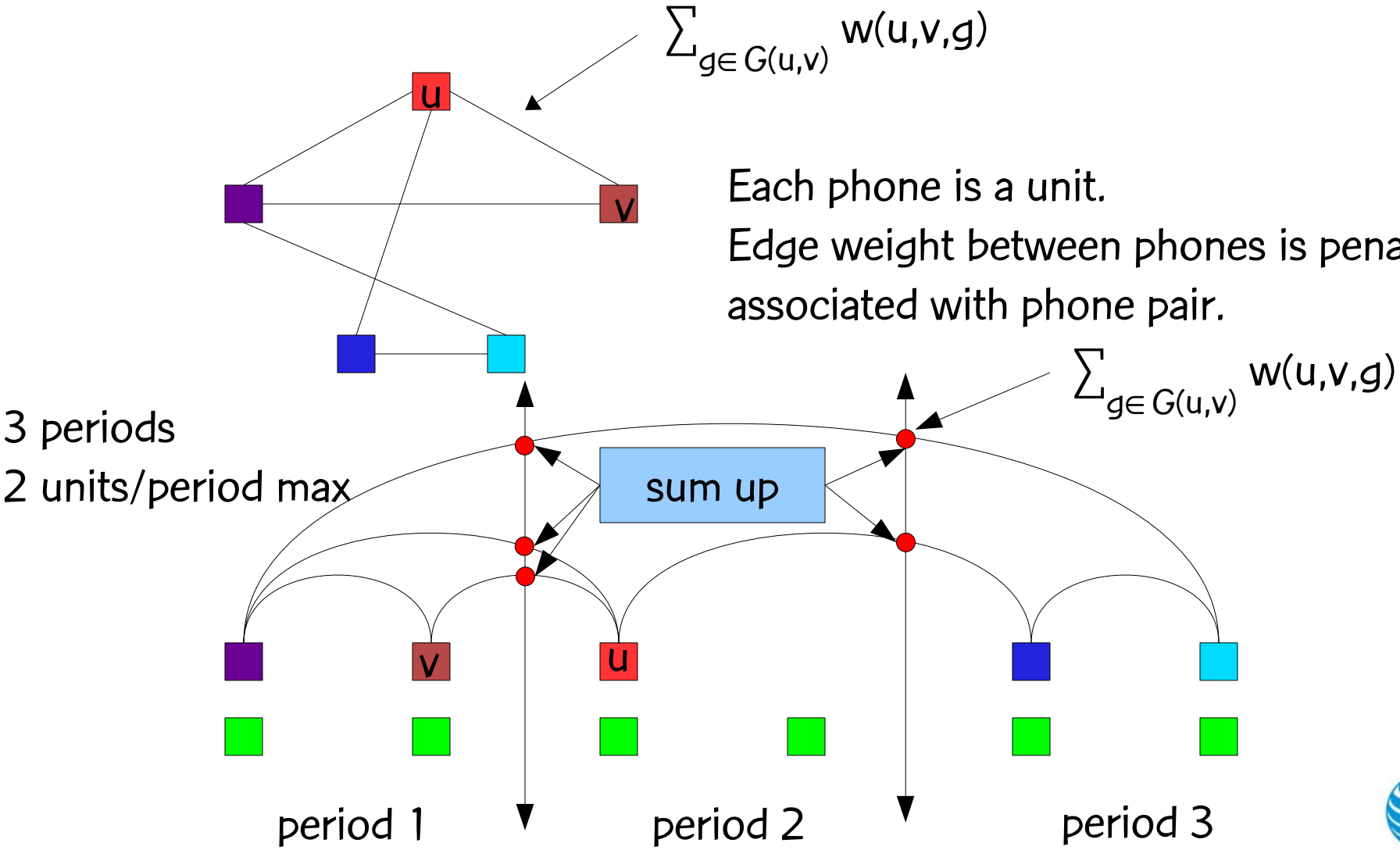
Each phone is a unit.
Edge weight between phones is penalty associated with phone pair.

