Cache me if you can

Capacitated Selfish Replication Games

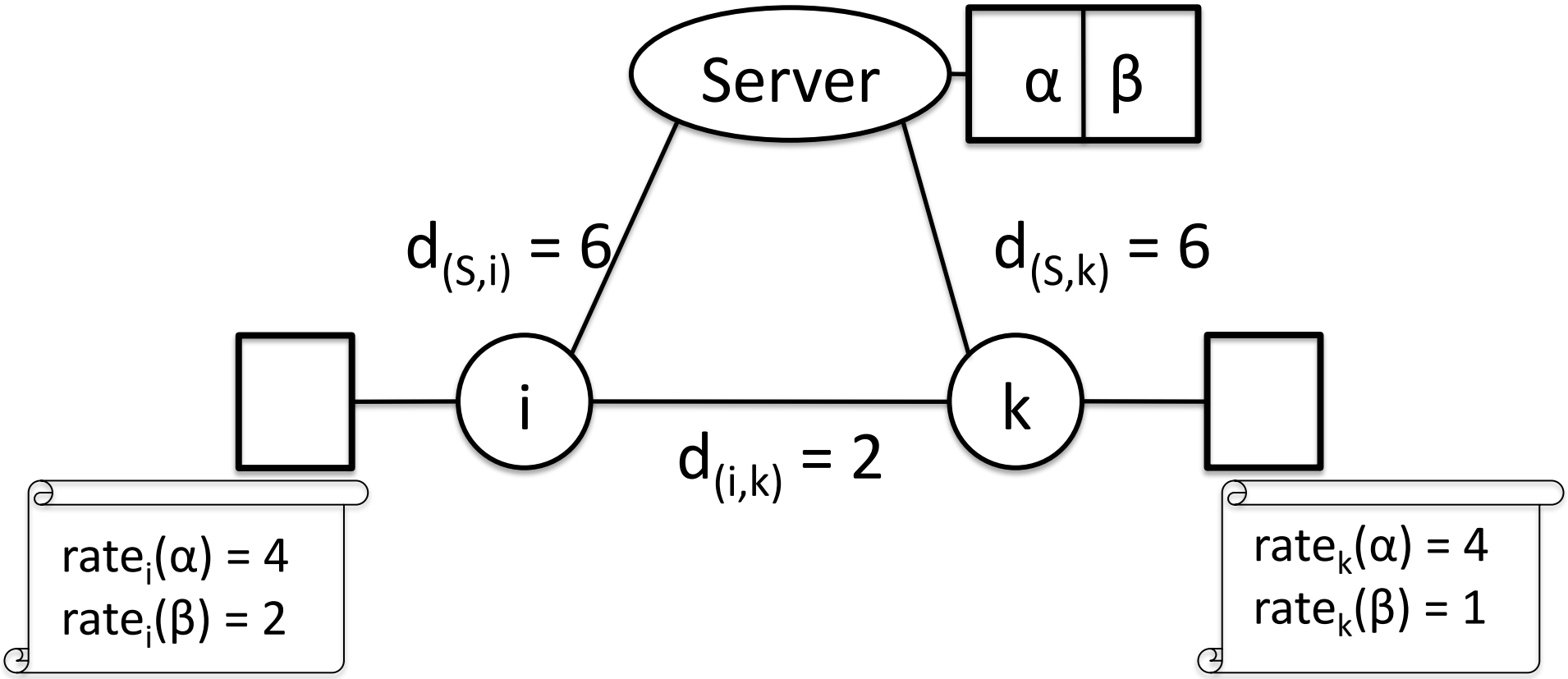
Dimitrios Kanoulas

R. Rajaraman, R. Sundaram

R. Gopalakrishnan, N. Karuturi, C. P. Rangan

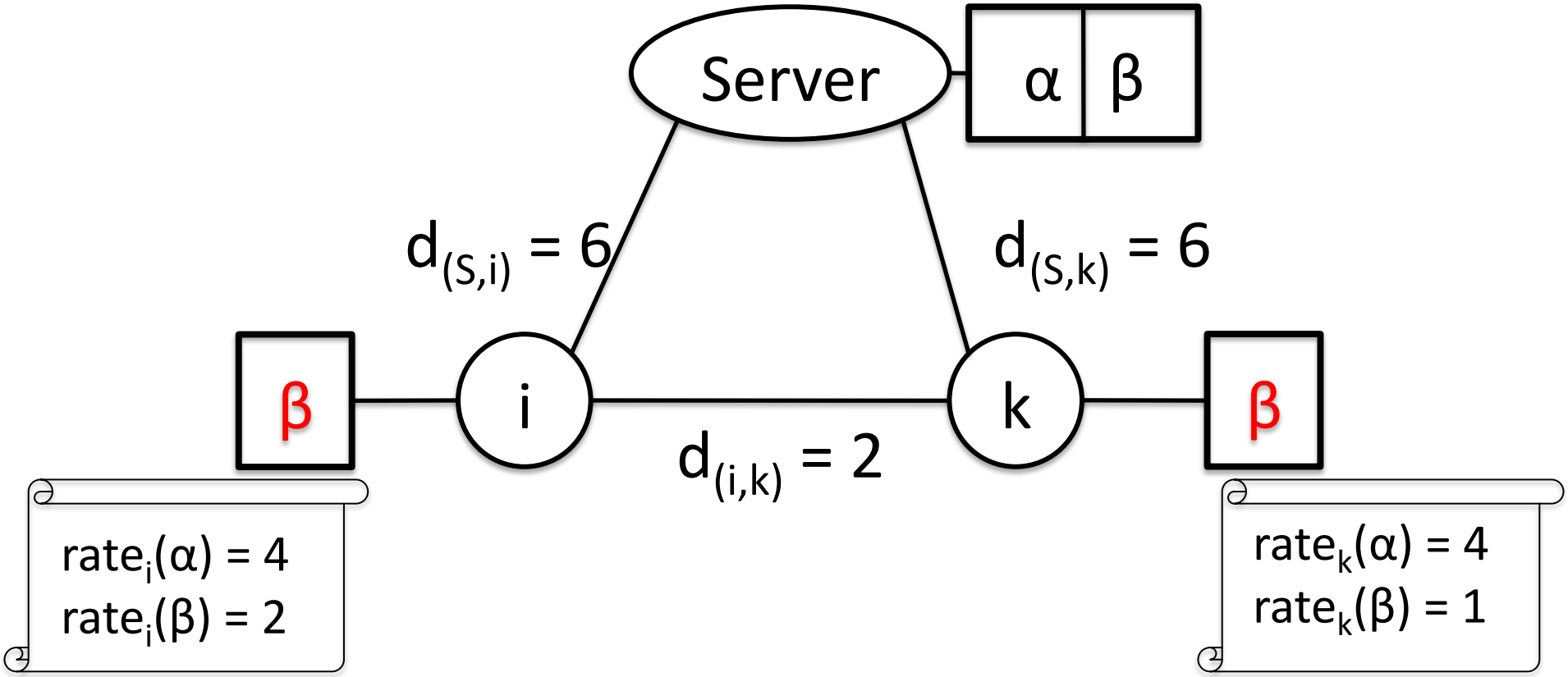
LATIN'12, CCIS Department, Northeastern University

Capacitated Selfish Replication Game Instance



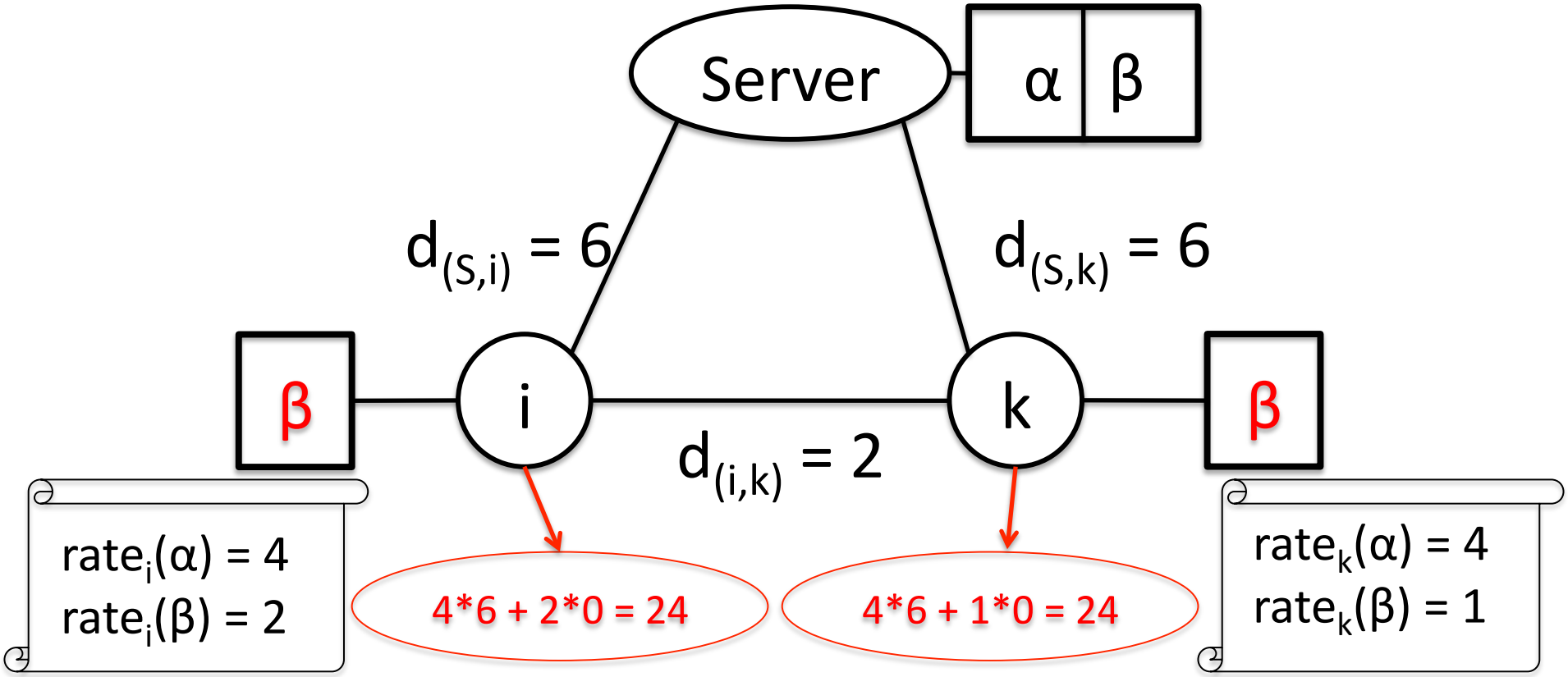
$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



