

# Network communities and spanning trees

Optimisation Techniques for Computer Science

Guillaume-Jean Herbiet <[guillaume.herbiet@uni.lu](mailto:guillaume.herbiet@uni.lu)>

Apivadee Piyatumrong <[apivadee.piyatumrong@uni.lu](mailto:apivadee.piyatumrong@uni.lu)>

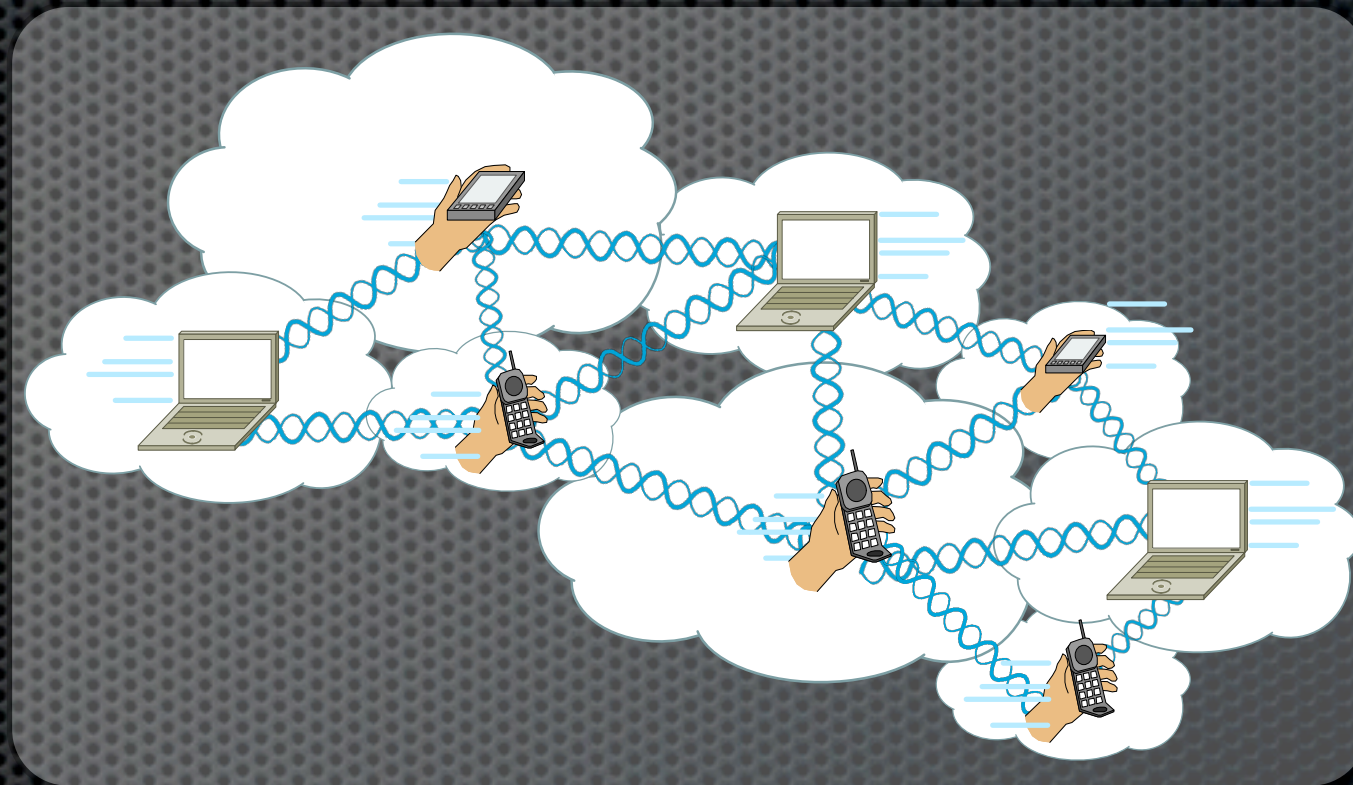


UNIVERSITÉ DU  
LUXEMBOURG



# Mobile ad-hoc networks

- ✧ All nodes are **mobile**
- ✧ **No infrastructure** is required
- ✧ All nodes act as **routers**
- ✧ Information is relayed from node to node using **multihop wireless paths**
- ✧ Nodes have to **cooperate**
- ✧ **Self-configuring** and **self-organizing** networks