$\sum_{g \in G(u,v)} w(u,v,g)$

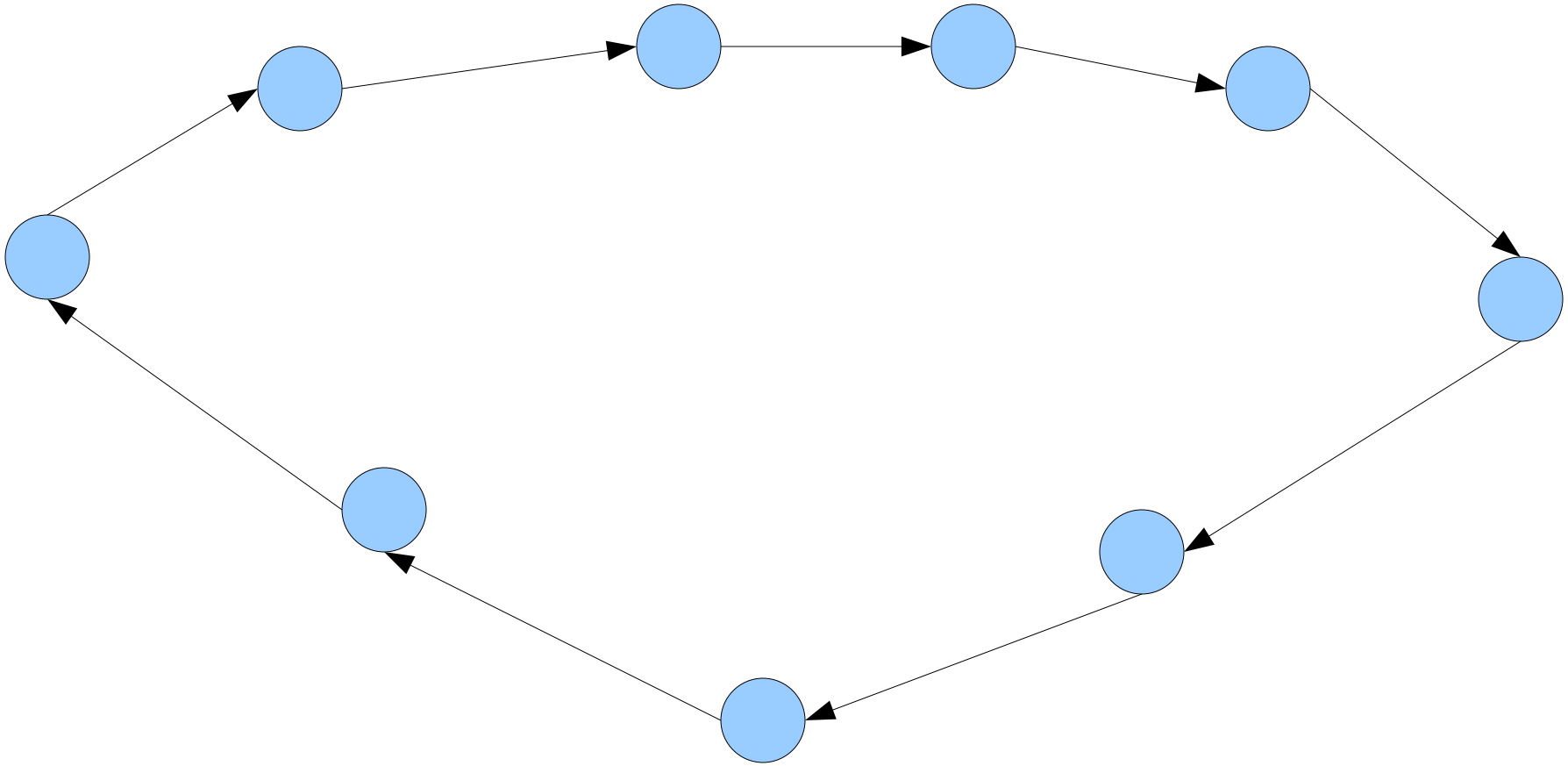
3 periods
2 units/period max



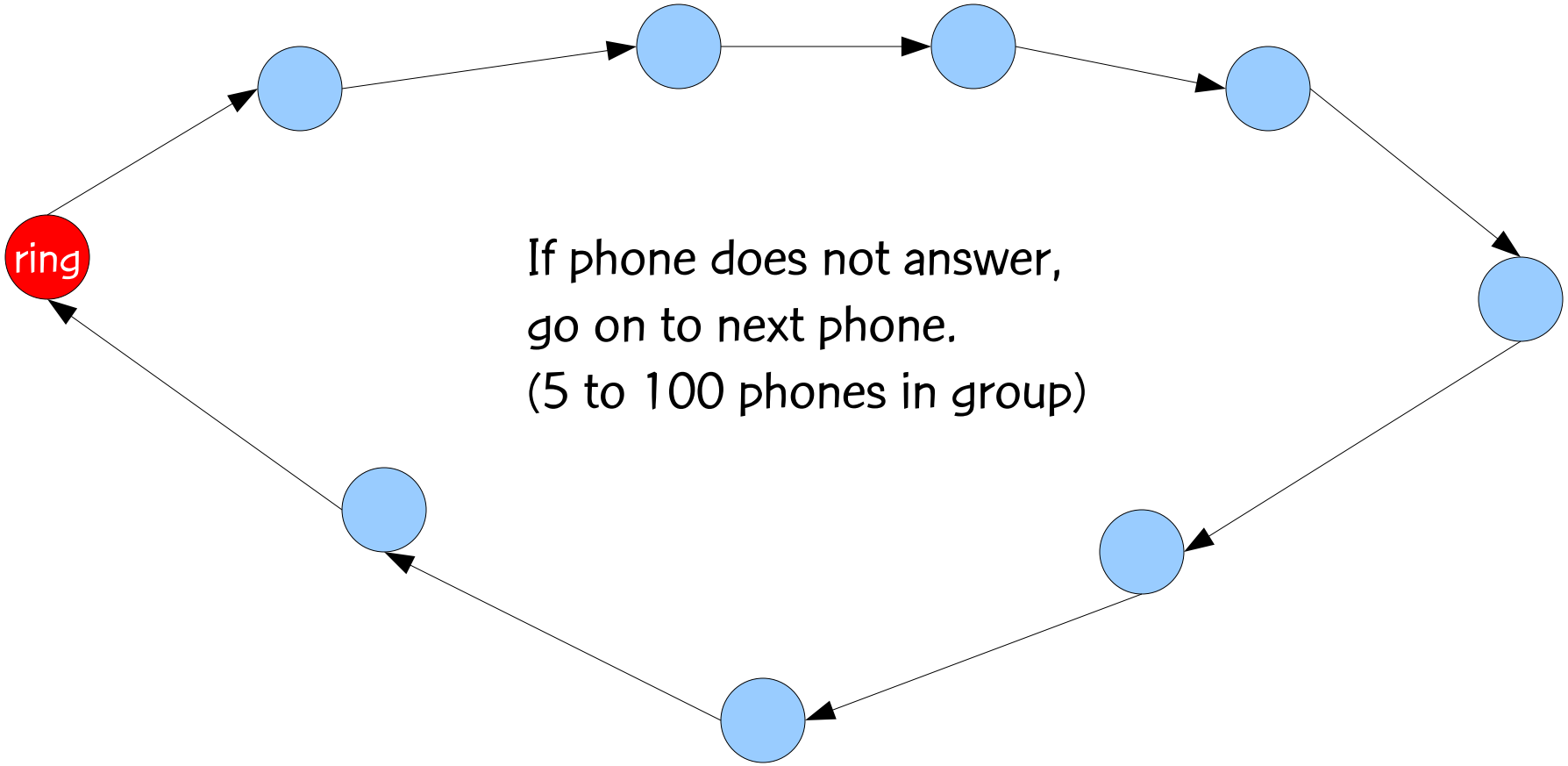
PBX telephone migration scheduling



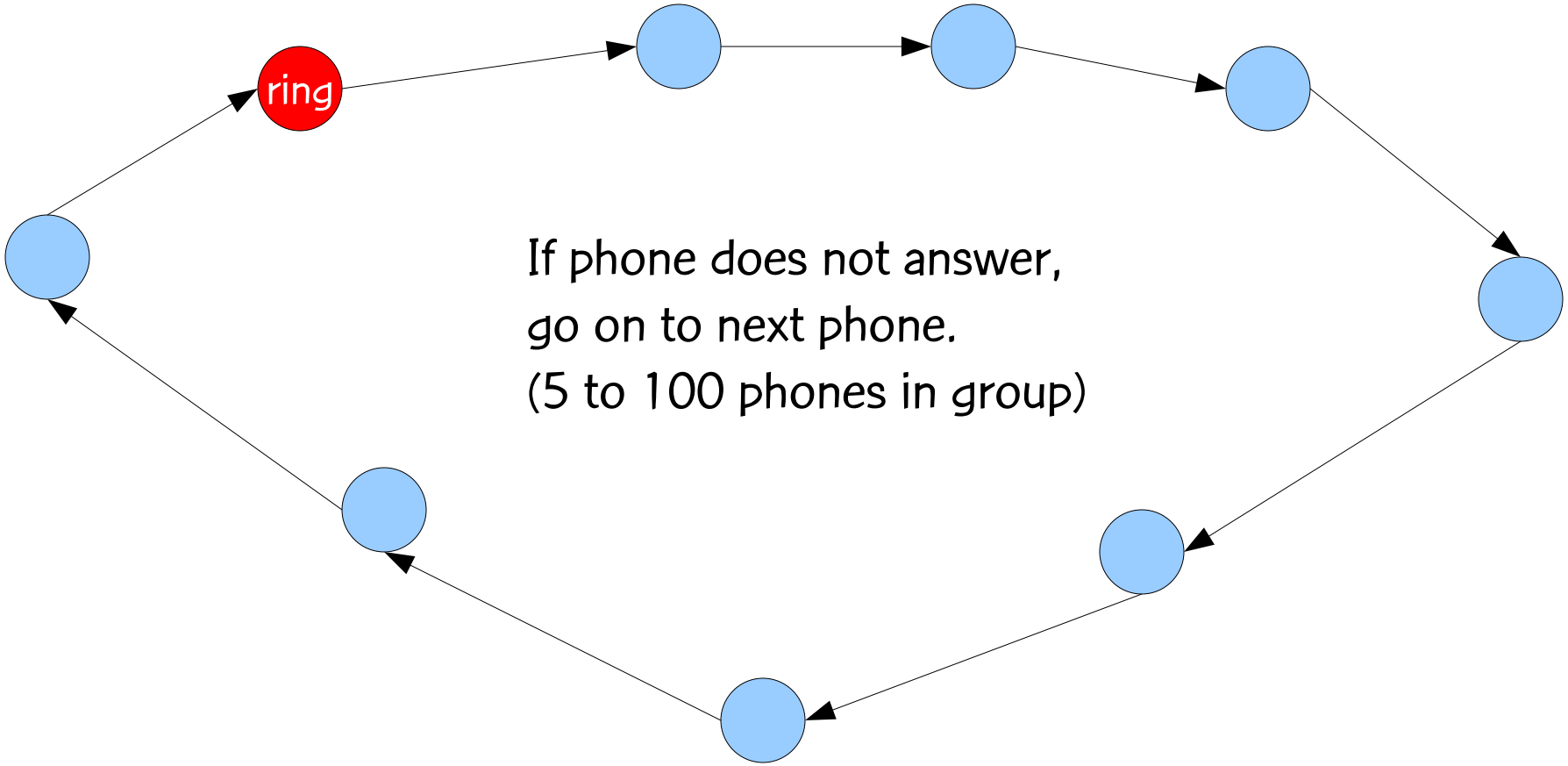
Multi-line hunt group



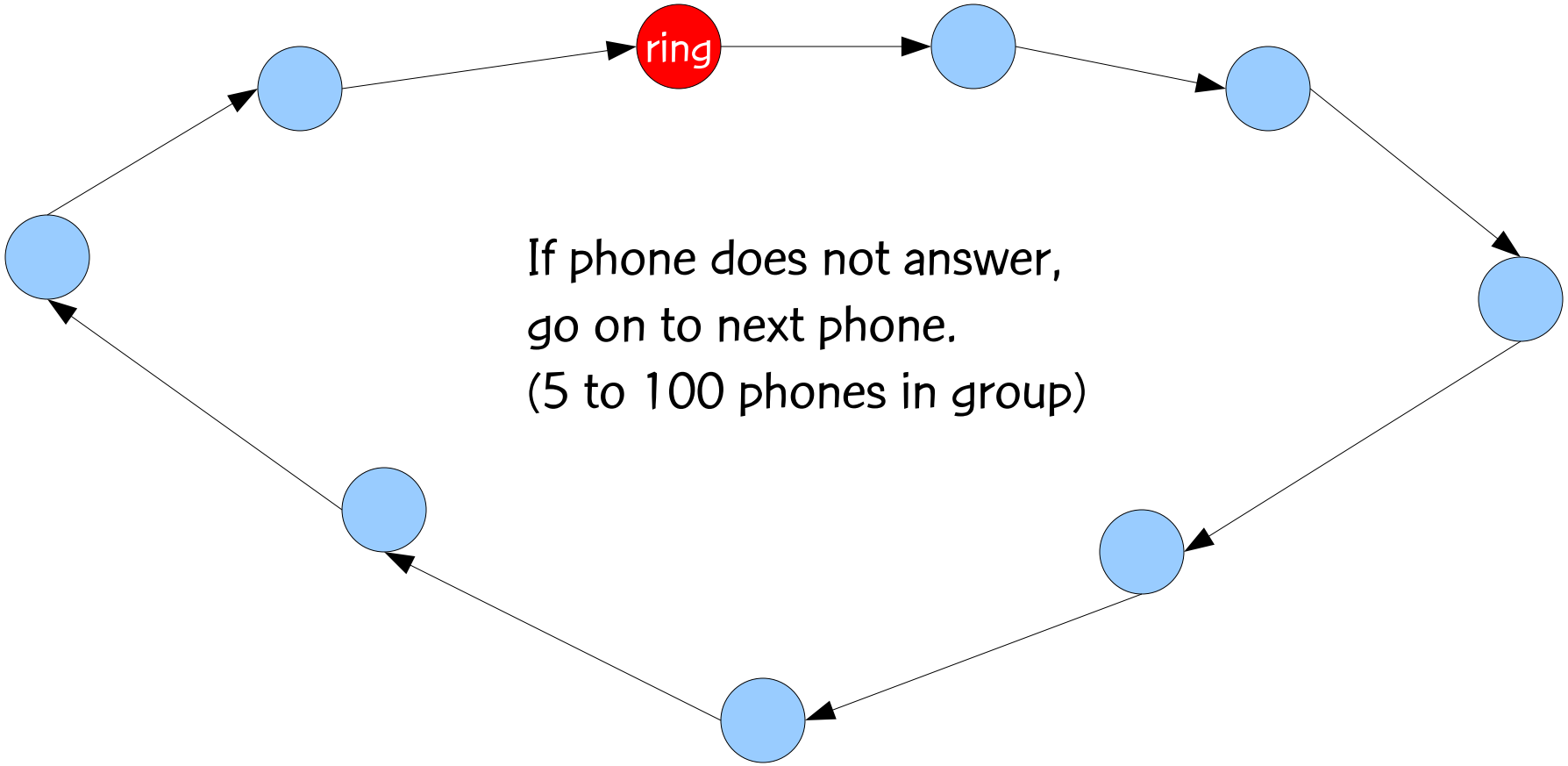
Multi-line hunt group



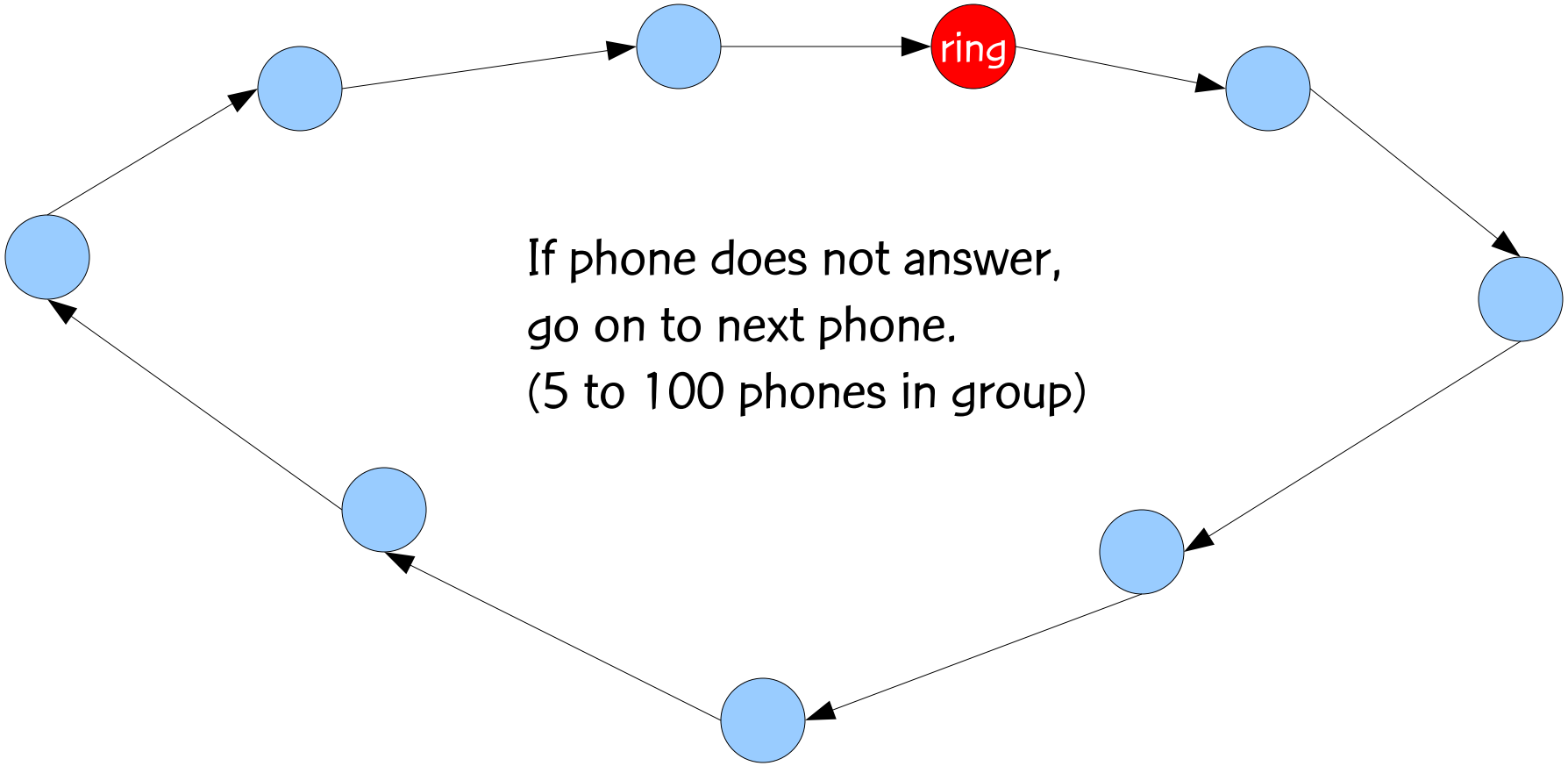
Multi-line hunt group



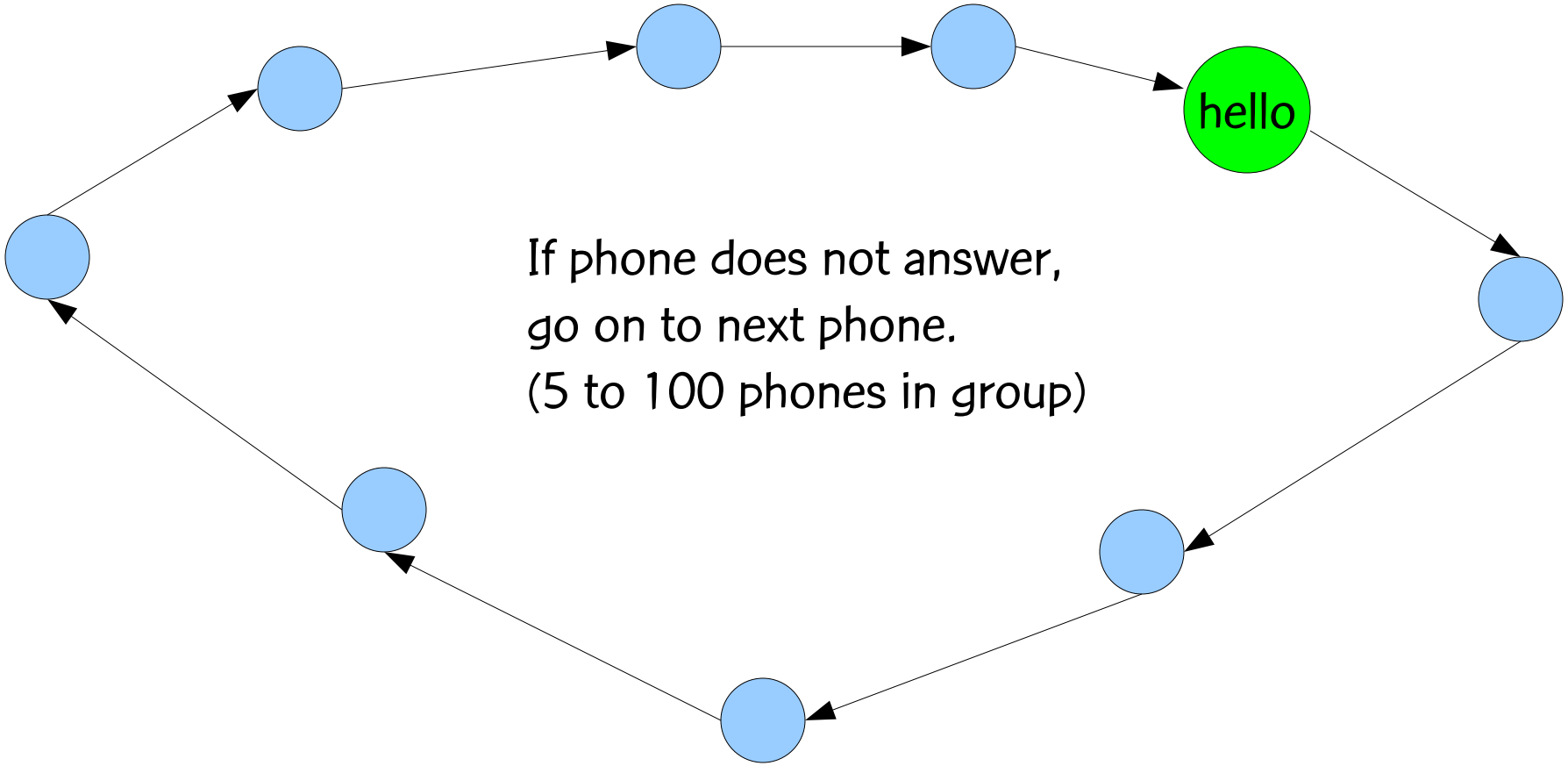
Multi-line hunt group



Multi-line hunt group

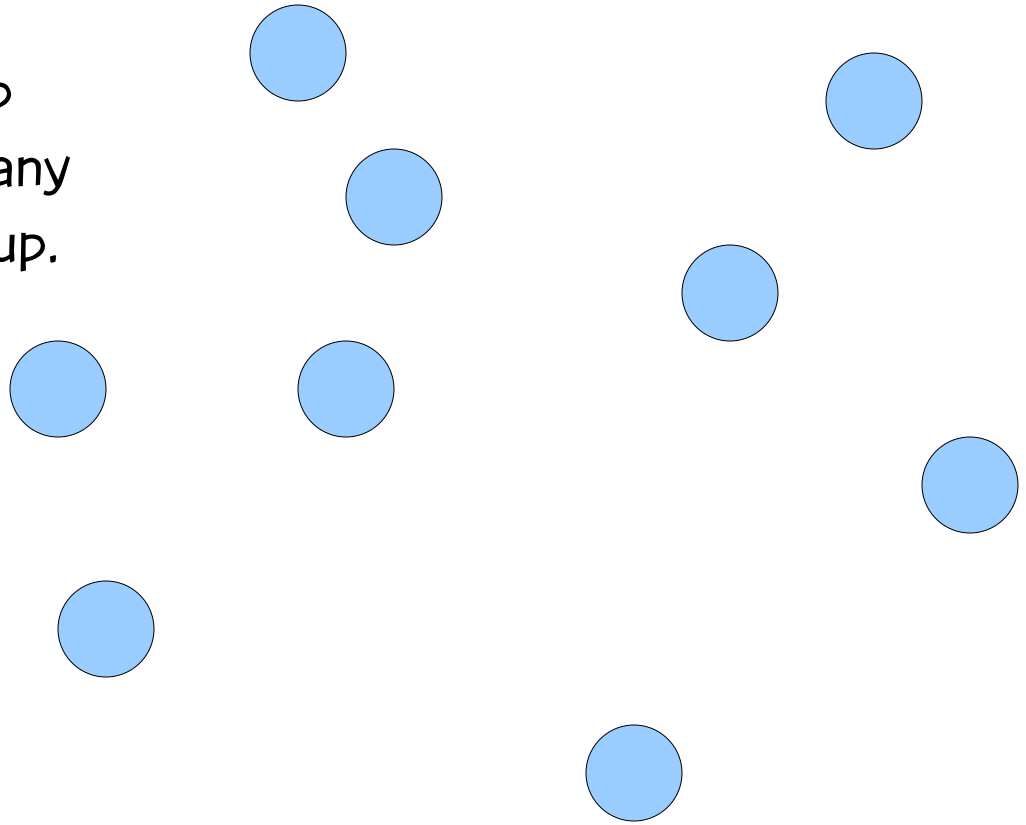


Multi-line hunt group



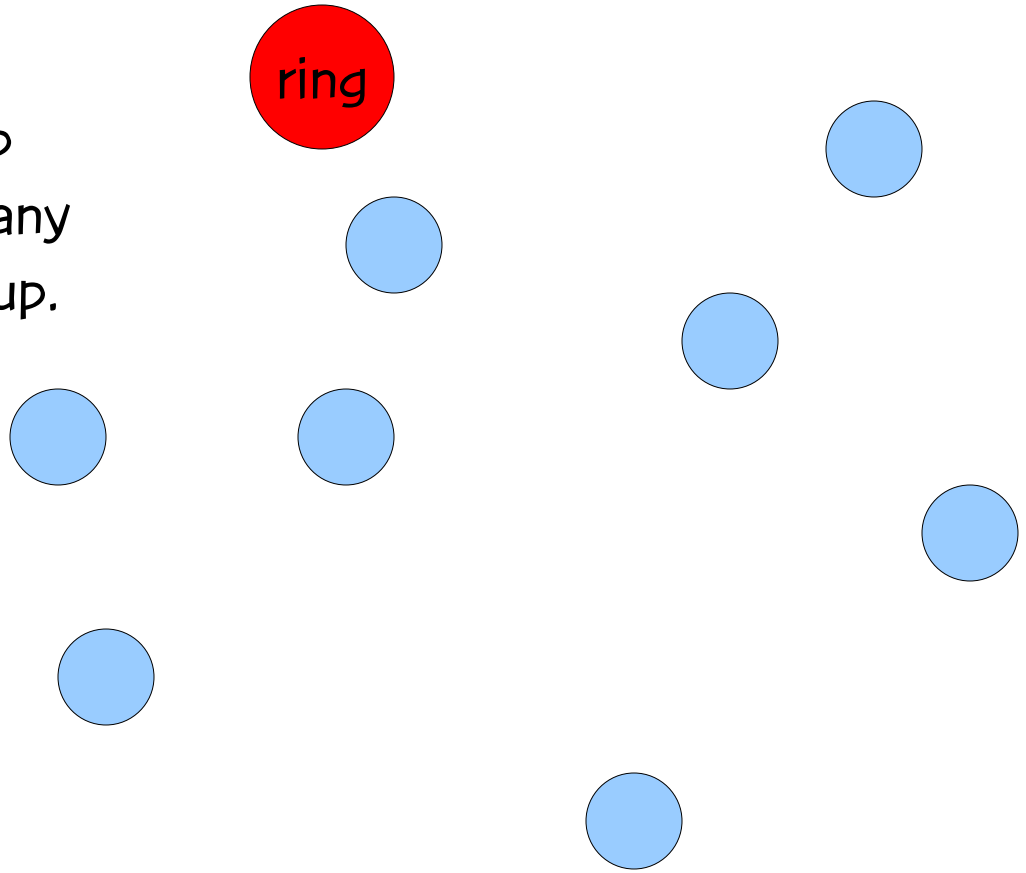
Call pickup (CPU)

Any phone in group can pickup call for any other phone in group.



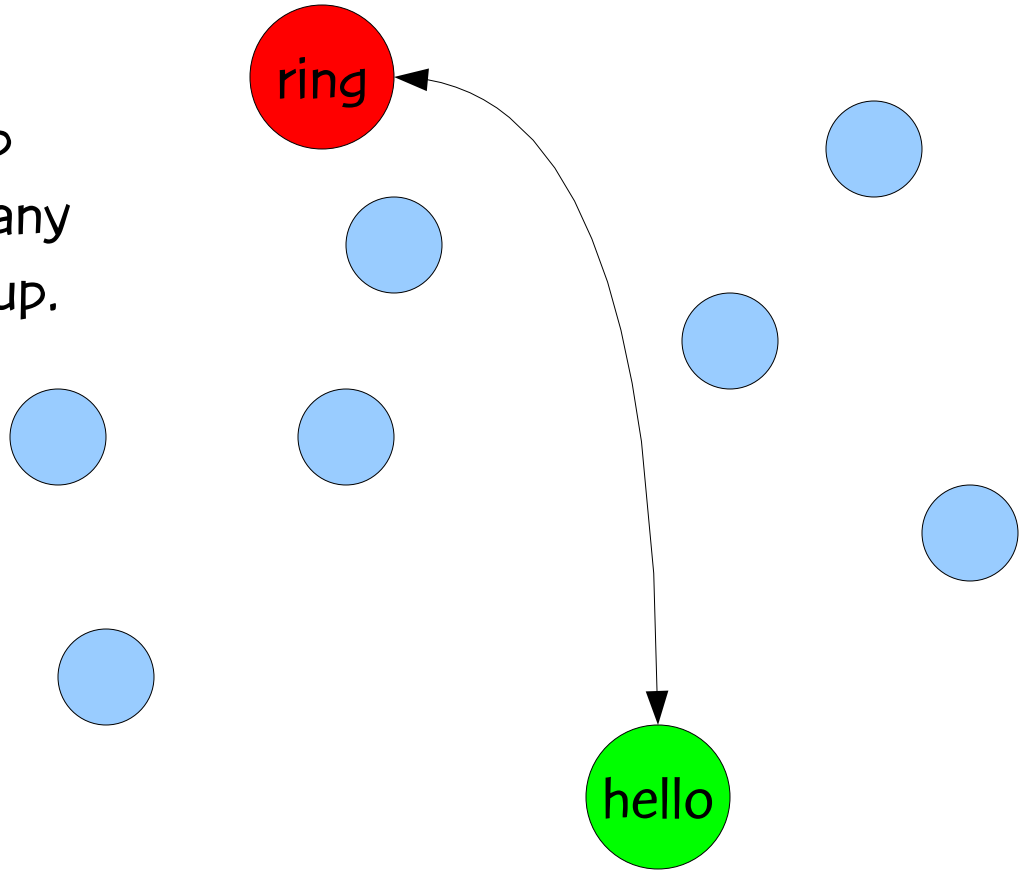
Call pickup (CPU)

Any phone in group can pickup call for any other phone in group.



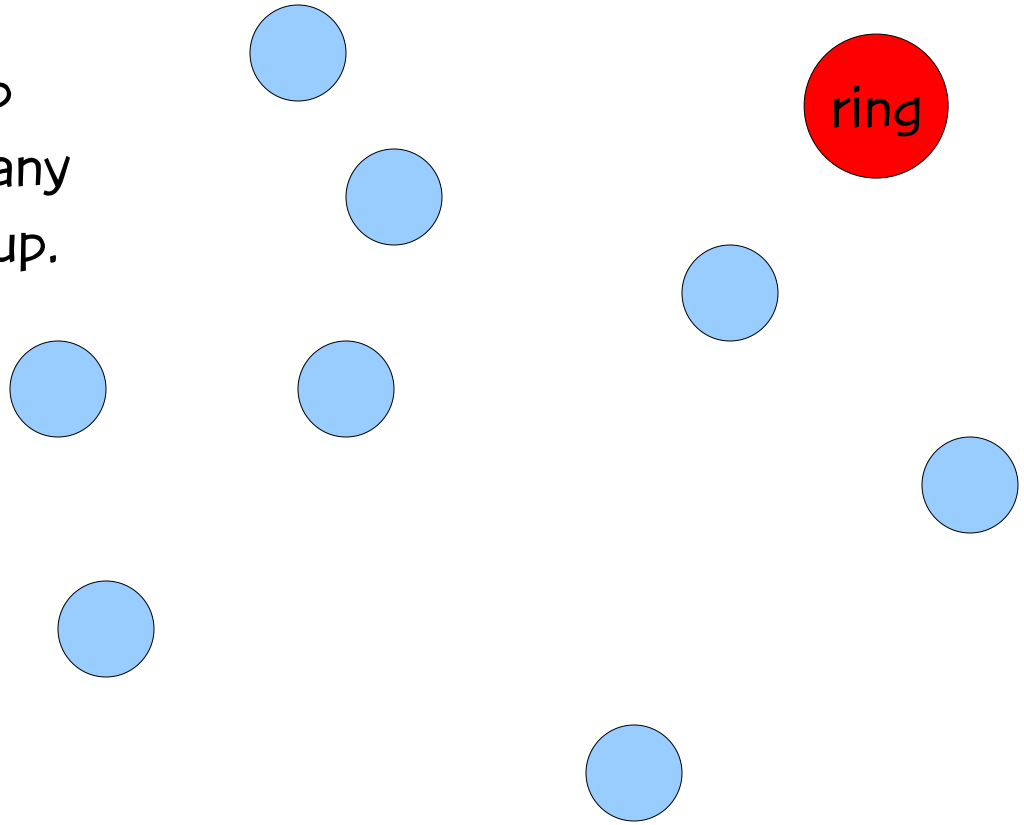
Call pickup (CPU)

Any phone in group can pickup call for any other phone in group.



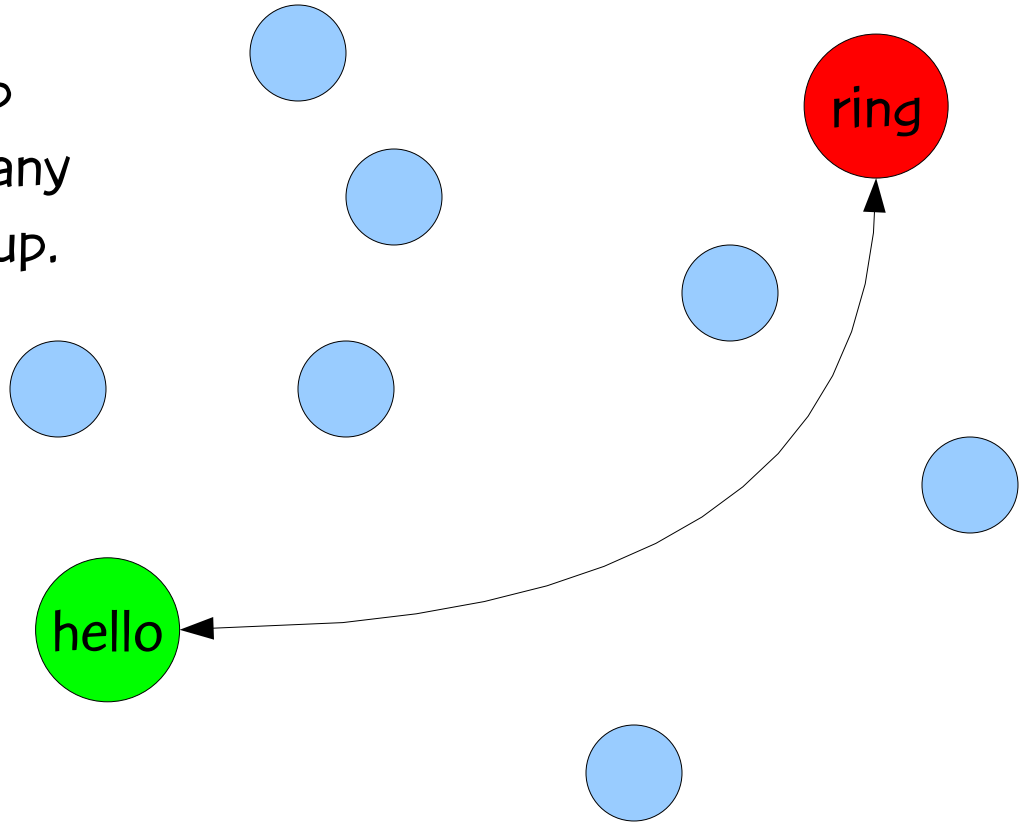
Call pickup (CPU)

Any phone in group can pickup call for any other phone in group.