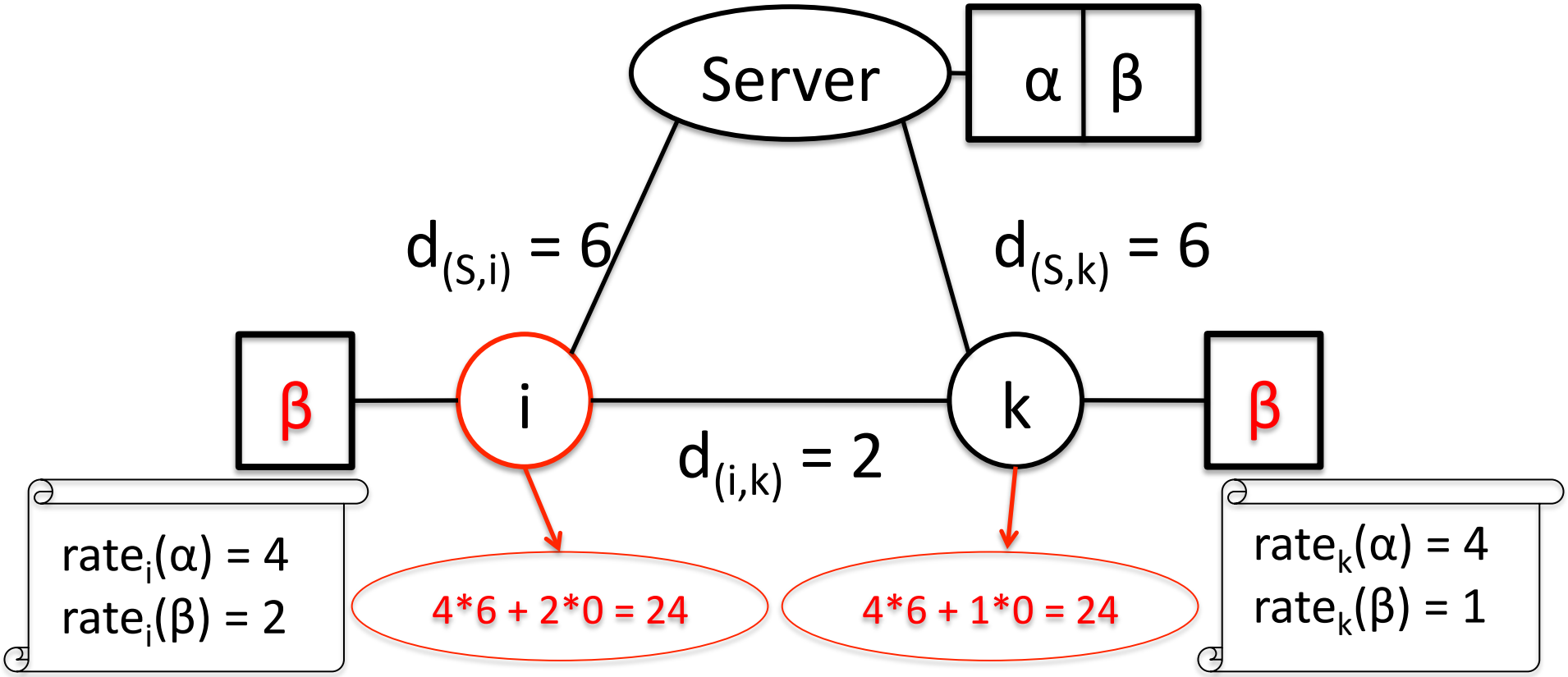
$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



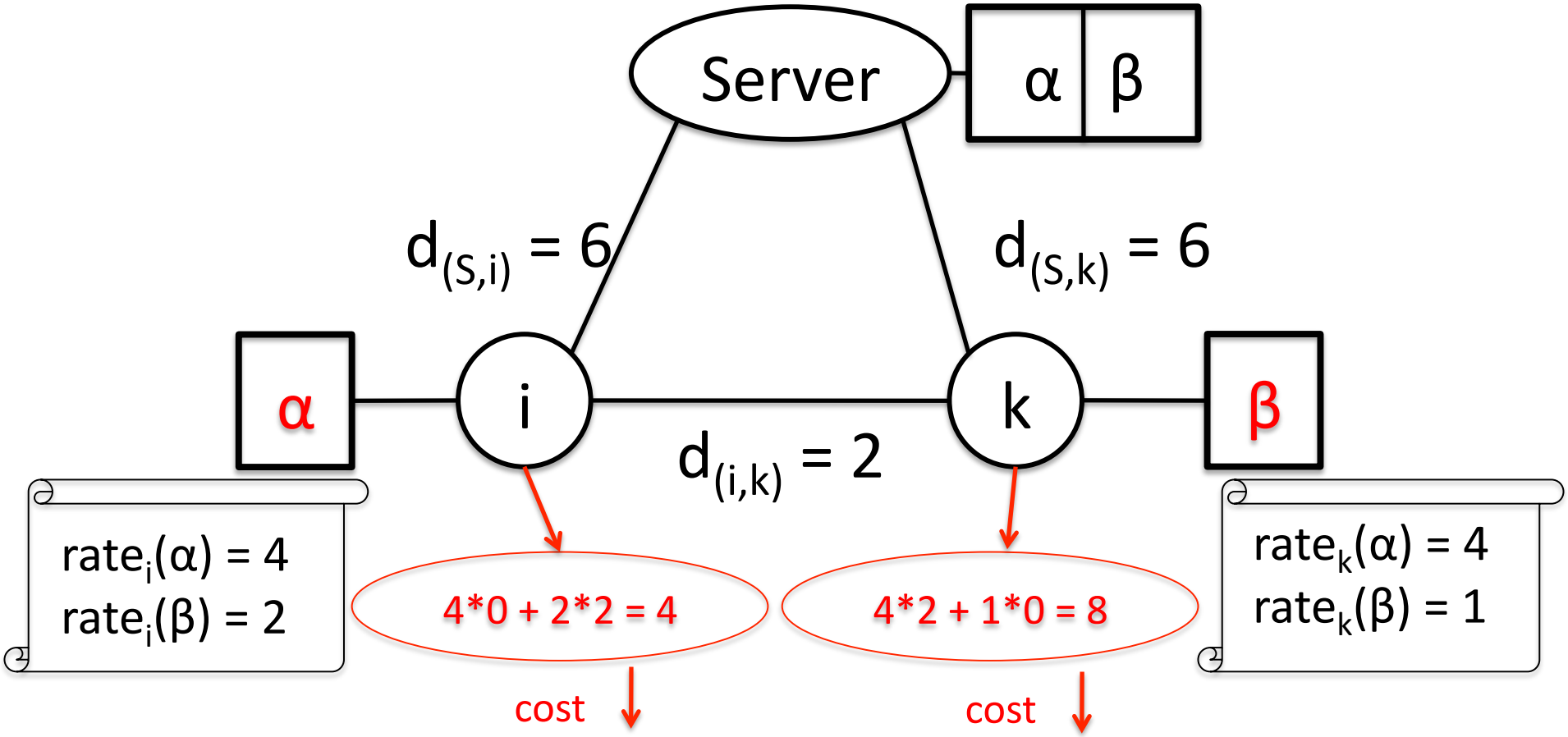
$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