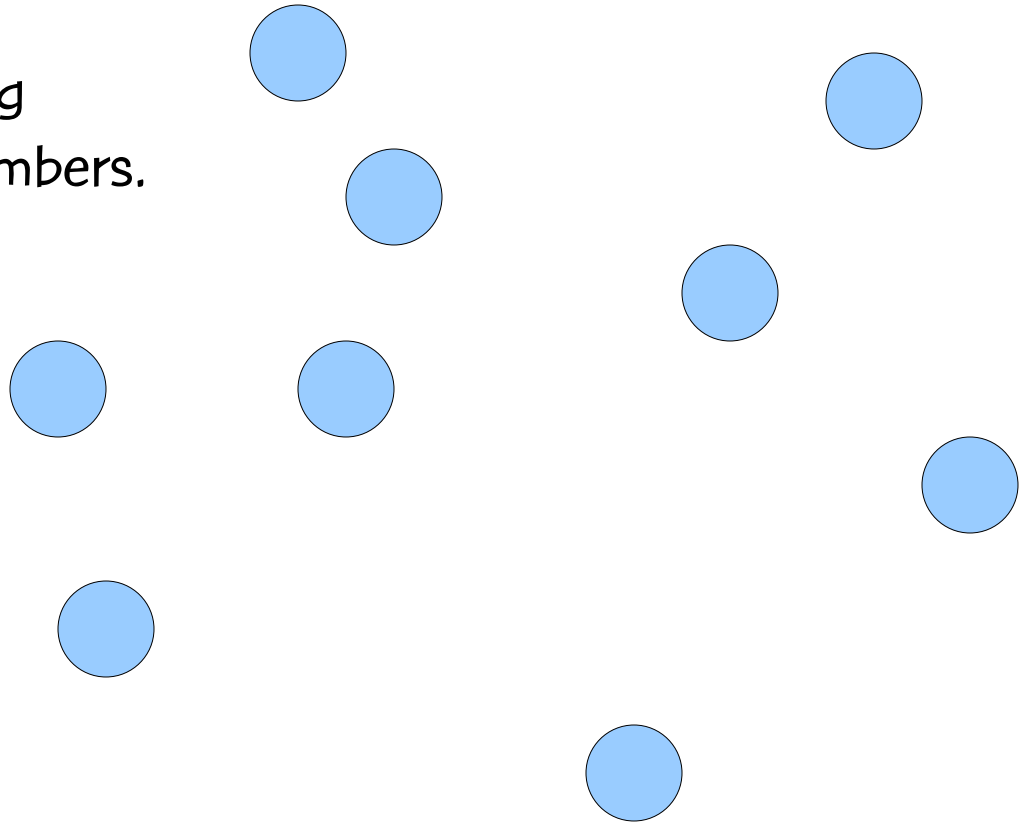
Call pickup (CPU)

Any phone in group can pickup call for any other phone in group.



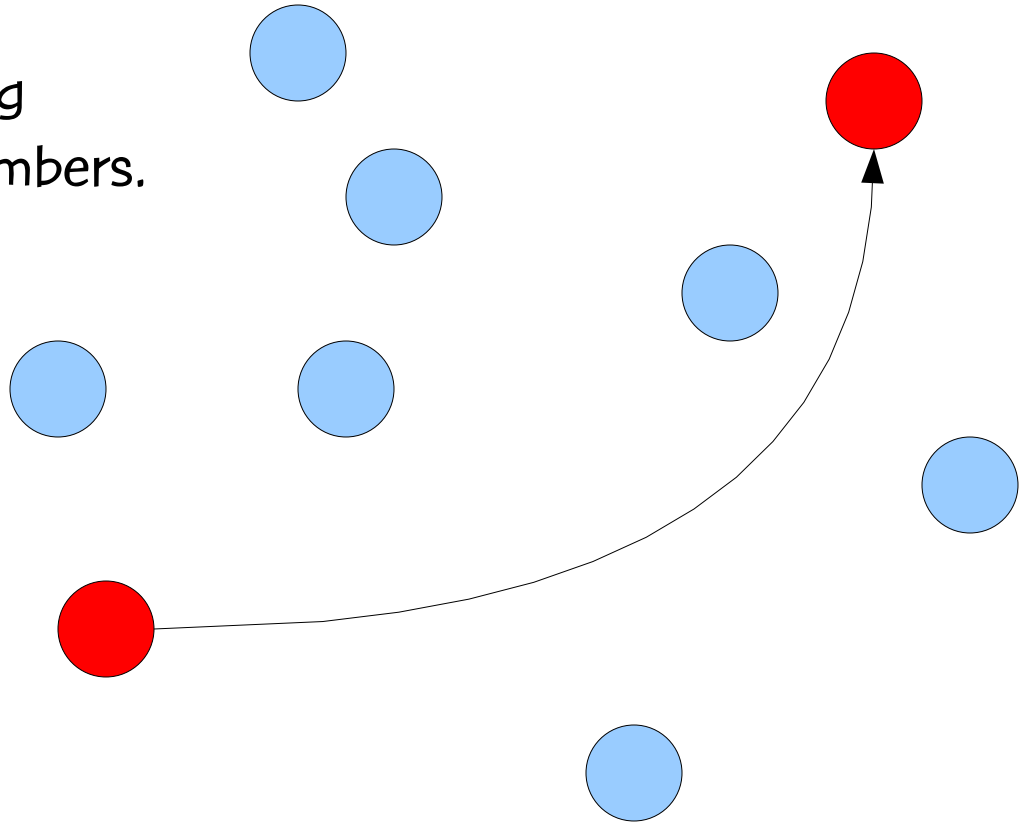
Intercomm (ICOM)

Allows speed dialing between group members.



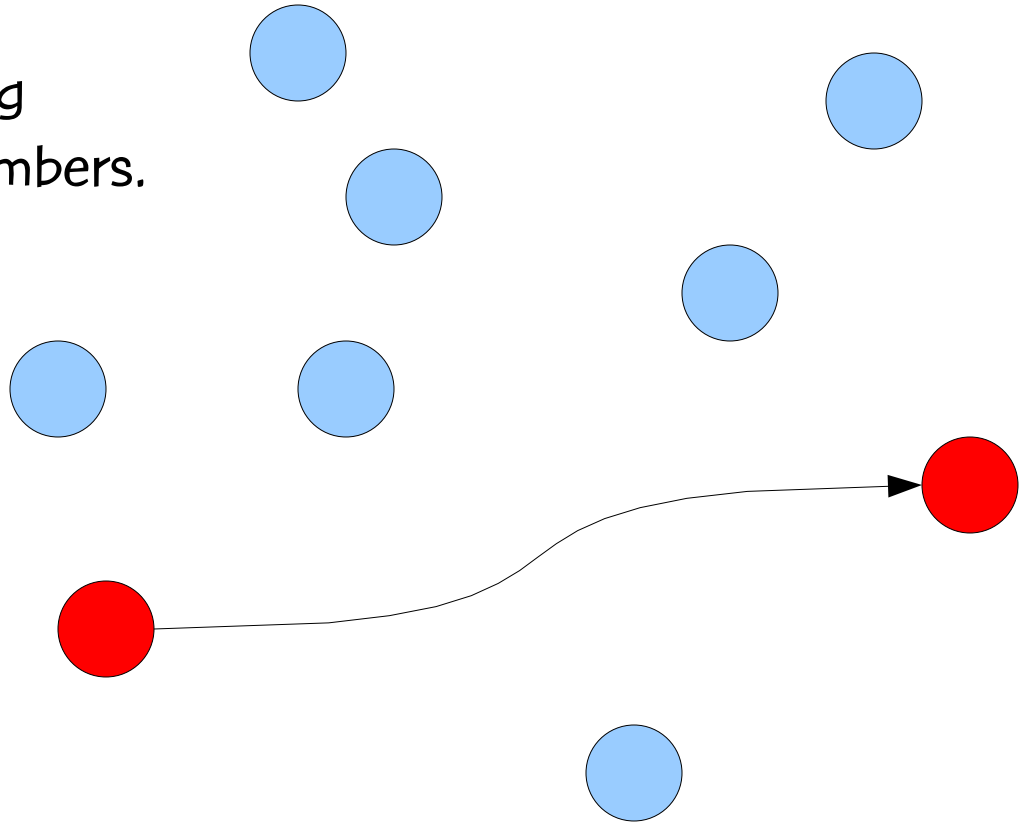
Intercomm (ICOM)

Allows speed dialing between group members.



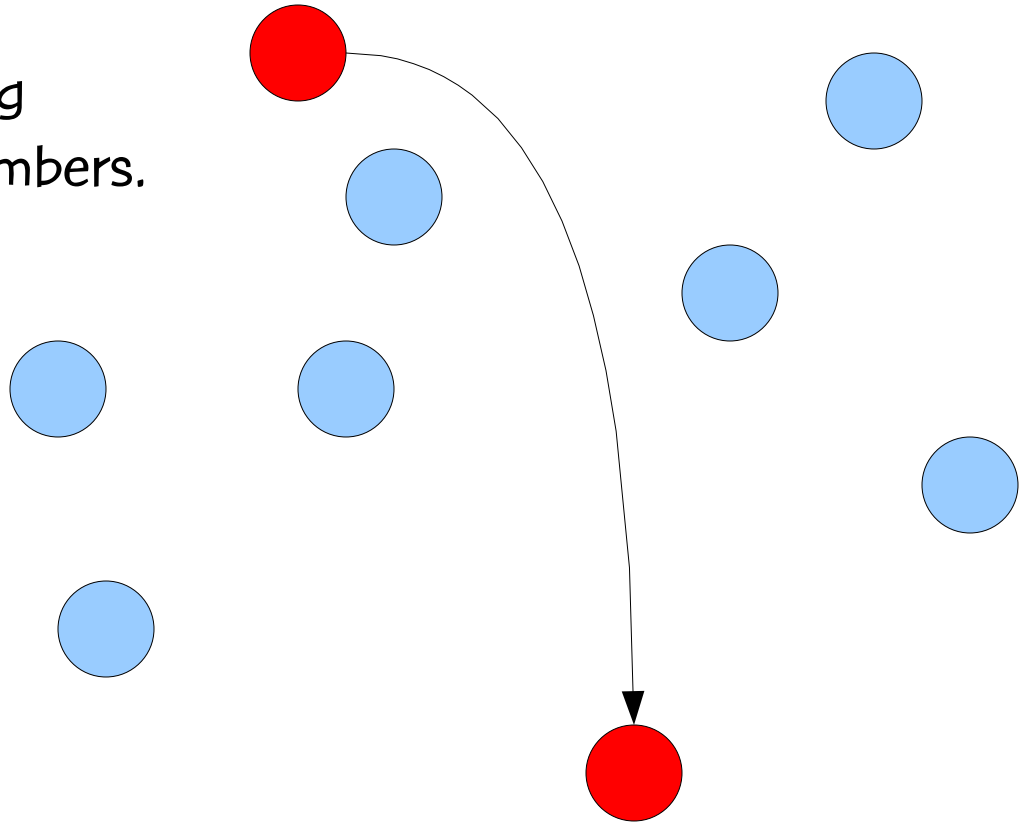
Intercomm (ICOM)

Allows speed dialing between group members.

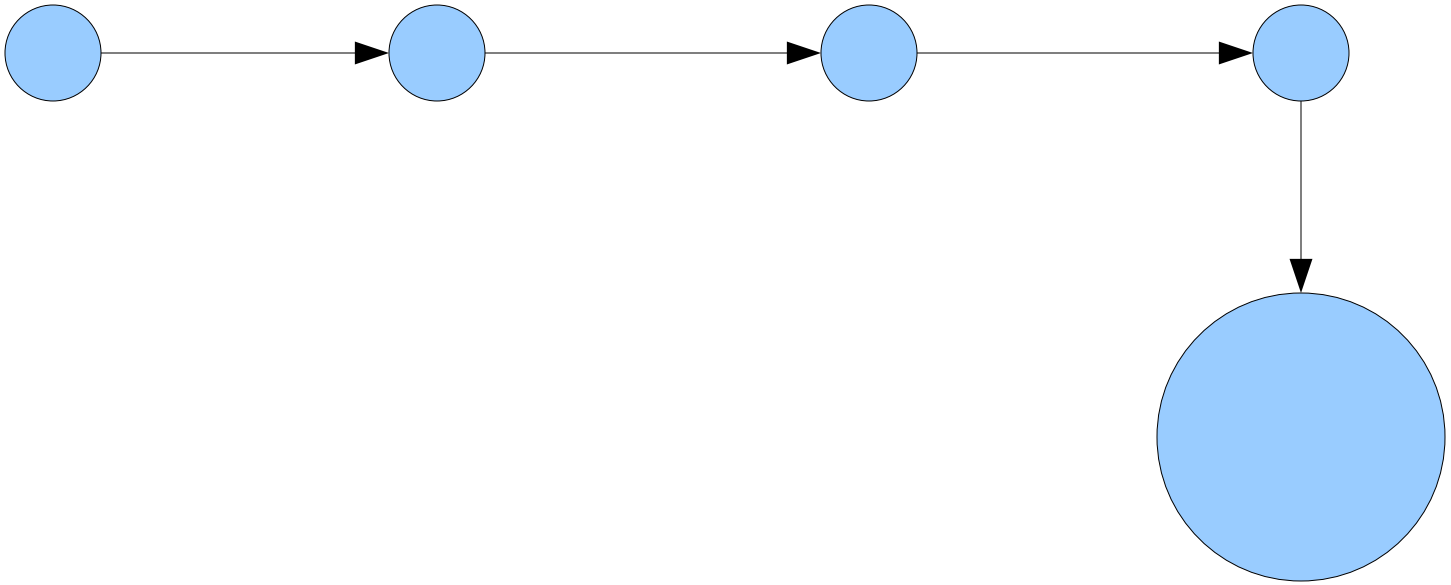


Intercomm (ICOM)

Allows speed dialing between group members.

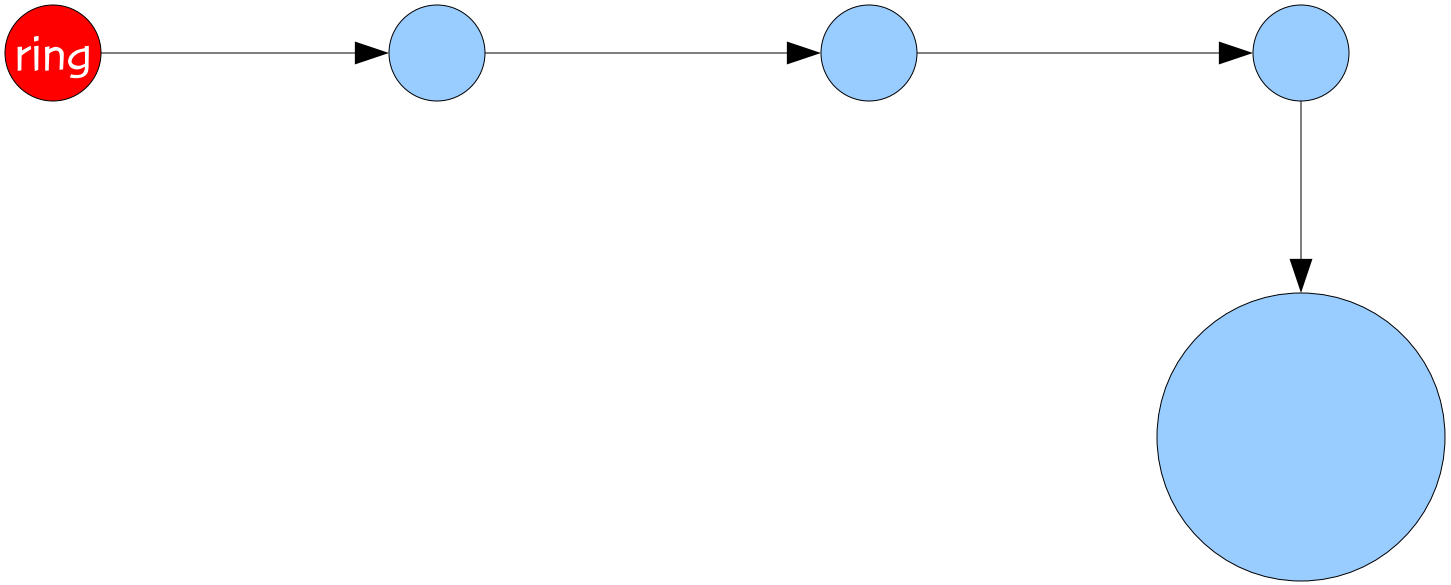


Series completion



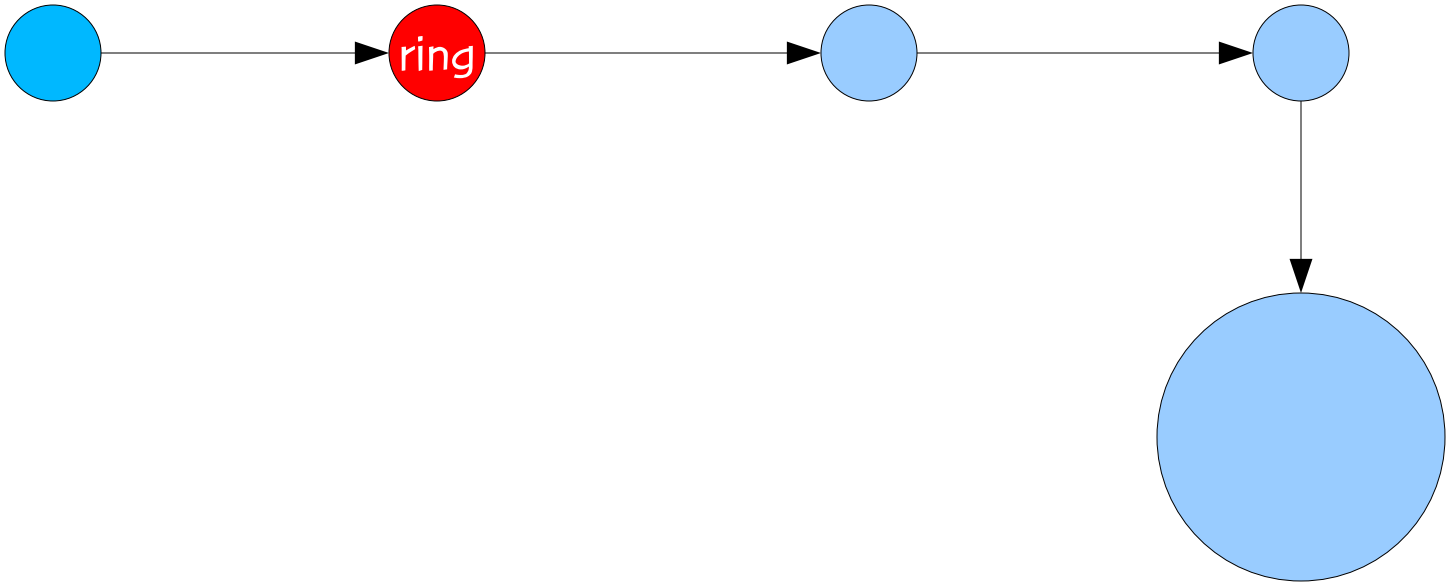
Series completion

If call not answered ...