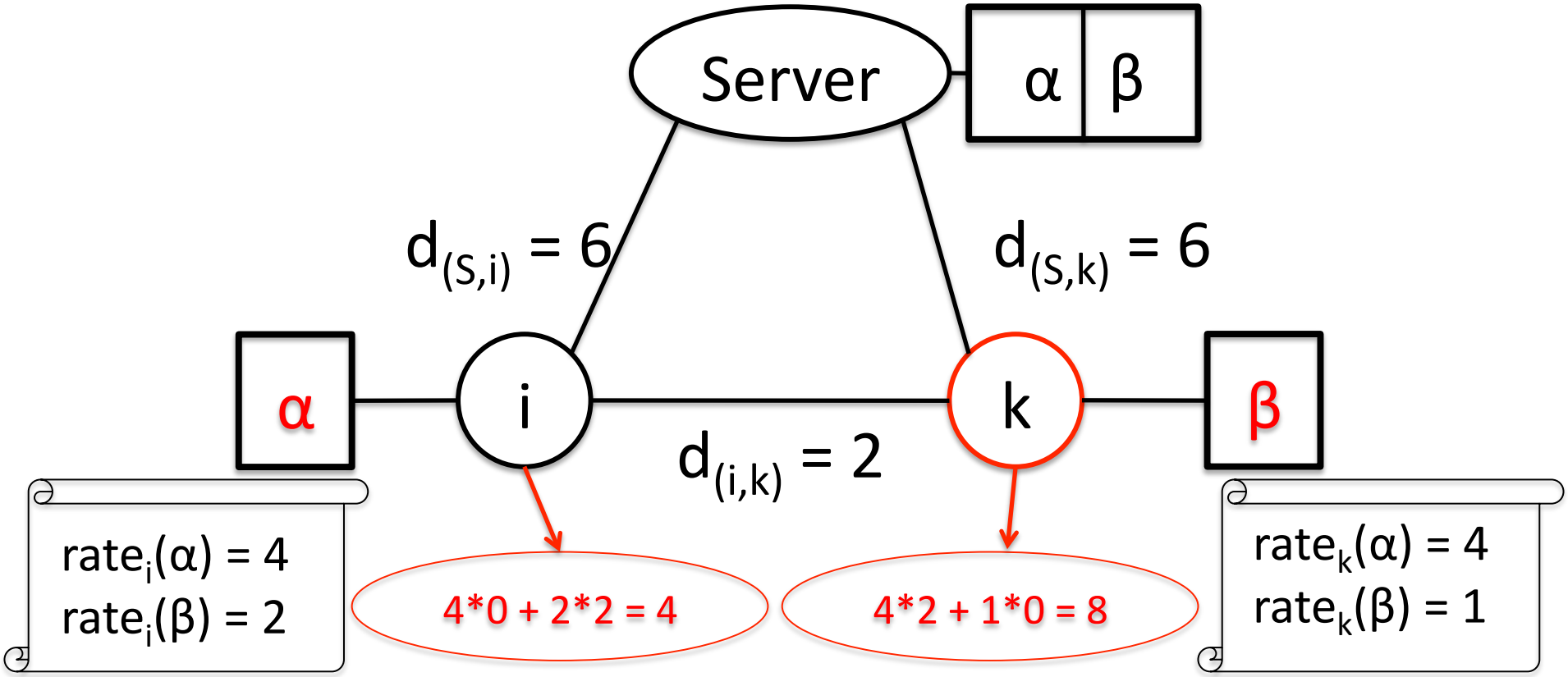
$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



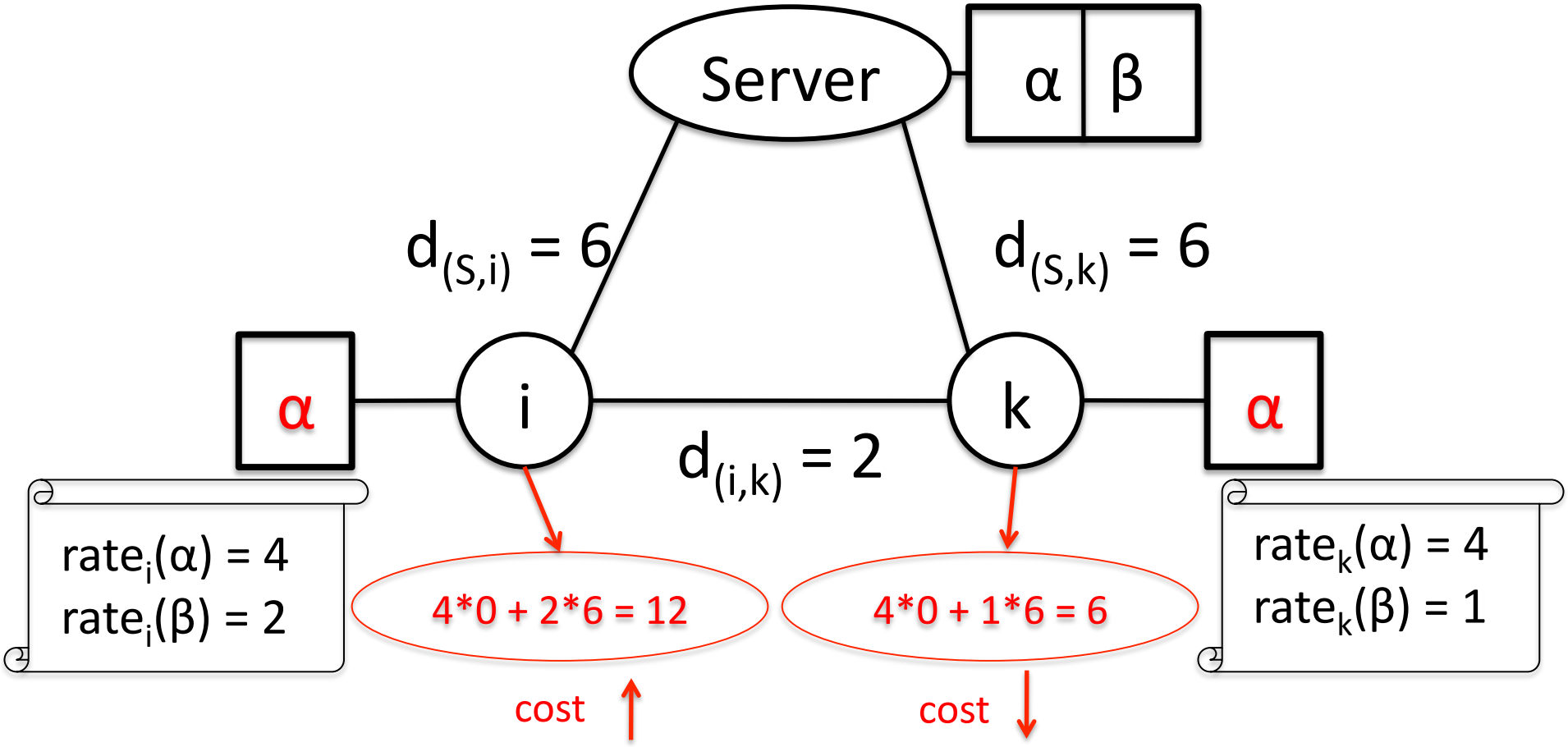
$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



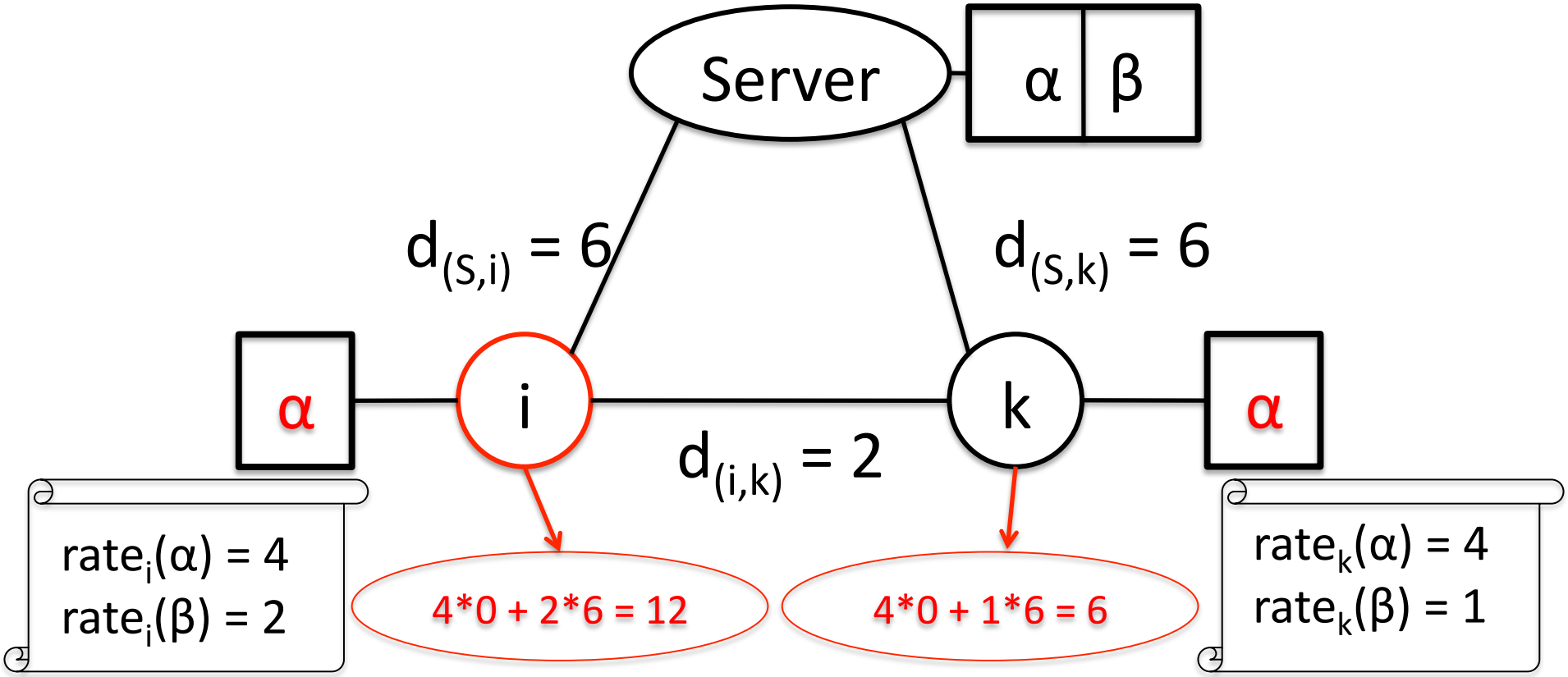
$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



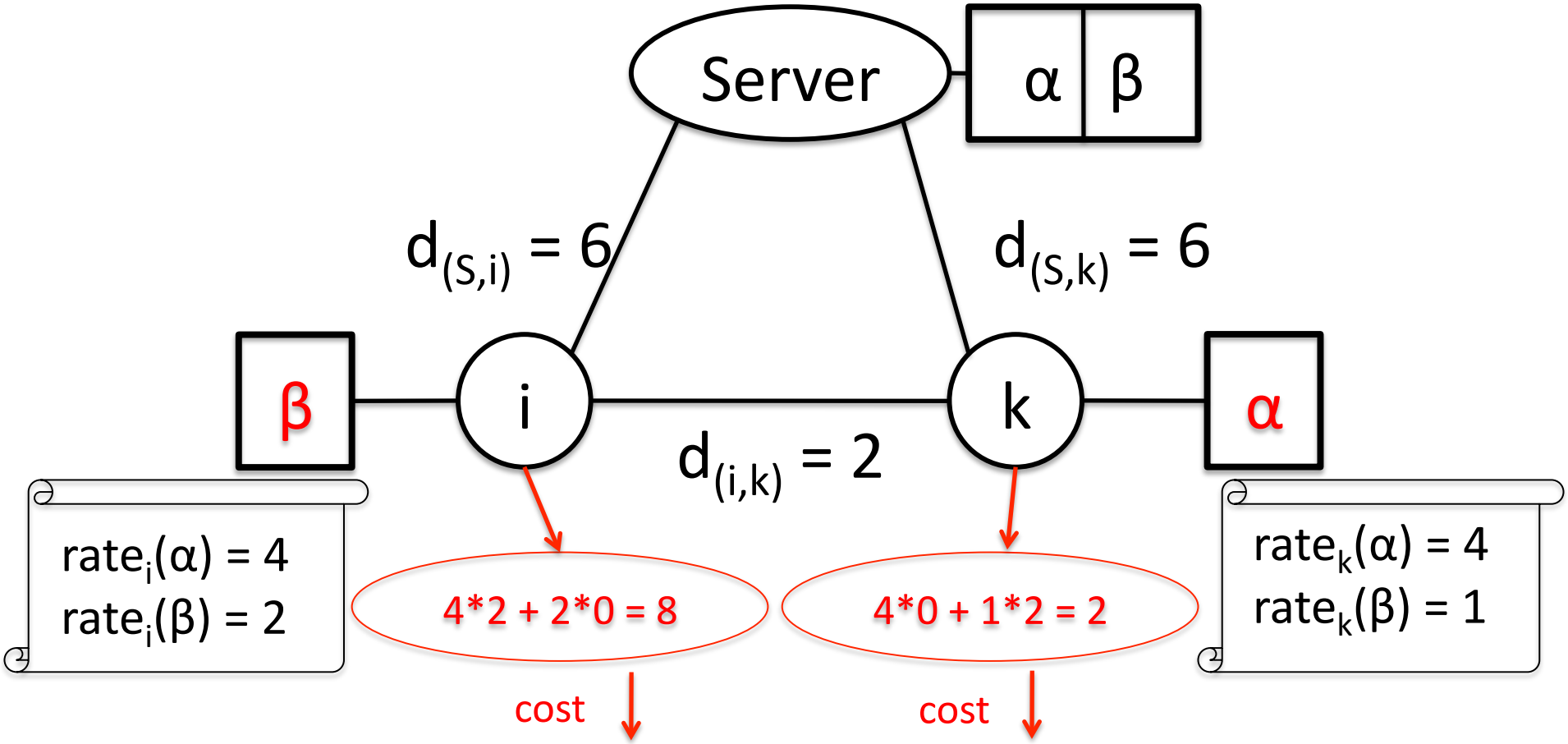
$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



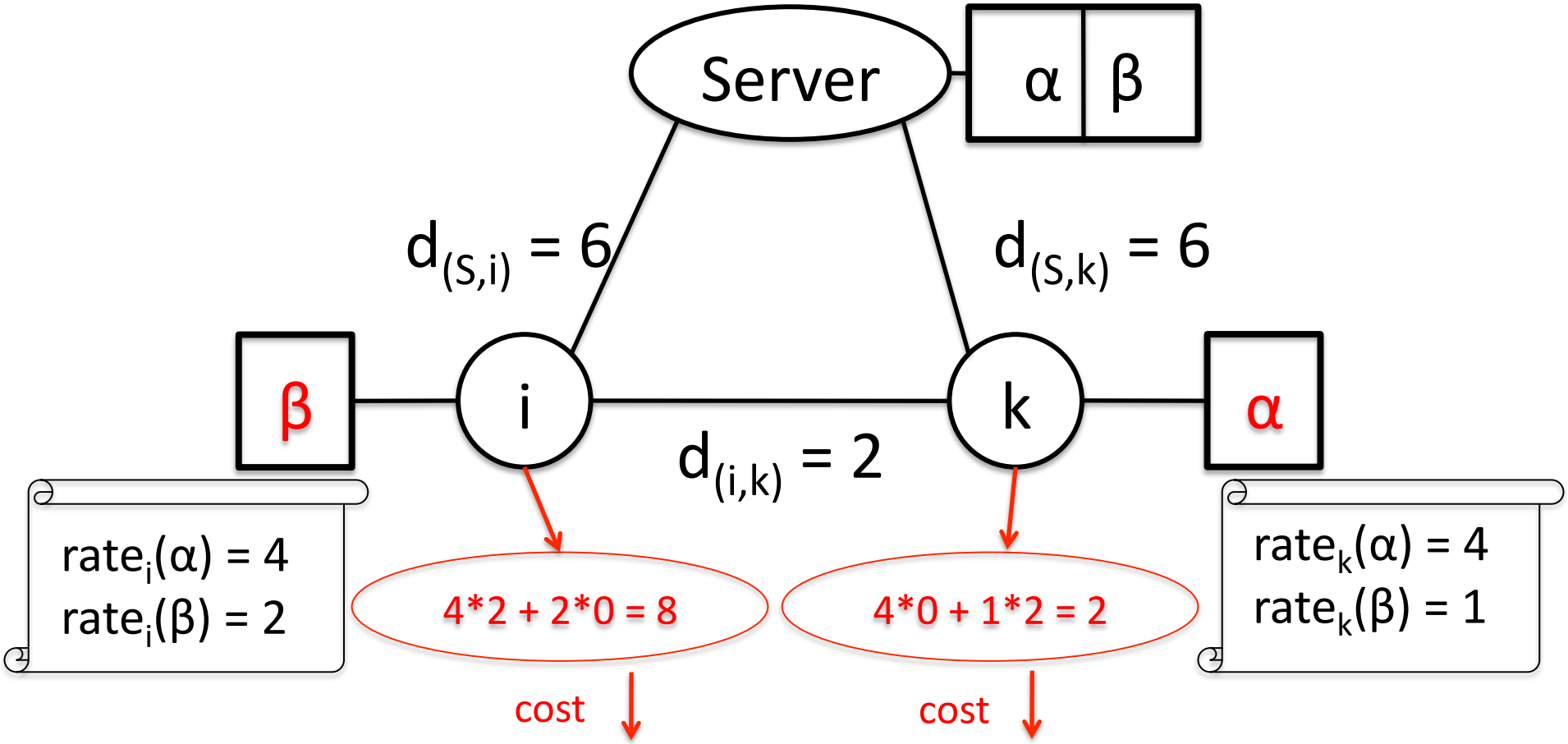
$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



$$\text{Cost}_j = \sum_{\alpha \in O} rate_j(\alpha) d_(j, \text{j's nearest node holding } \alpha)$$

Capacitated Selfish Replication Game Instance



Stable

Capacitated Selfish Replication (CSR)

CSR Network:

- A set *V* of **nodes** sharing a collection *O* of unit-sized **objects**.

- **Access cost** function ***d***: **V x V -> ℜ**
  - ***d(i,j)*** is the **cost** incurred at node *i* for accessing an object at node *j*.

- Each node has a **cache** with **limited capacity** to store a certain number of objects.
  Server node: has the capacity to store all objects.

- **Node's placement:** set of stored objects.
  Global placement: the set of all nodes' placements.

Capacitated Selfish Replication (CSR)

CSR Game:

A **CSR network**, where each node *i*:

- attaches a **utility** U_i to each global placement
- has a **rate** r_i for each object α, representing the rate at which node accesses the object.
- has a **strategy** set which is the set of all feasible placements at the node.