Series completion

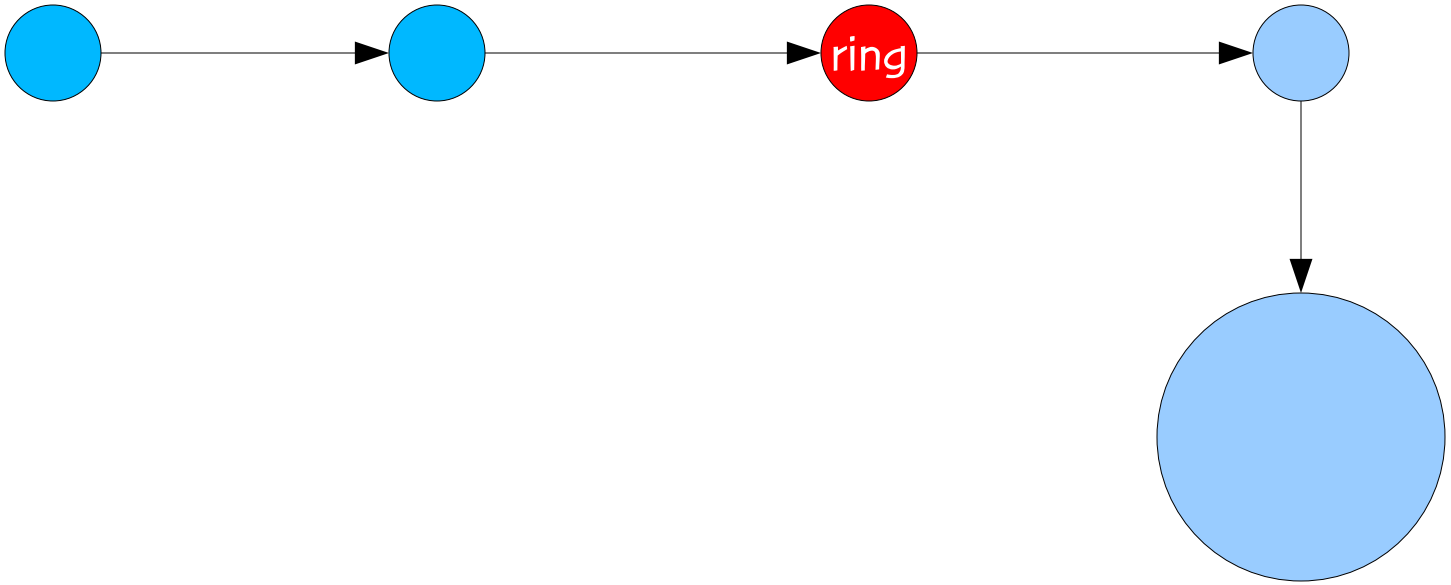
If call not answered, it moves
to .....next in series.



Series completion

If call not answered, it moves

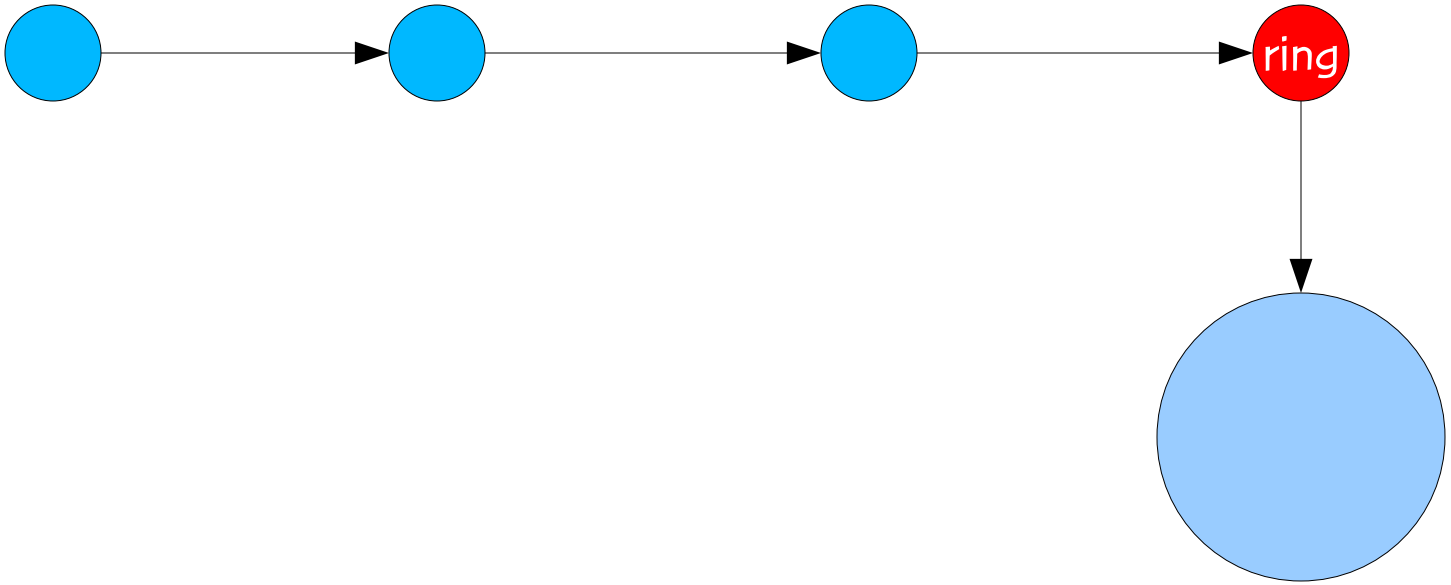
to ..... next in series.



Series completion

If call not answered, it moves

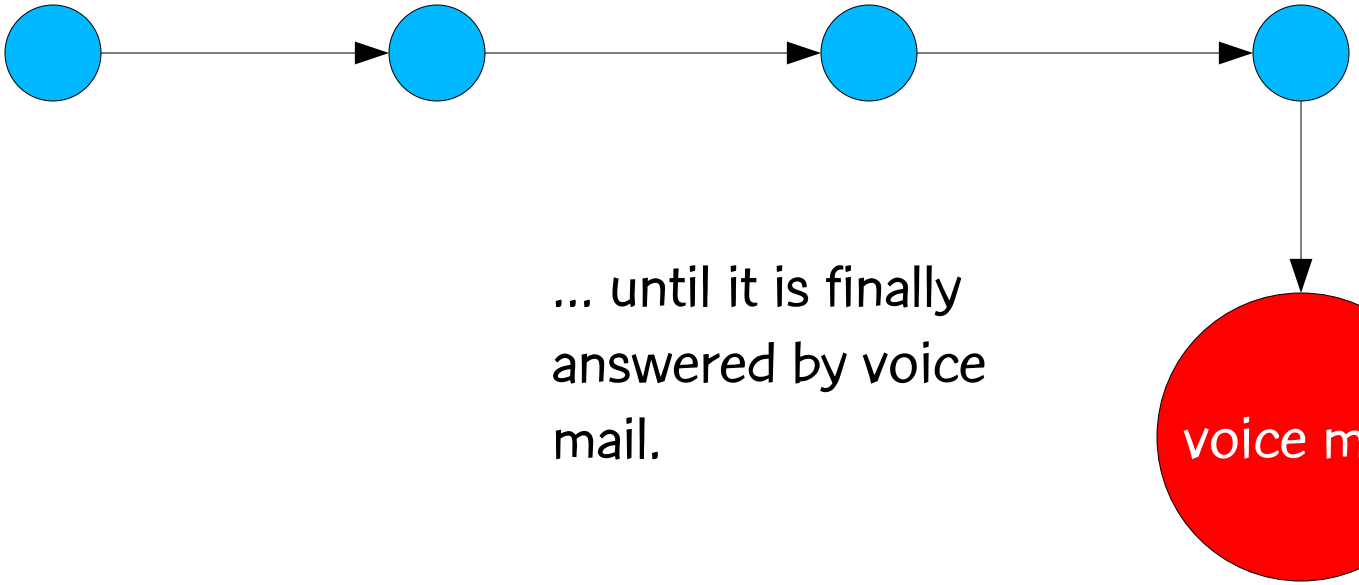
to ..... next in series.



Series completion

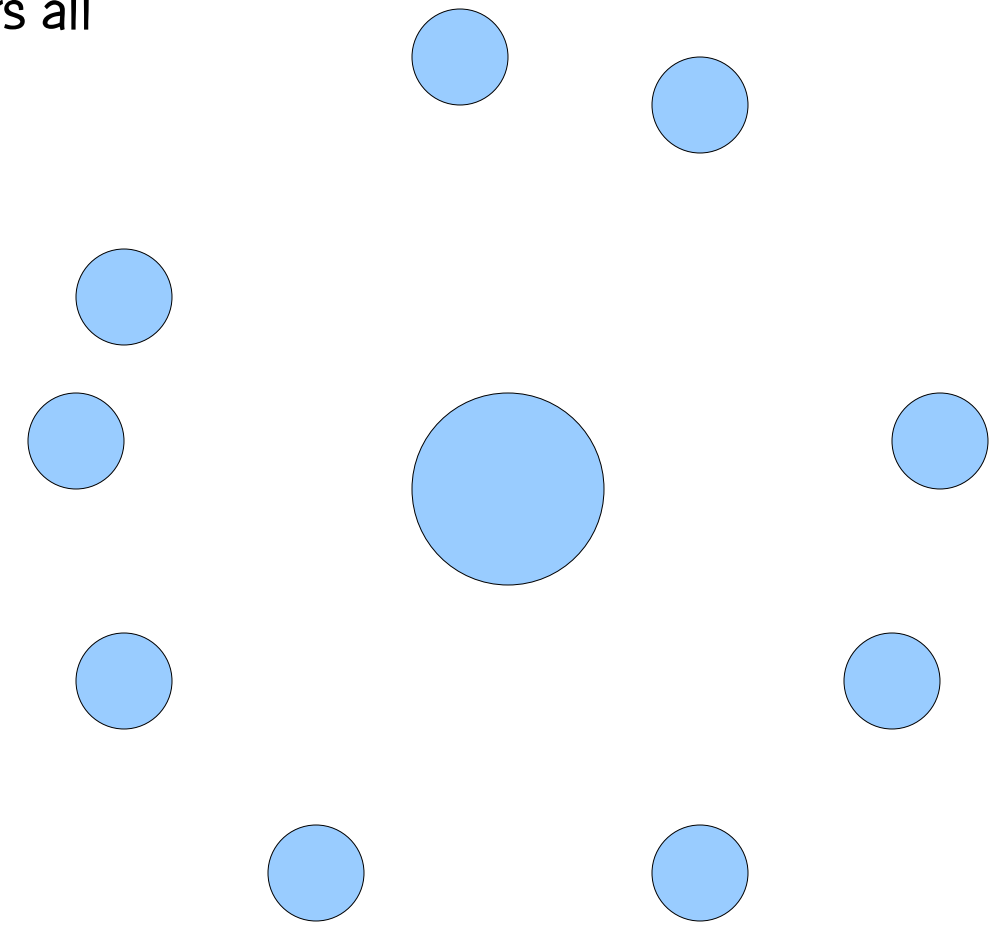
If call not answered, it moves

to ..... next in series ...



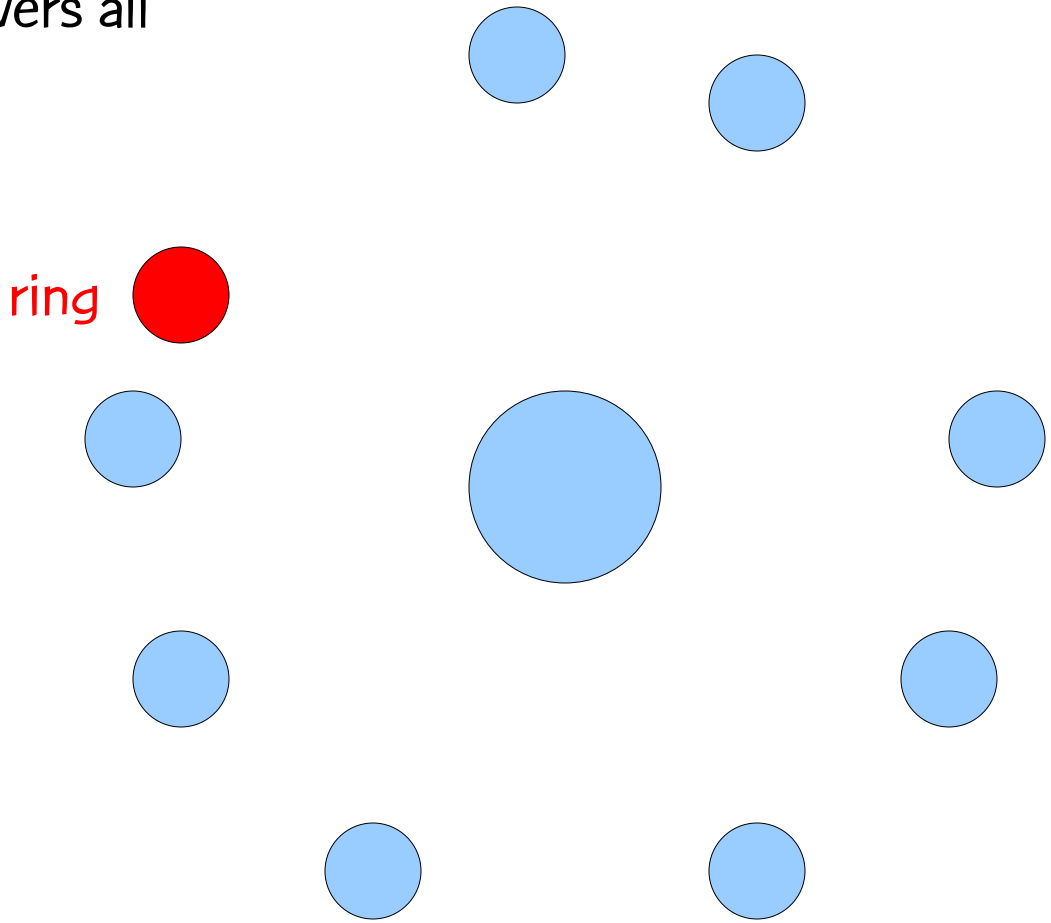
Shared TN

Assistant answers all calls to group.



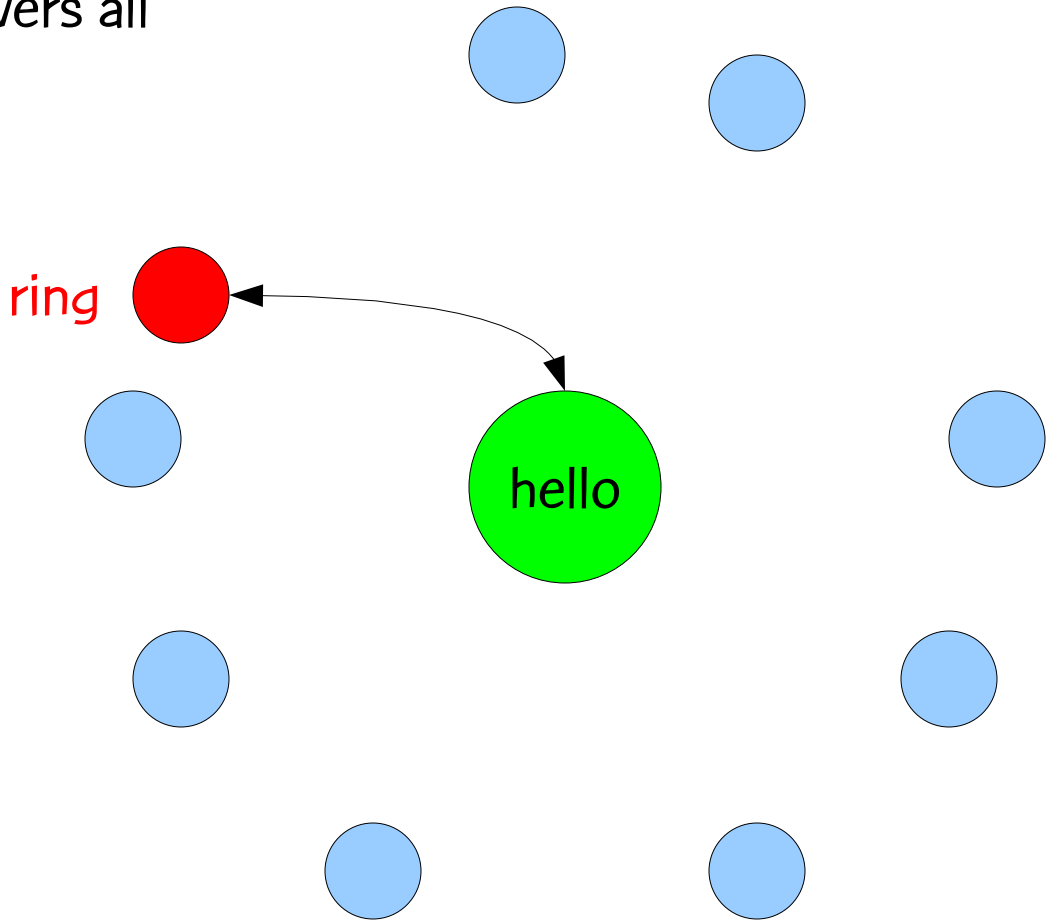
Shared TN

Assistant answers all calls to group.



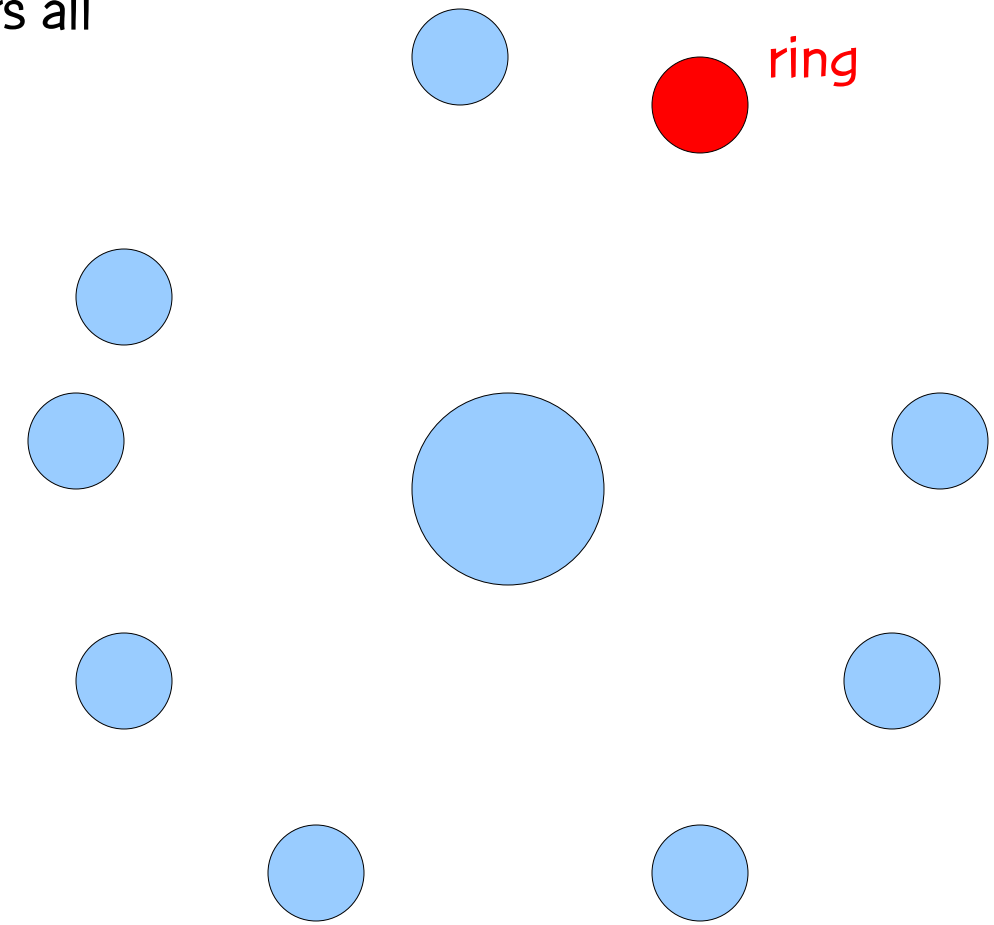
Shared TN

Assistant answers all calls to group.



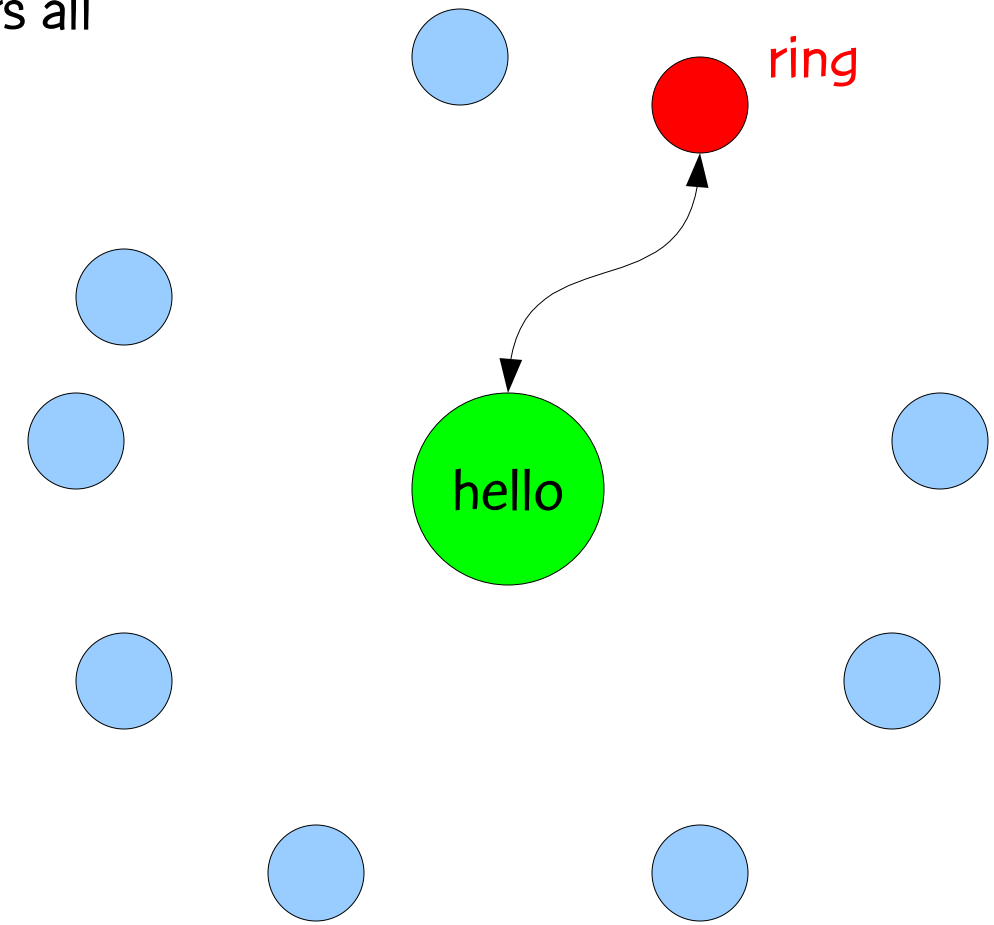
Shared TN

Assistant answers all calls to group.



Shared TN

Assistant answers all calls to group.



Real-world example

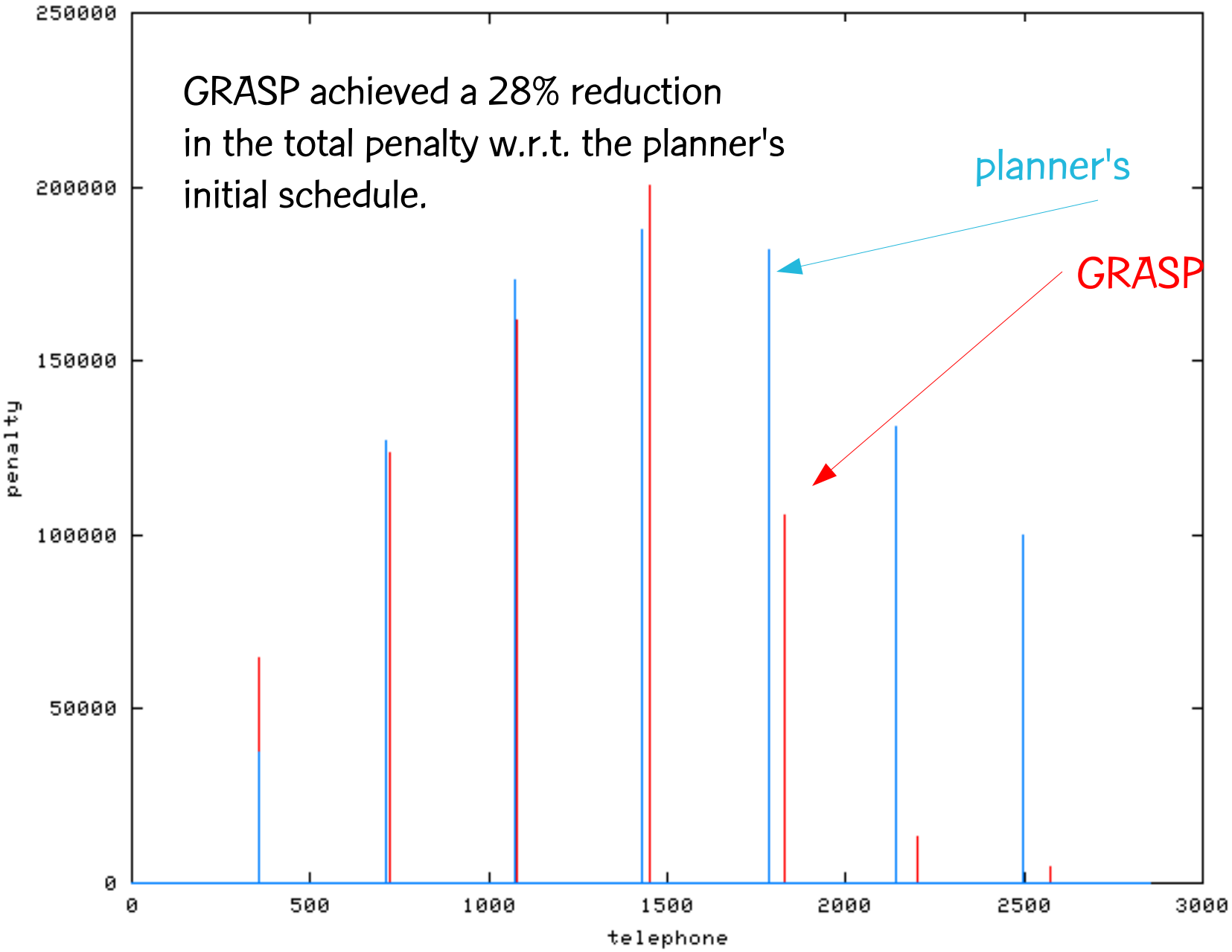
- AT&T sold new switch to large NYC-based investment bank.
- Post sales contacted us about improving a tentative schedule they had produced.
- Objective was to minimize business disruption during migration to new switch.
- Problem size:
  - 2855 phone numbers, 397 groups.
  - At most 375 phones could be moved in a period: 8 periods.

Real-world example

- Objective: Minimize disruption by minimizing total migration penalty
  - Groups and Penalties:
    - Multi-Line Hunt : 10
    - Call Pickup : 4
    - Intercomm : 3
    - Series Completion: 2
    - Shared TN : 1

Real-world example

- Objective: Minimize disruption by minimizing total migration penalty
- Groups and Penalties:
  - Multi-Line Hunt : 10
  - Call Pickup : 4
  - Intercomm : 3
  - Series Completion: 2
  - Shared TN : 1



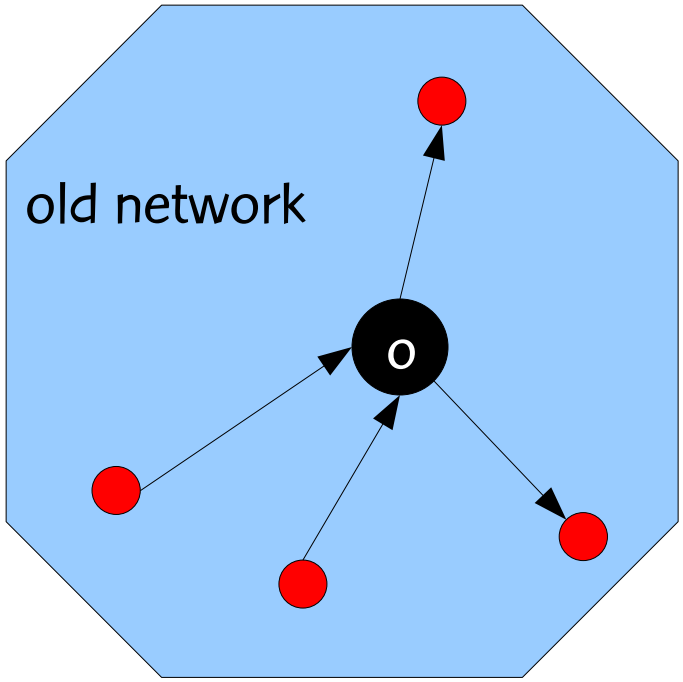
at&t
Your world. Delivered.

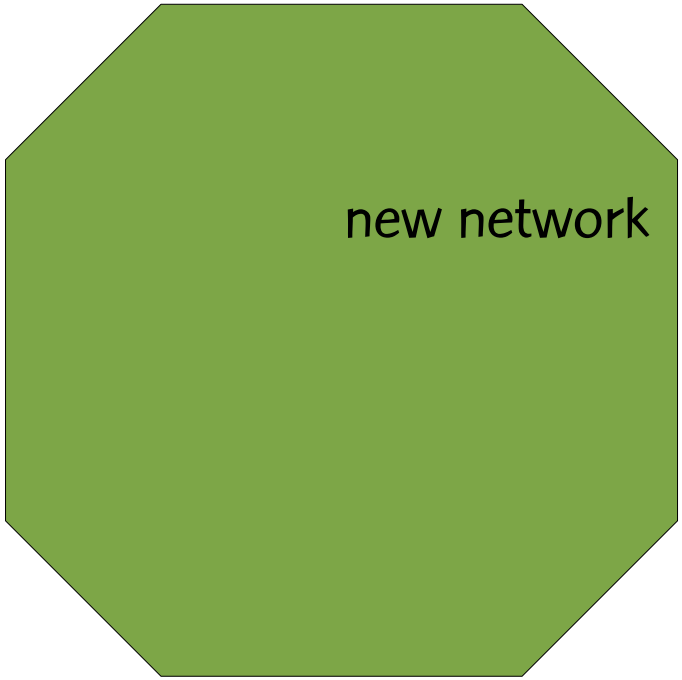
Network traffic migration scheduling

Network traffic migration scheduling

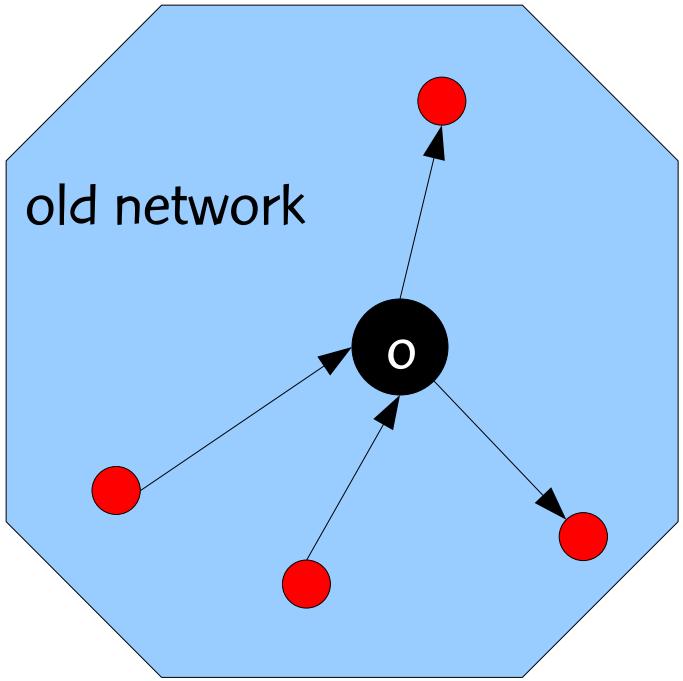
- Traffic from outdated telecommunications network is to be migrated to a new network.
  - e.g. phone traffic is to migrate from 4ESS switch-based network to IP router-based network.
- Nodes in old network are decommissioned, one at a time, and all traffic originating or terminating at the node is moved to a specific node in the new network.