Example of a numerical utility function (sum):

$$U_i(P) = -\sum_{\alpha \in O} r_i(\alpha) d_{(i,i\text{'s nearest node holding } \alpha)}$$

Pure Nash Equilibria: global placement such that there is no node that can increase its utility by unilaterally deviating from its strategy, i.e. by replicating different set of objects.

Applications

P2P networks and content delivery applications

- **P2P movie sharing service:**
distributed version of ***netflix*** where you can access movies (*objects*) both from other users (*nodes*) and from the netflix server at some *cost* ($$).

- **Brave new 4G wireless world:**
being both a consumer and provider of **apps** (*objects*) to others (*nodes*) around you, where the *server* is the app store.

Related Work

[*B. Chun, K. Chaudhuri, H. Wee, M. Barreno, C. H. Papadimitriou, and J. Kubiatowicz*] - PODC 2006

Pure Nash equilibria in a setting with storage cost, but no cache capacities.

[*N. Laoutaris, O. Telelis, V. Zissimopoulos, and I. Stavrakakis*] – Trans. Parallel Distr. Syst. 2006
[*G. Pollatos, O. Telelis, and V. Zissimopoulos*] - Networking 2008
Hierarchical networks: one-level using sum utility function.

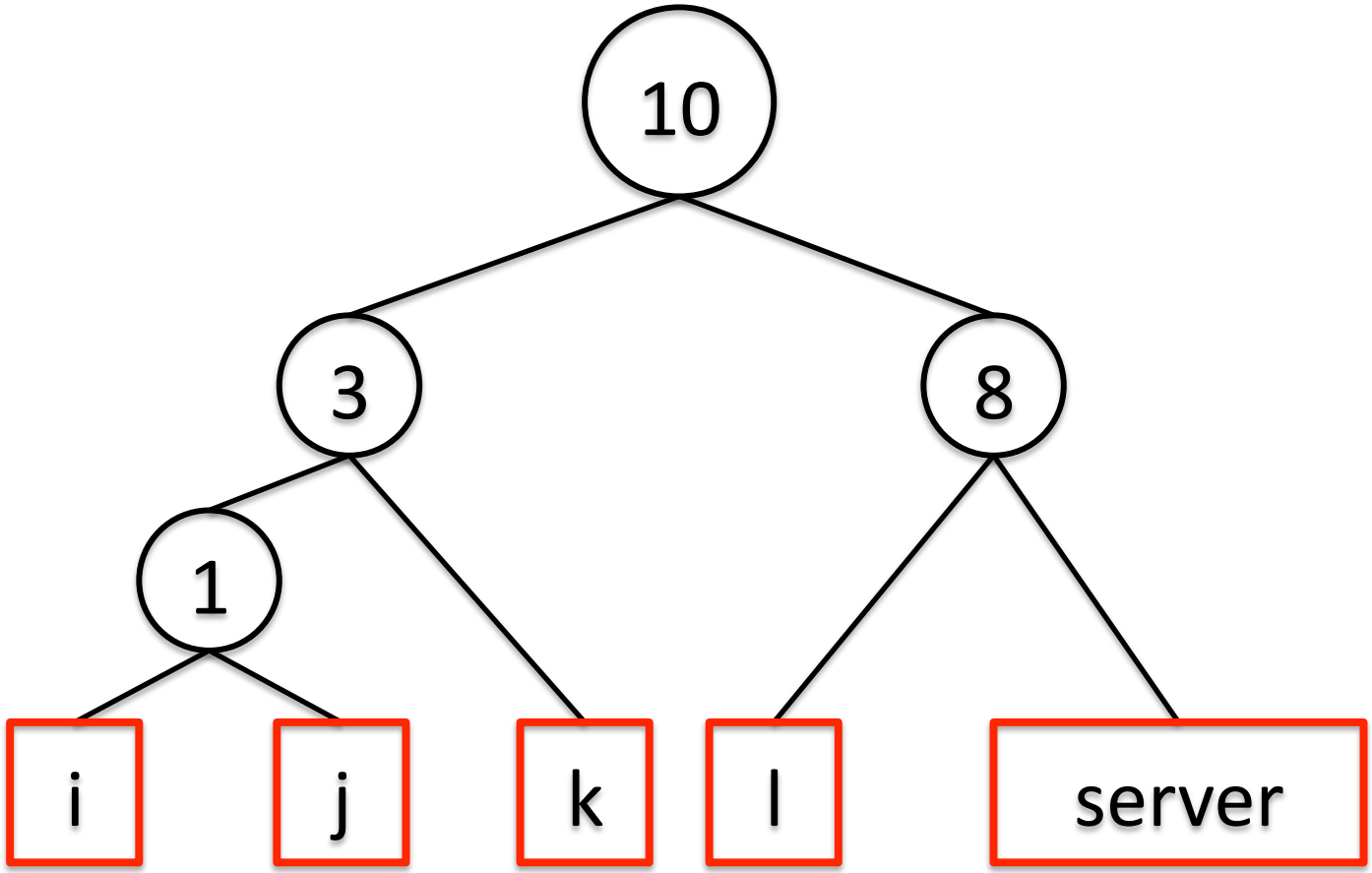
Our work: Capacitated *in* general networks

Our Results
(Does pure Nash equilibrium always exist?)

Object rates & count	Undirected networks	Directed Networks
Binary, two objects	Yes (Finding an equilibrium: in **P**)	No (In **P** to decide)
Binary, three or more objects	Yes (Finding an equilibrium: we don't know if in **P**)	No (**NP-complete** to decide)
General, two objects	Yes (Finding an equilibrium: in **P**)	No (**NP-complete** to decide)
General, three or more objects	No (**NP-complete** to decide) ***Hierarchical***: Yes (Finding an equilibrium: in **P**)	No (**NP-complete** to decide)

Hierarchical Networks

Example of a 3-level hierarchical network:



Why Hierarchical Networks?

Have been extensively used to model communication costs of **content delivery** and **P2P** systems:

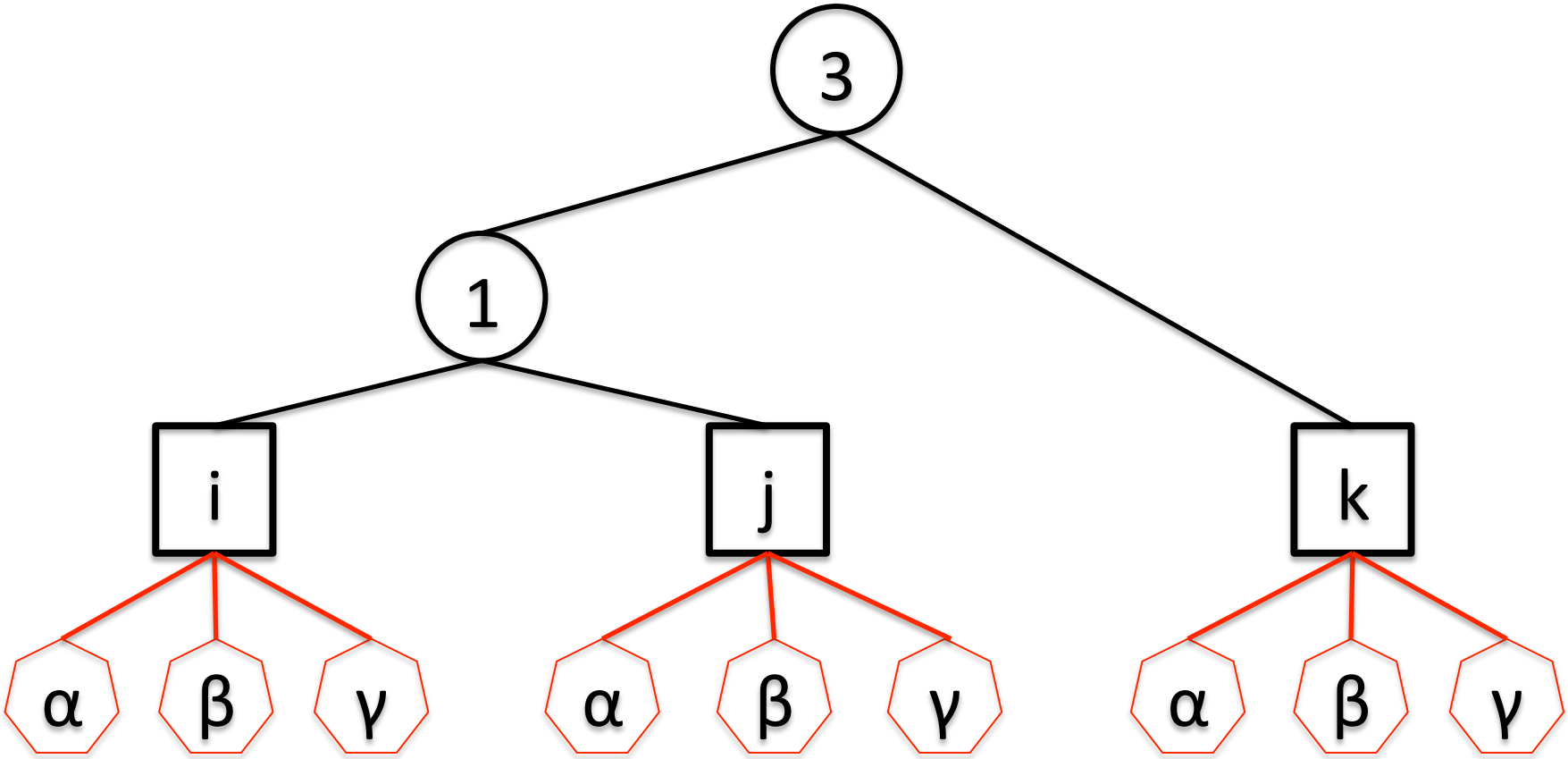
- [*Karger et al. – STOC 1997*]: for modeling **content delivery networks**.