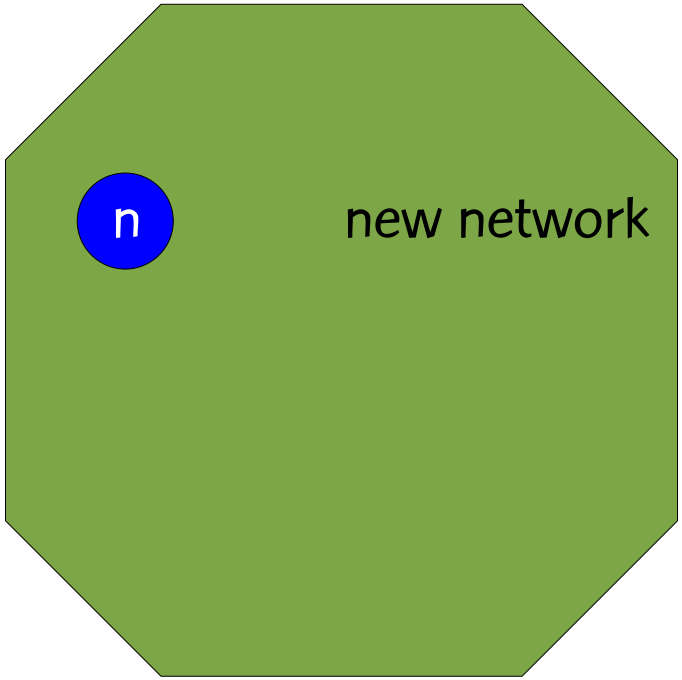
Node decommissioning



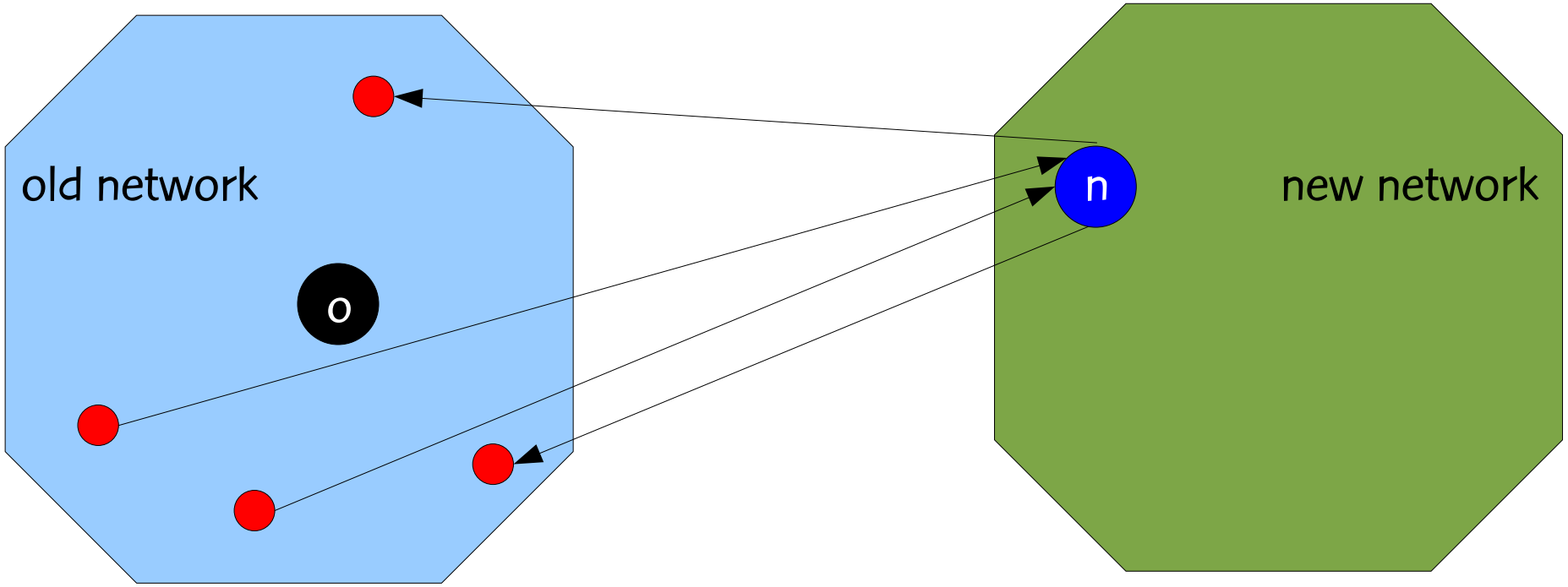


Node decommissioning

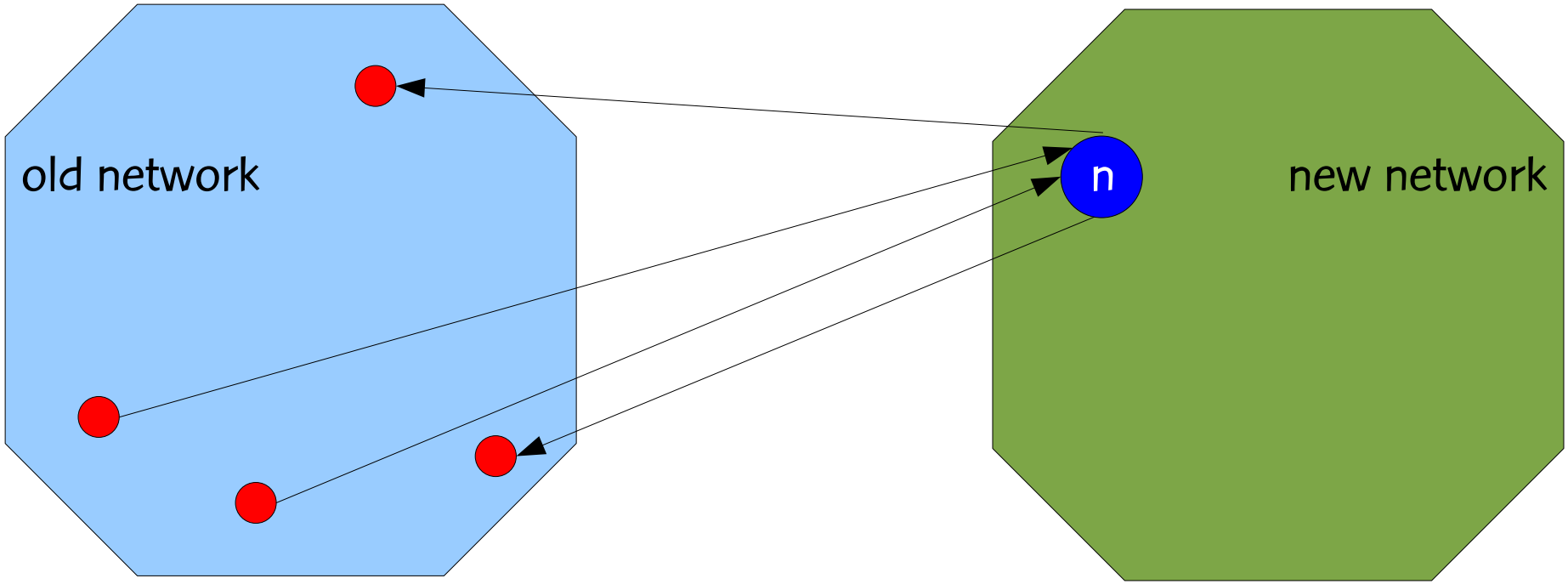




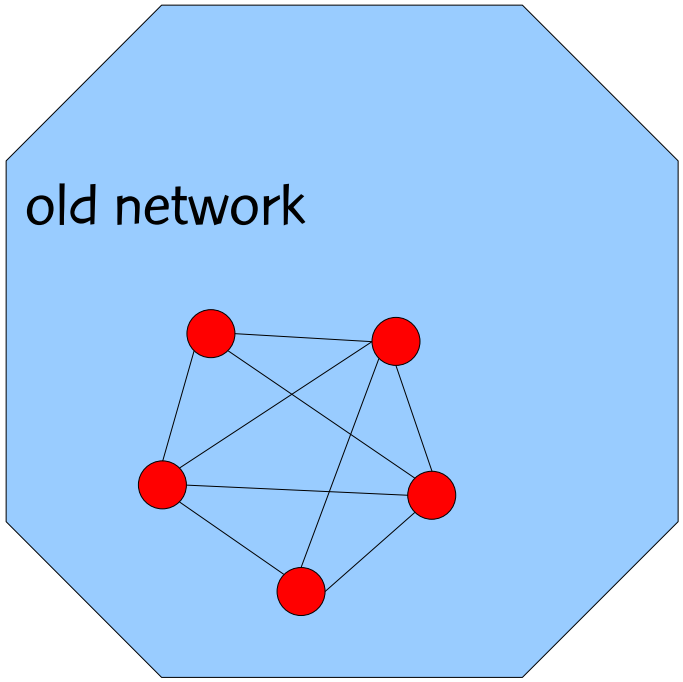
Node decommissioning

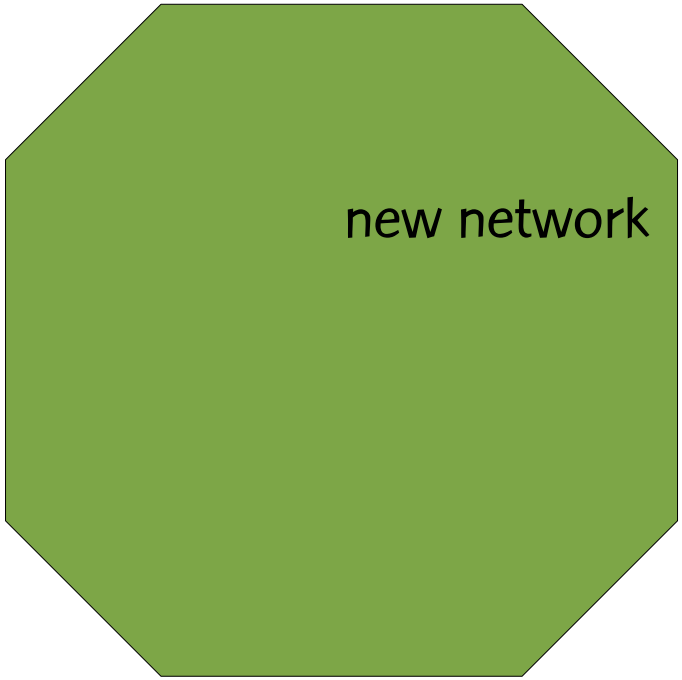


Node decommissioning



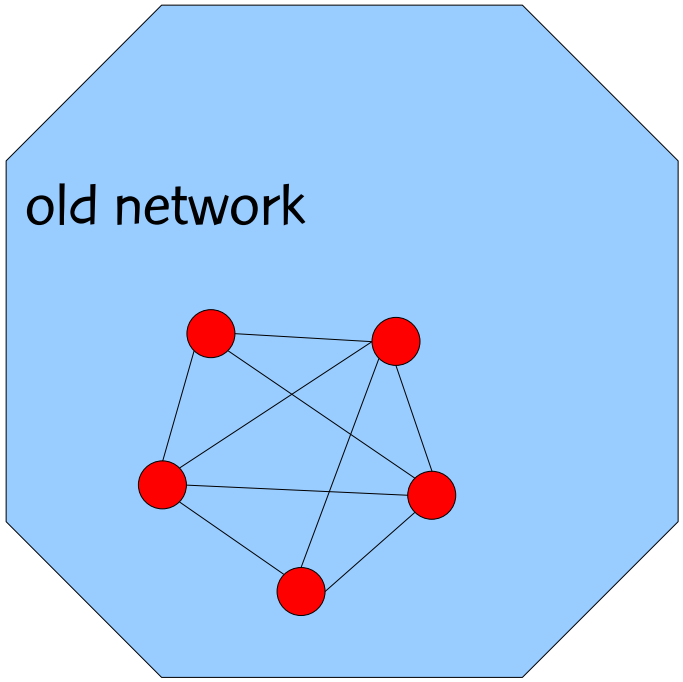
After partial decommissioning of nodes

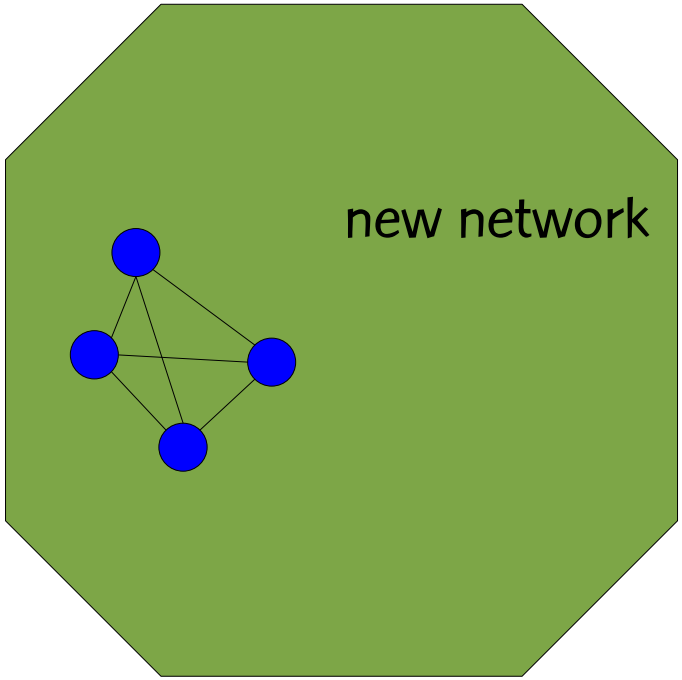




traffic in old network

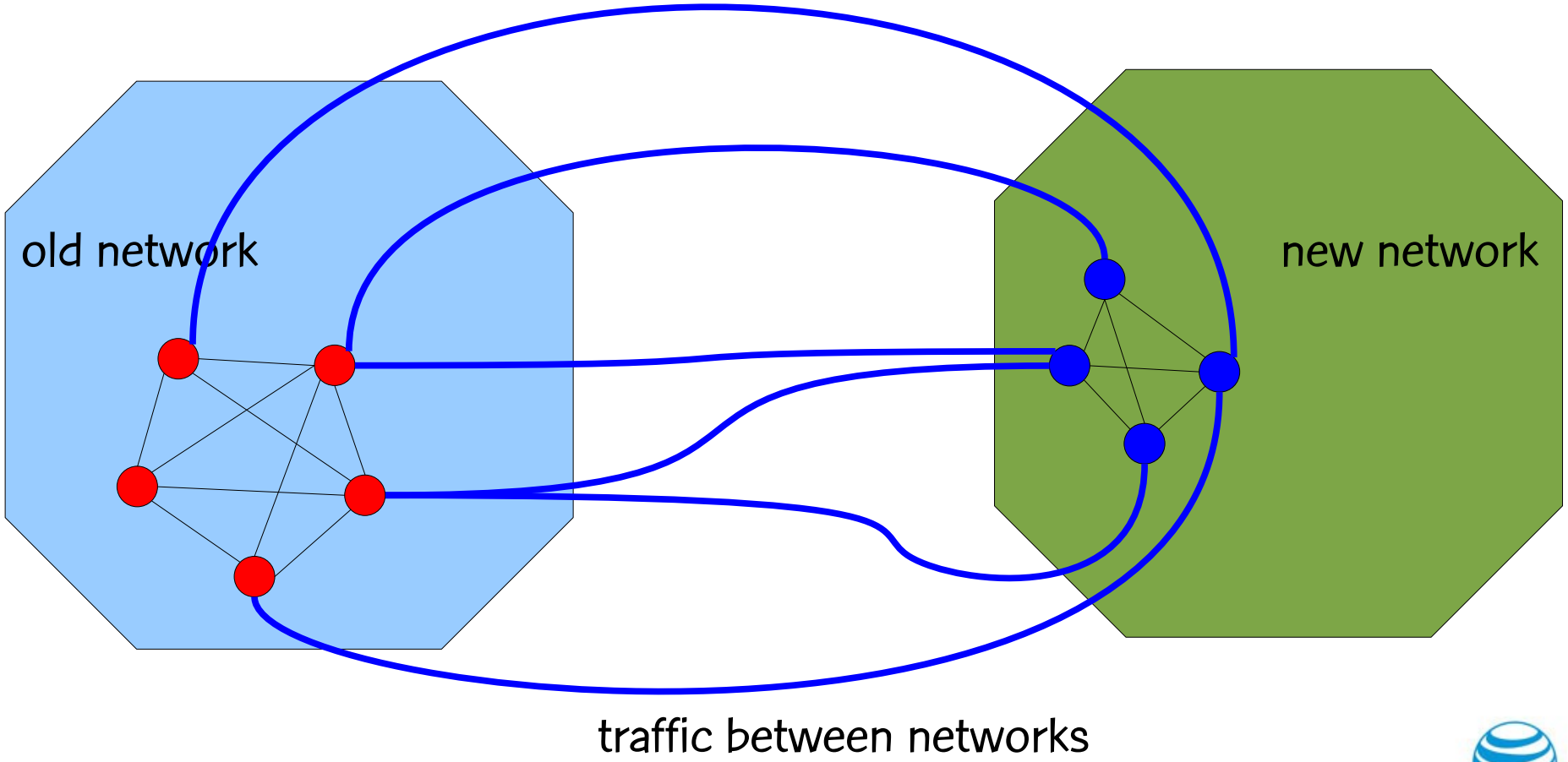
After partial decommissioning of nodes

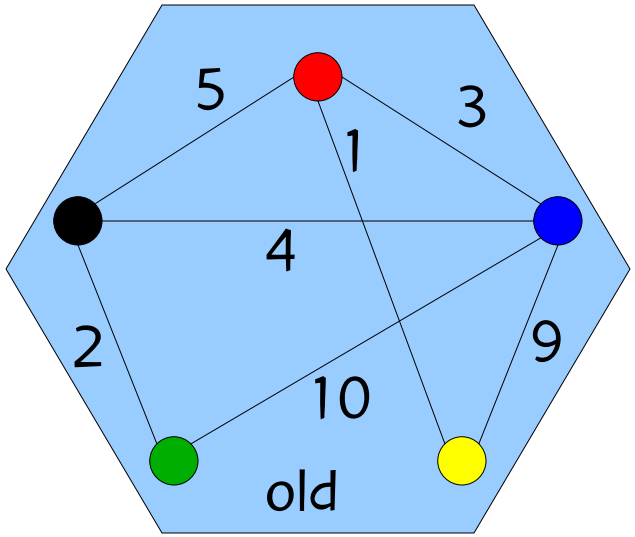


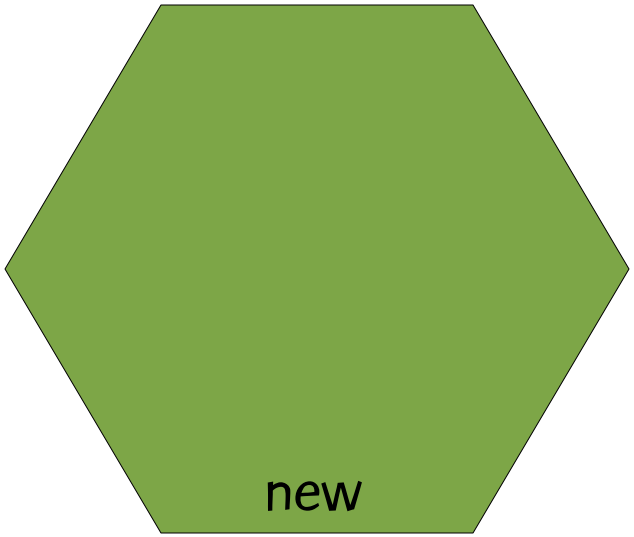


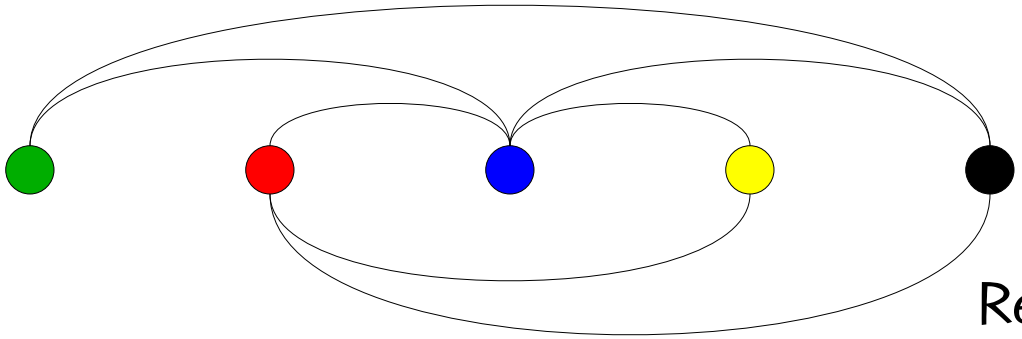
traffic in new network

After partial decommissioning of nodes

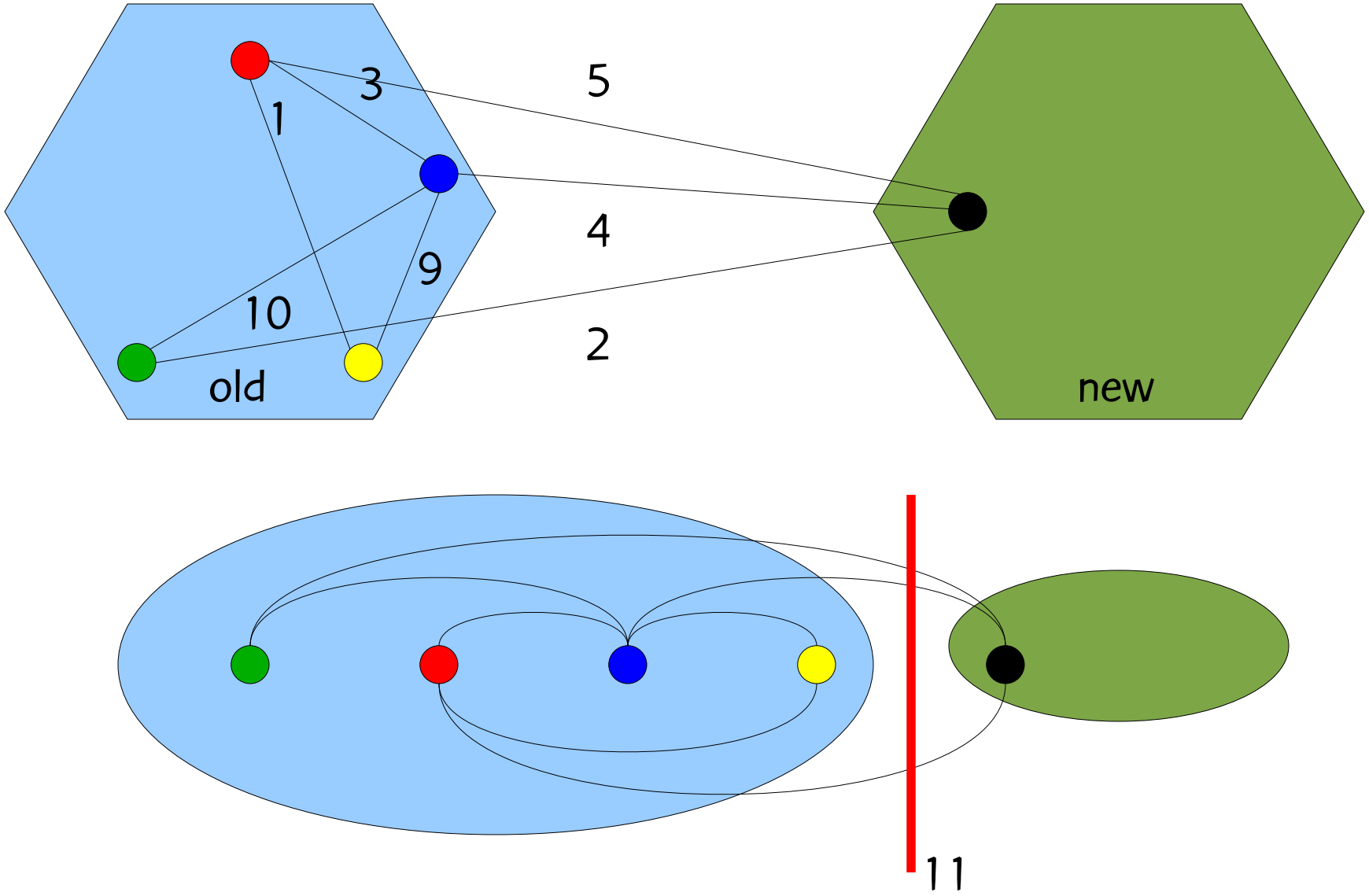


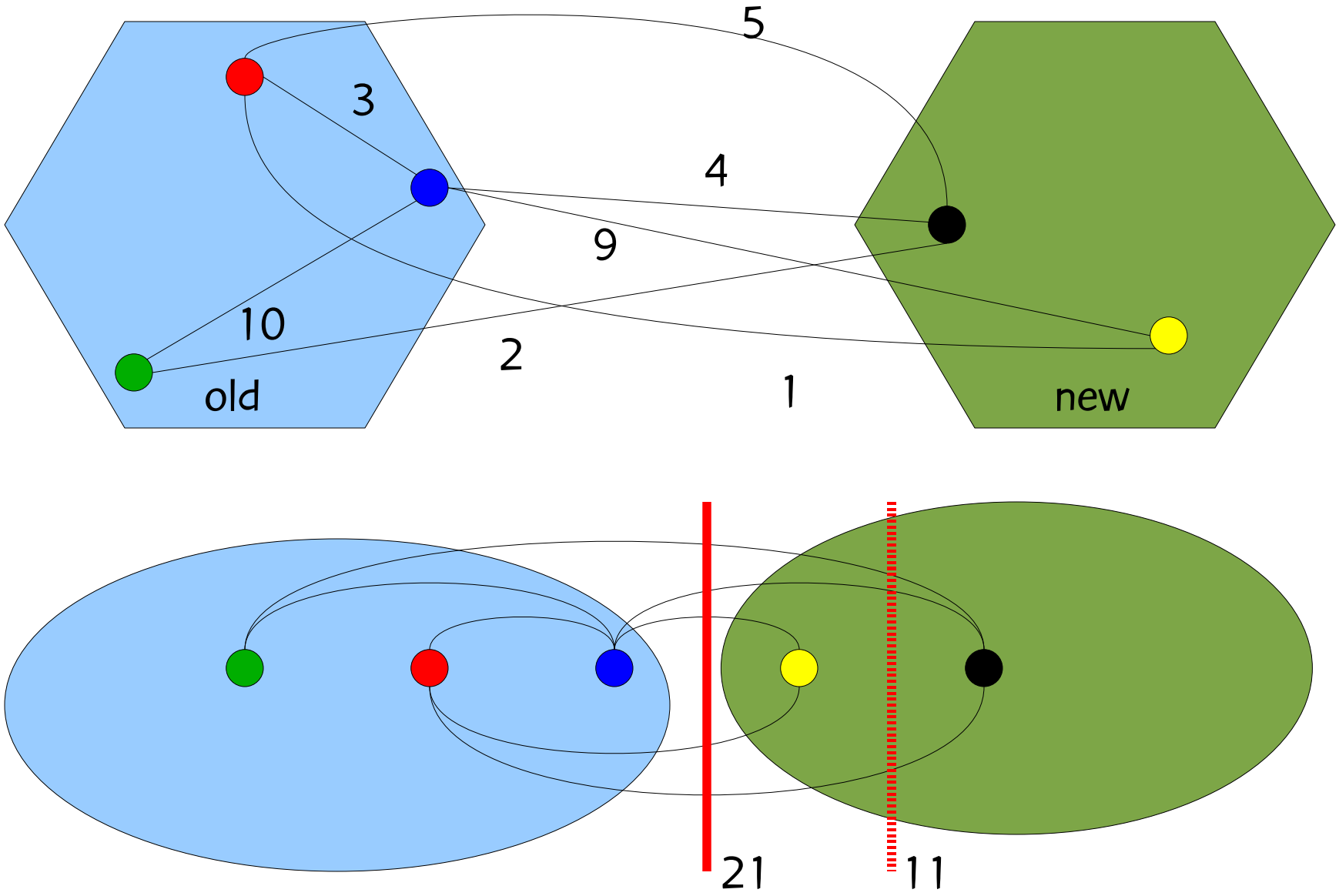


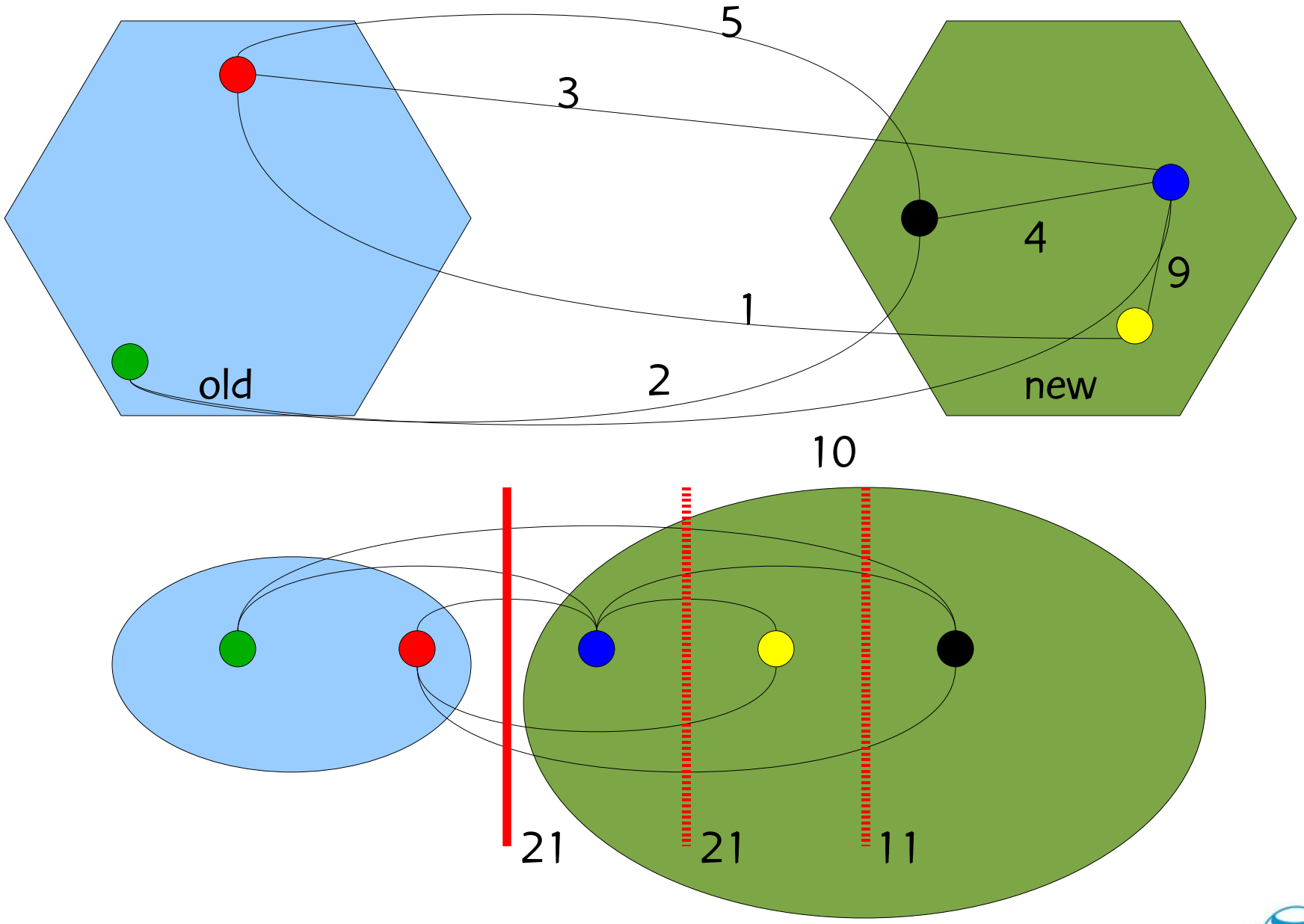


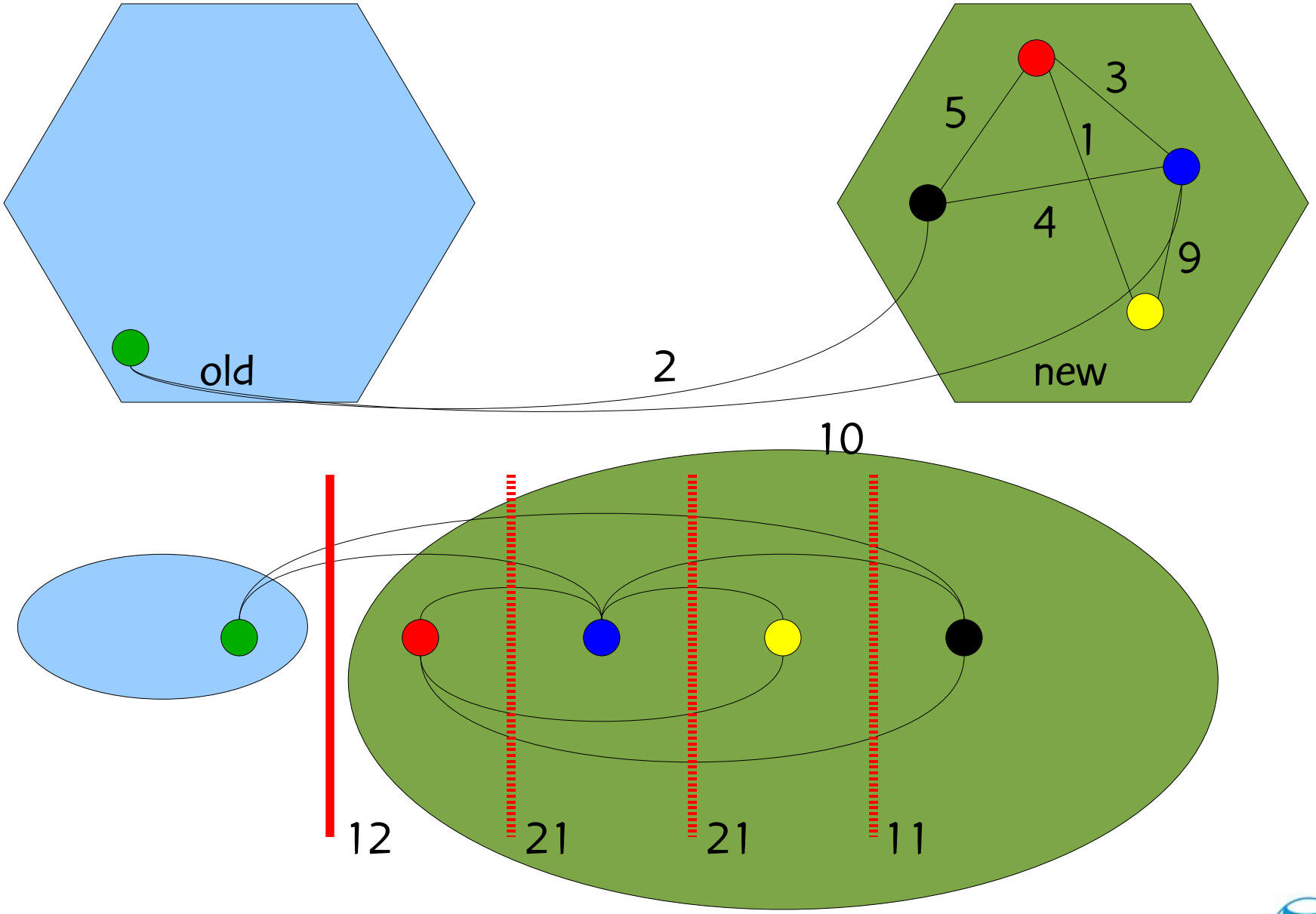


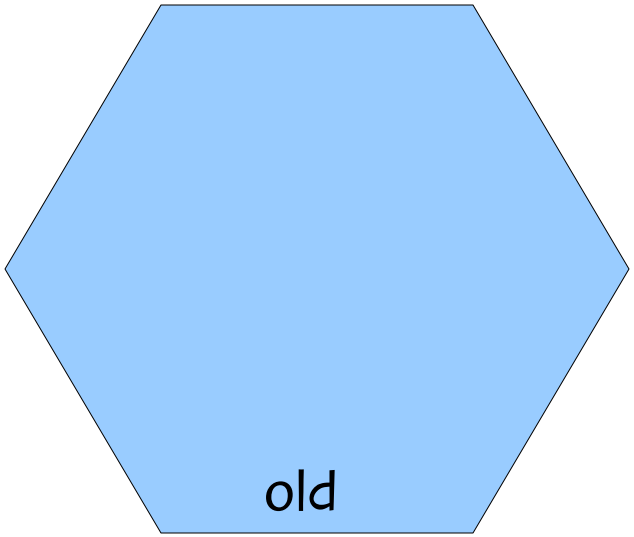
Redraw graph with nodes in line giving order in which nodes are migrated.

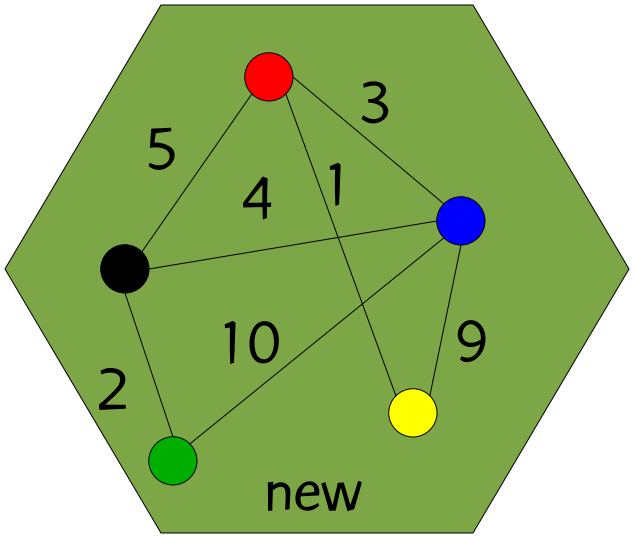


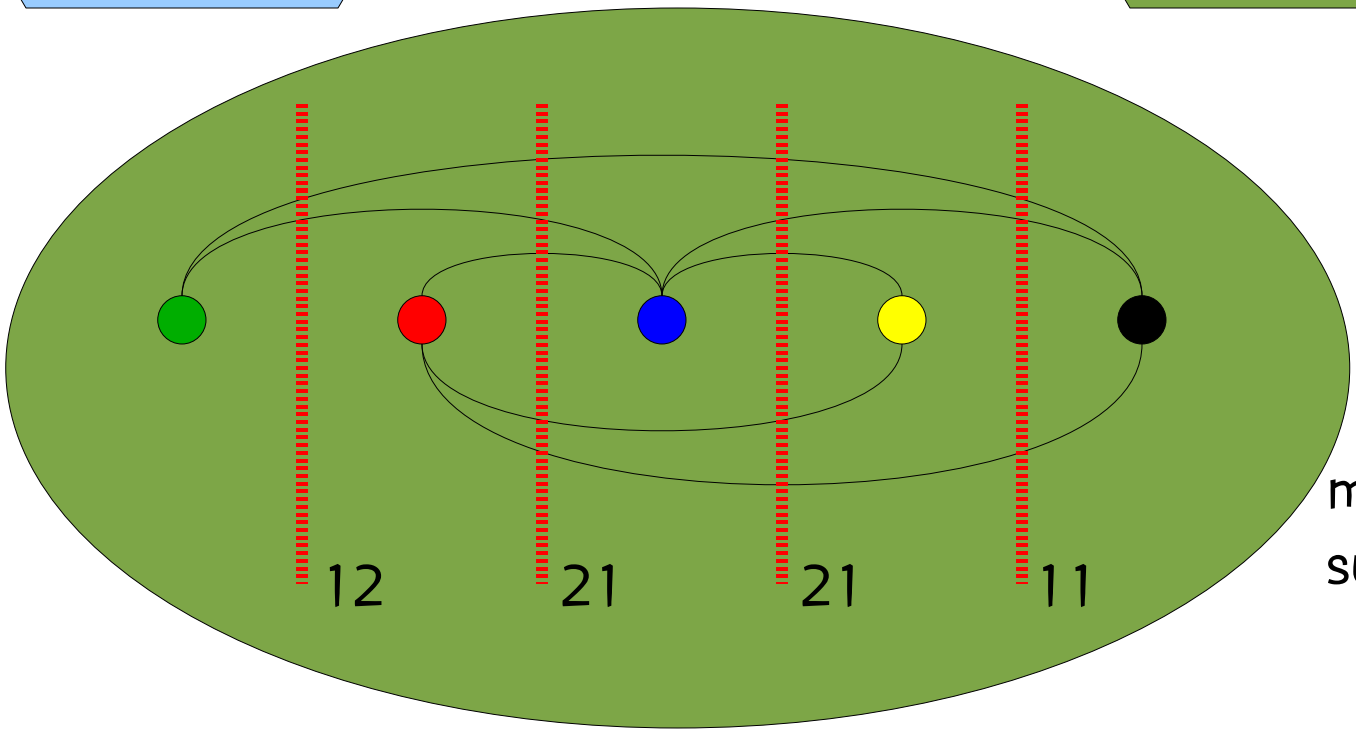






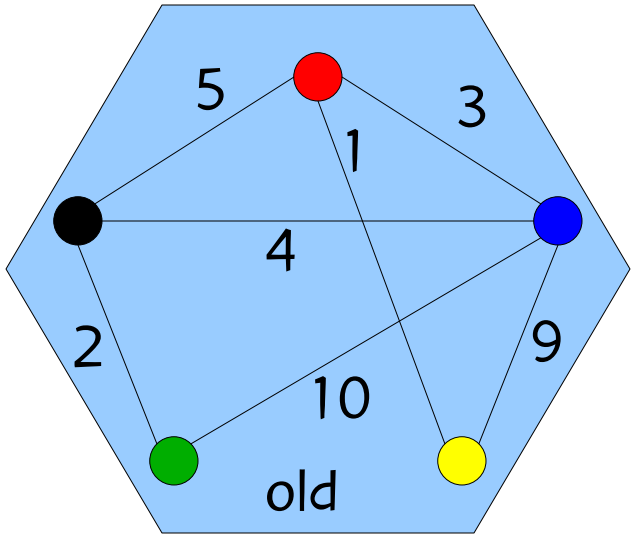




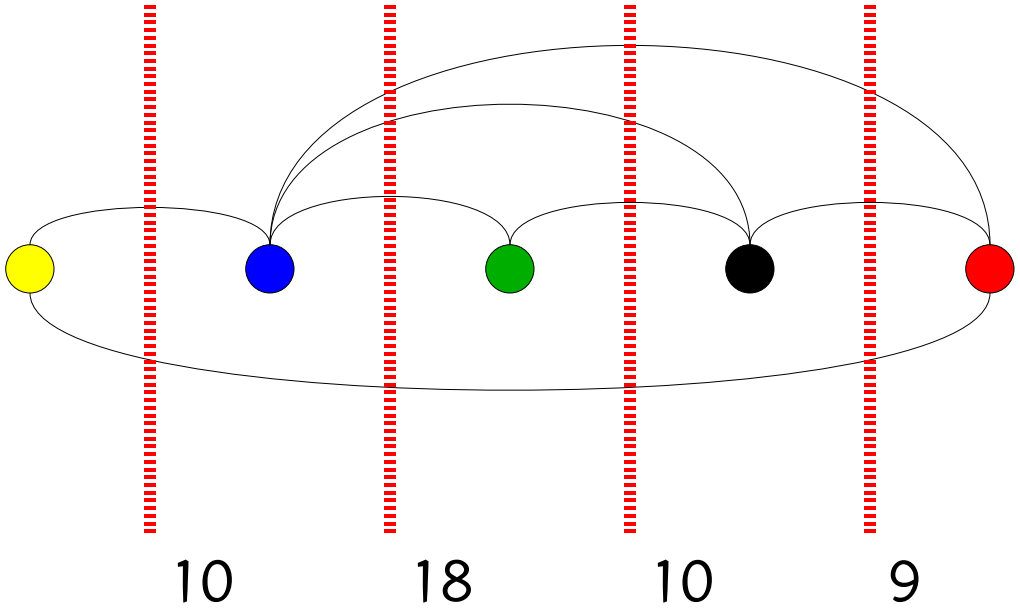


max = 21

sum = 65



Consider another ordering.

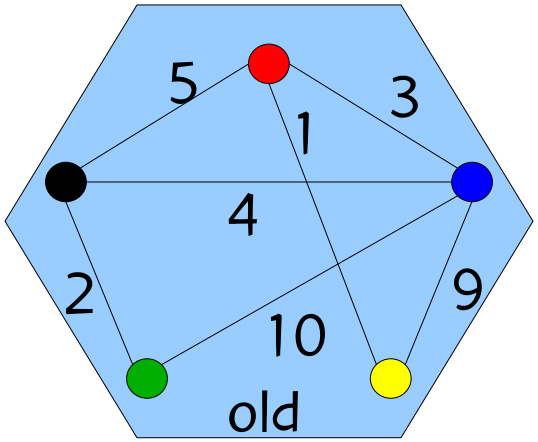


max = 18 < 21

sum = 47 < 65

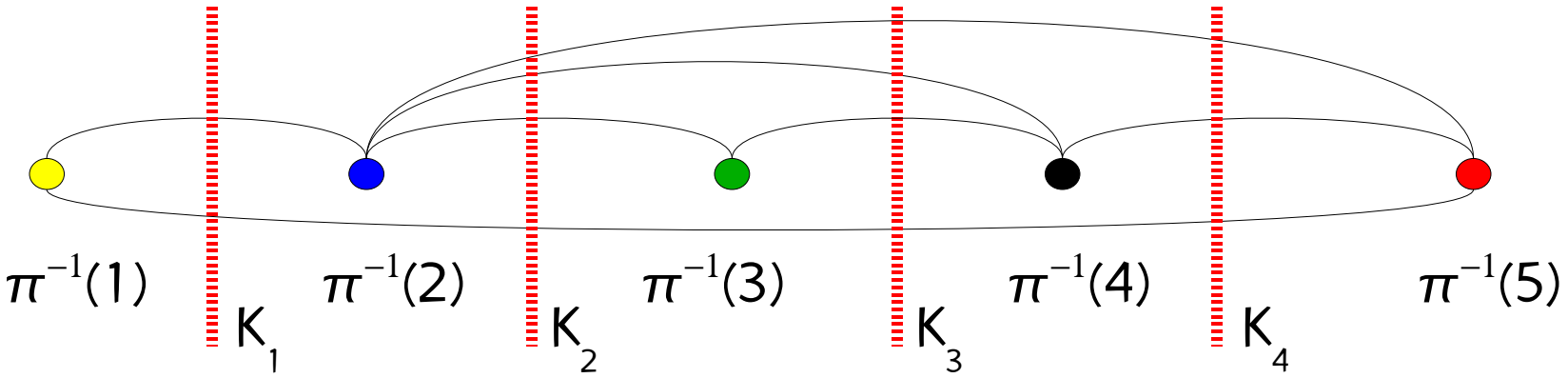
Optimization problem

Given an edge-weighted graph G = (V,E,w), where node set V is the set of switches (routers), edge set E is the set of links between switches (routers), and w is the traffic volume on the links.



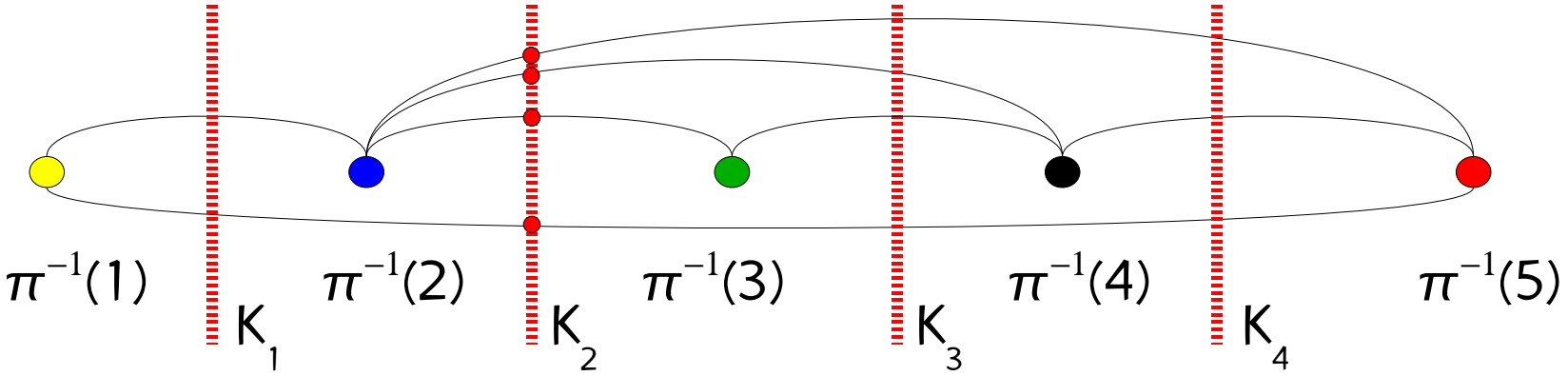
Optimization problem

- Let $\pi: V \rightarrow \{1,\ldots,n=|V|\}$ be an ordering of the switches.
- For $1 \leq i < n$, let cut K_i between nodes $\pi^{-1}(i)$ and $\pi^{-1}(i+1)$ be the sum of the weights of all links with one endpoint in $\pi^{-1}(j)$ and the other in $\pi^{-1}(k)$, for all $j \leq i$ and all $k > i$.



Optimization problem

- Let $\pi: V \rightarrow \{1,\ldots,n=|V|\}$ be an ordering of the switches.
- For $1 \leq i < n$, let cut K_i between nodes $\pi^{-1}(i)$ and $\pi^{-1}(i+1)$ be the sum of the weights of all links with one endpoint in $\pi^{-1}(j)$ and the other in $\pi^{-1}(k)$, for all $j \leq i$ and all $k > i$.



Two measures of solution quality we want to optimize (minimize).

- The value K_{max} of the largest cut is the max-cut,

  i.e., $K_{max} = \max \{K_1, \ldots, K_{n-1}\}$.

- The value K_{sum} is the cut sum,

  i.e., $K_{sum} = K_1 + \cdots + K_{n-1}$.