- [*Leff et al. - IEEE Trans. Parallel Distrib. Syst. 1993*], [*Tewari et al. – ICDCS 1999*], [*Korupolu et al. - Journal of Algorithms 2001*]: for modeling **cooperative caching** in hierarchical networks.

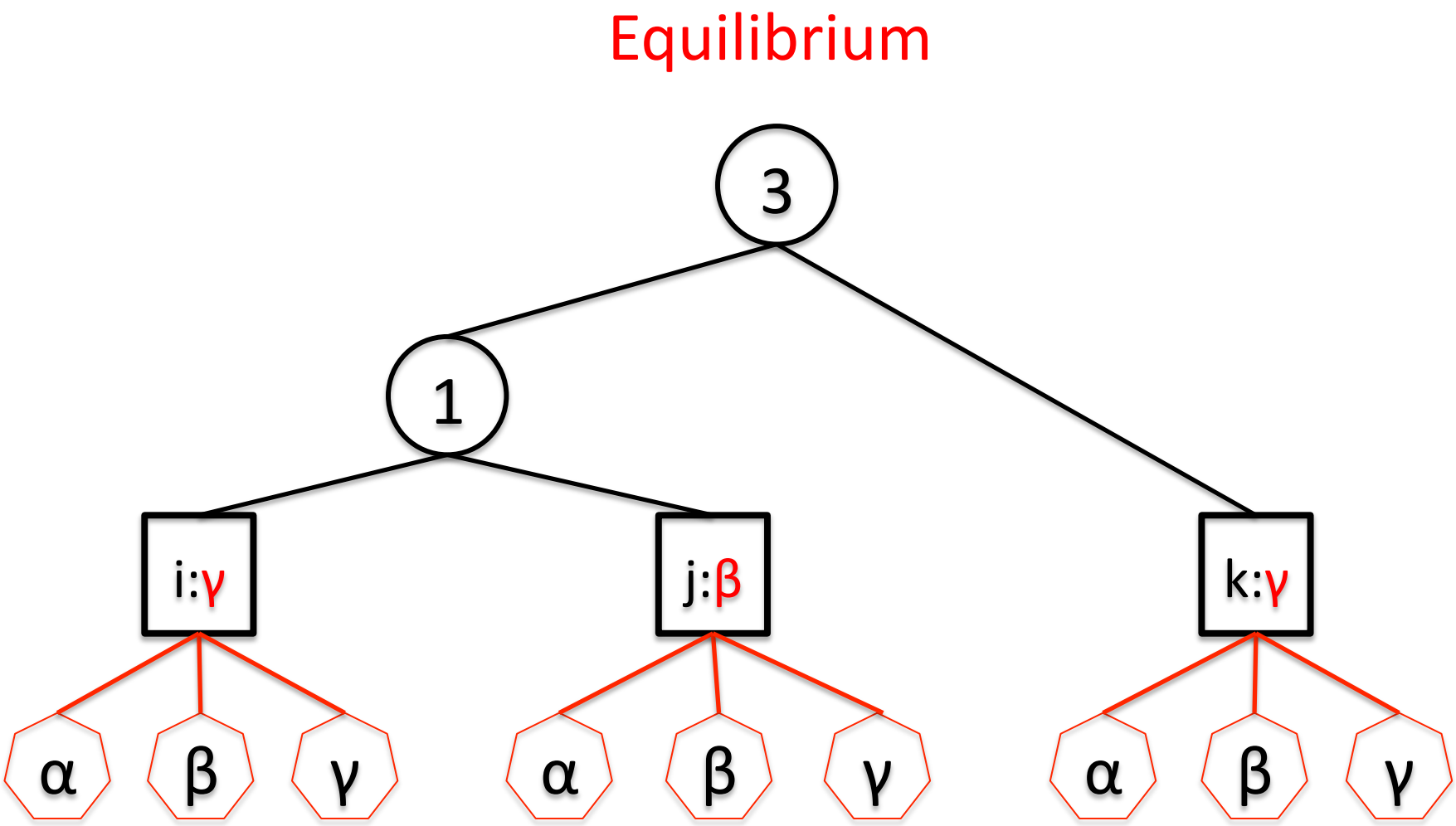
Our Results on Hierarchical Networks

Poly-time construction of equilibrium:

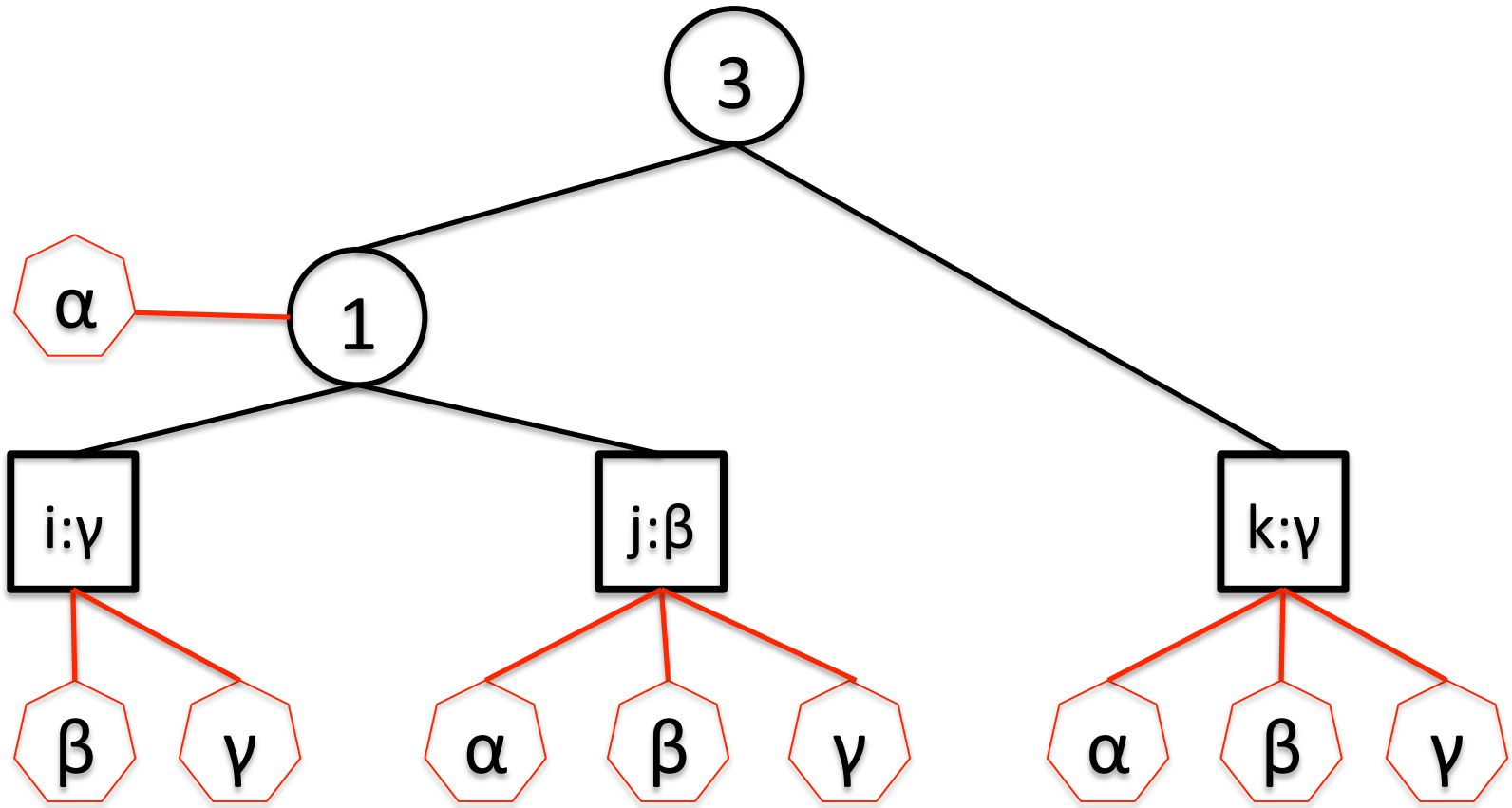
Fictional Players Algorithm



Our Results on Hierarchical Networks



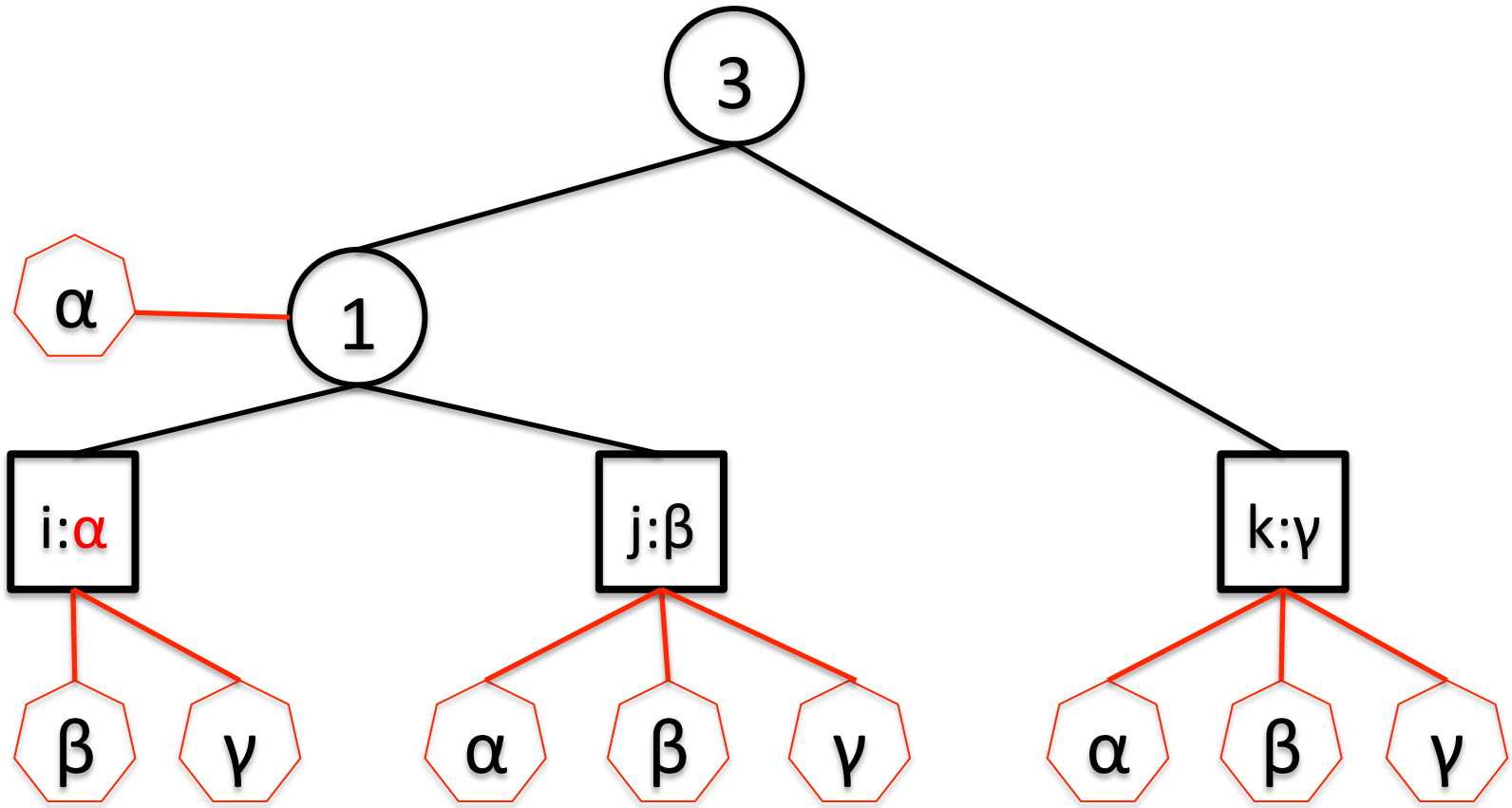
Our Results on Hierarchical Networks



Our Results on Hierarchical Networks

Equilibrium

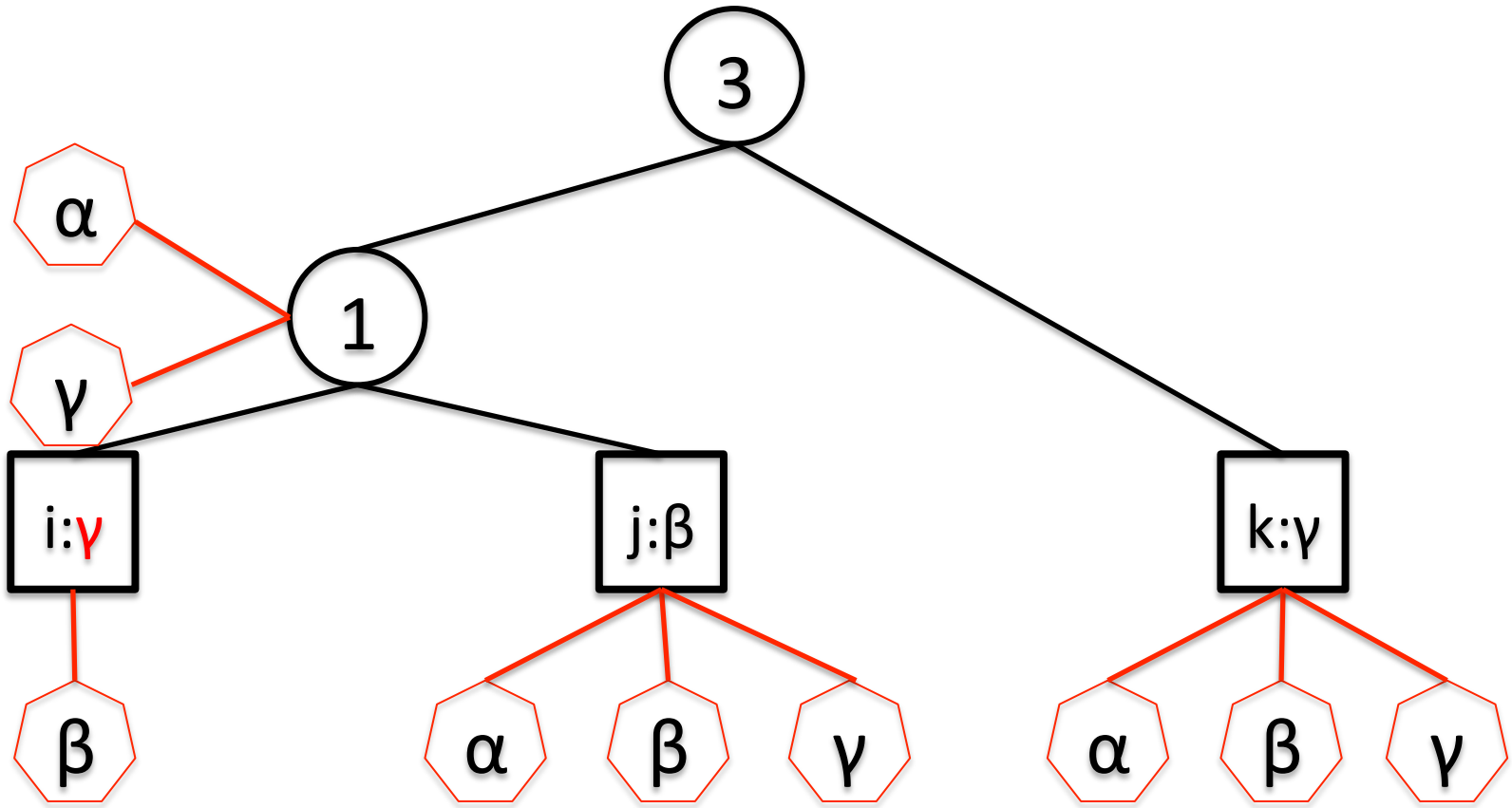
(in polynomial number of steps)



Our Results on Hierarchical Networks

Equilibrium

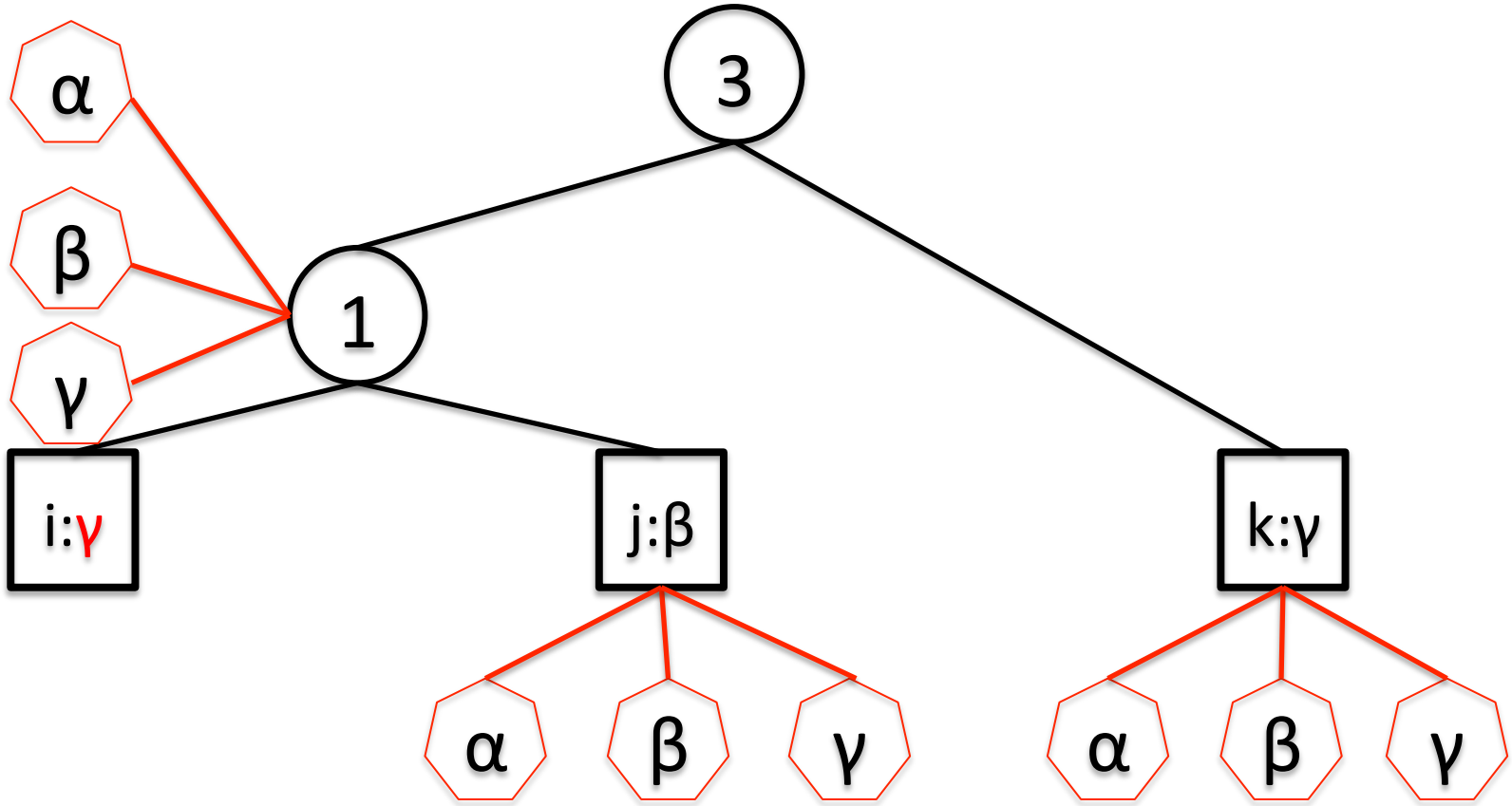
(in polynomial number of steps)



Our Results on Hierarchical Networks

Equilibrium

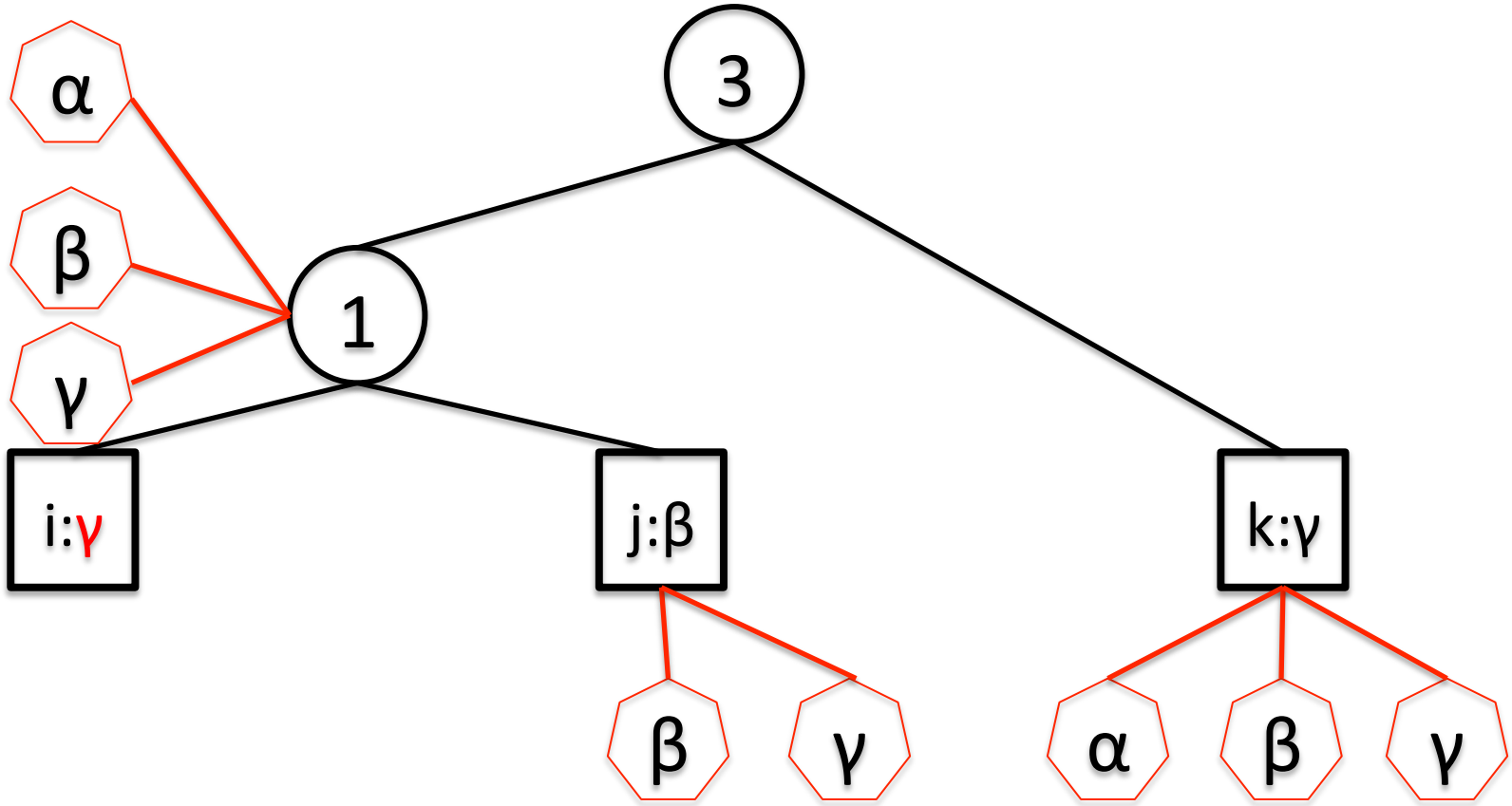
(in polynomial number of steps)



Our Results on Hierarchical Networks

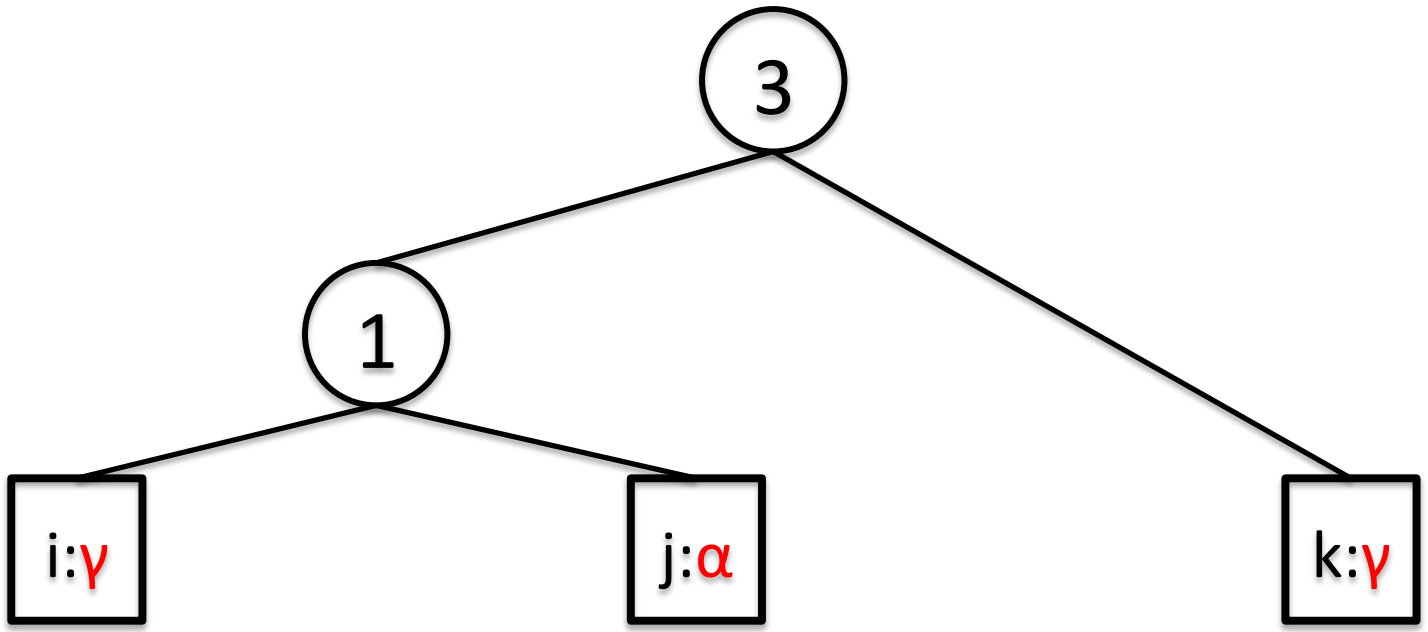
Equilibrium

(in polynomial number of steps)



Our Results on Hierarchical Networks

Equilibrium



Concluding Remarks

- Numerical utilities -> General framework
- Fractional version: nodes hold fractions of objects.

Open Questions

- Open complexity questions: finding a pure Nash equilibrium for undirected networks with binary preferences.
- Dynamically changing networks.
- Finding approximate equilibria (mixed or correlated).