Optimization problem

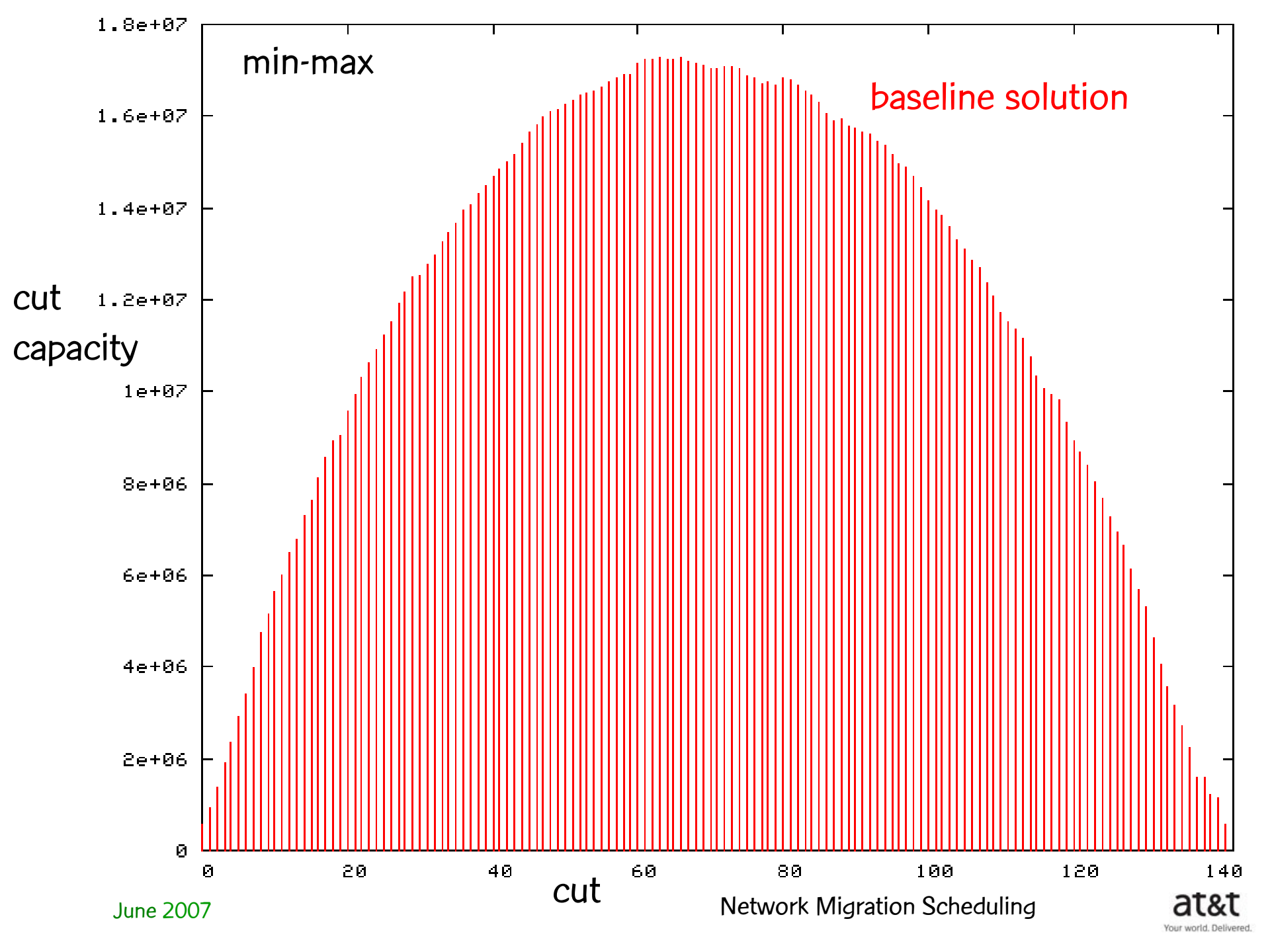
- This is an instance of batch scheduling of multi-grouped units:
  - each switch is a unit
  - each trunk $e \in E$ defines a group g_e
  - group g_e penalty is traffic w_e on trunk e
  - number of periods is $|V|$ (number of switches)
  - one switch is scheduled per period

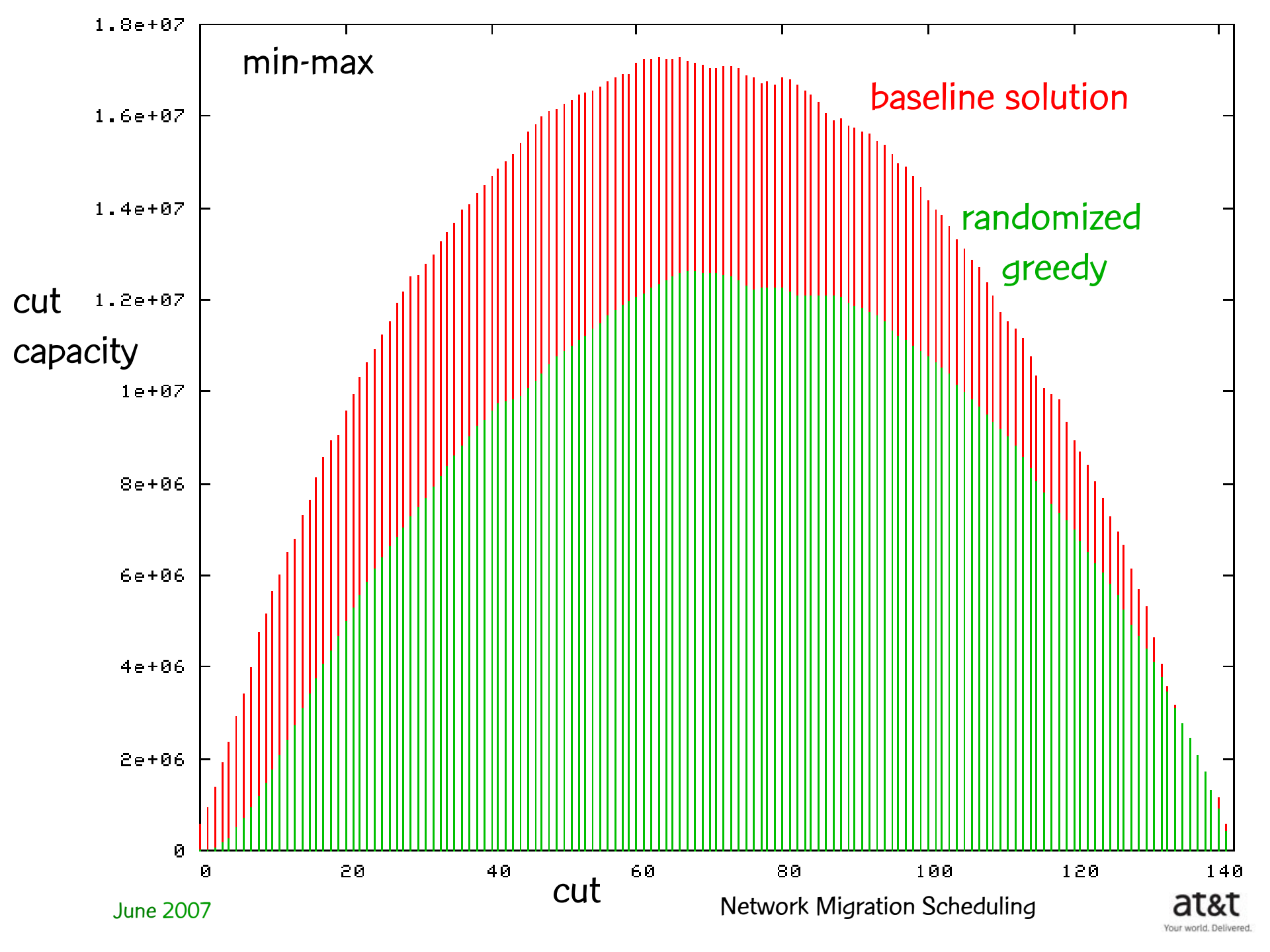
Scheduling 4ESS deloading process

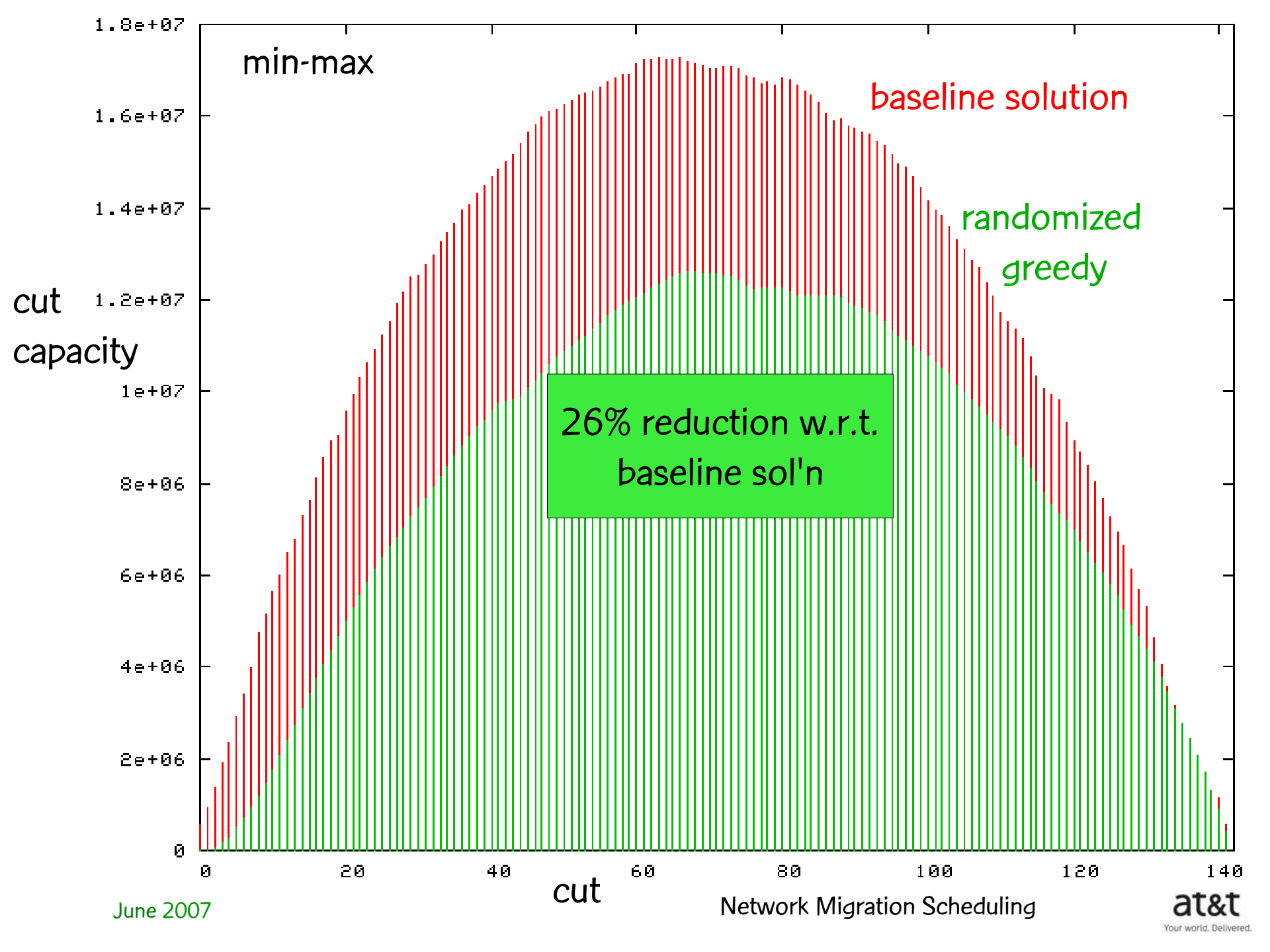
- Legacy AT&T phone network had 140 4ESS switches (nodes) and 9730 trunks (links): 100% edge density.
- 4ESS switches were to be "deloaded" and traffic moved to new IP network.
- One 4ESS switch is "deloaded" at each time period.

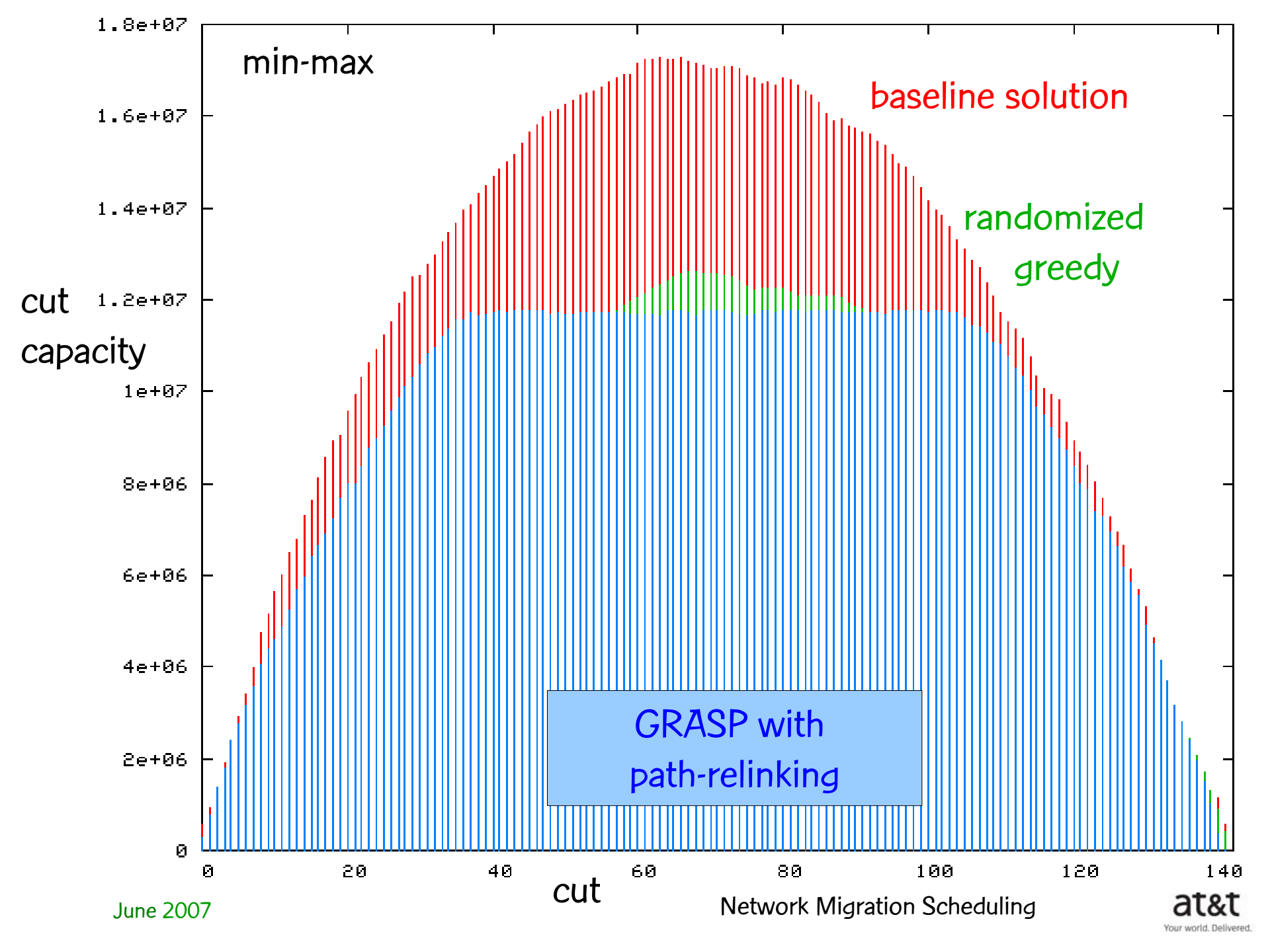
Scheduling 4ESS deloading process

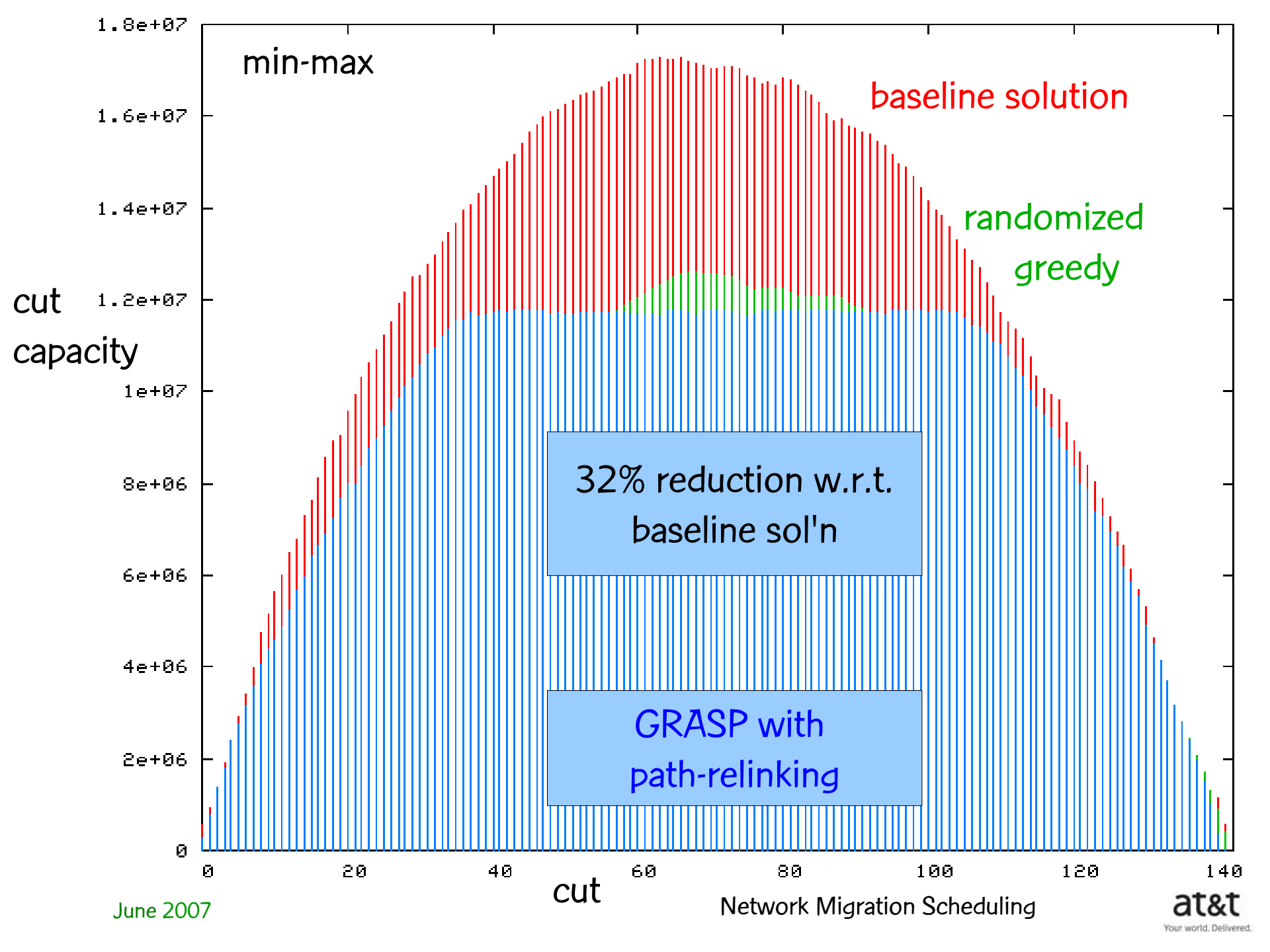
- With David Johnson & Howard Karloff, we built a web-based tool to simulate the deloading process.
- The tool estimates traffic before and after a switch is deloaded.
- Being optimizers, we asked:
  - What is the best ordering for the 4ESS deloading process?
  - We want to minimize the amount of capacity that will need to be built to accomplish the deloading.


min-max
baseline solution
cut capacity
1.8e+07
1.6e+07
1.4e+07
1.2e+07
1e+07
8e+06
6e+06
4e+06
2e+06
0
0
20
40
60
80
100
120
140
cut


min-max
baseline solution
randomized
greedy
cut
capacity
1.8e+07
1.6e+07
1.4e+07
1.2e+07
1e+07
8e+06
6e+06
4e+06
2e+06
0
0
20
40
60
80
100
120
140
cut


min-max
baseline solution
randomized greedy
26% reduction w.r.t. baseline sol'n
cut capacity
cut
1.8e+07
1.6e+07
1.4e+07
1.2e+07
1e+07
8e+06
6e+06
4e+06
2e+06
0
0
20
40
60
80
100
120
140





The End