# Use cases of MANETs

- ✦ Smart devices / Ubiquitous computing
- ✦ Vehicular ad hoc networks
- ✦ Disaster relief situations
- ✦ Wireless sensor networks
- ✦ Digital battlefield



# MANET challenges

- ✦ Dynamic network topology change due to nodes mobility
- ✦ All decisions should be distributed
- ✦ Limited bandwidth for communication
- ✦ Channel usage should be efficiently used
- ✦ Volume of signalization (overhead) should be limited
- ✦ Efficient power management
- ✦ Misbehaving/selfish nodes

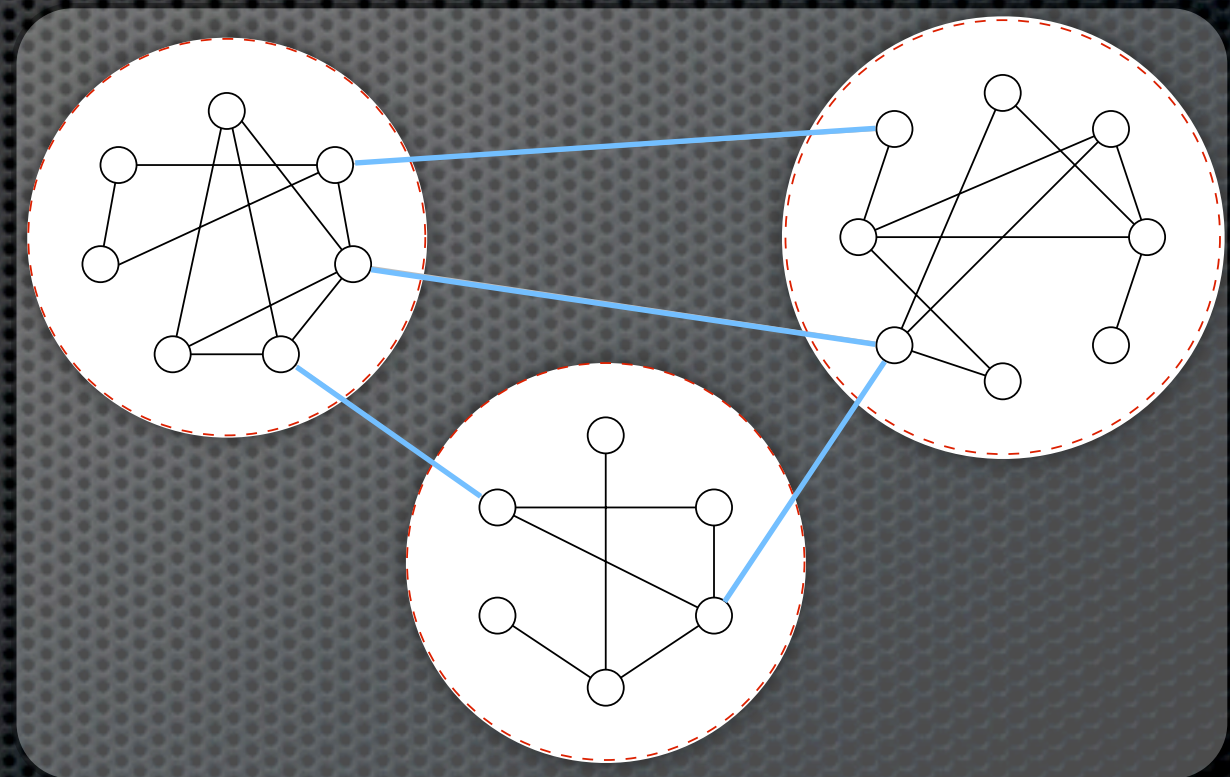
Still maintain an acceptable quality of service



# Network communities

*“A community is a subgraph in which the density of internal connections is larger than the connections with the rest of nodes in the network”*

- ✦ Detecting communities is a **NP-hard** problem
- ✦ Most approaches are non-local (i.e. not suitable for MANETS)
- ✦ Detecting network communities is an **optimization problem**



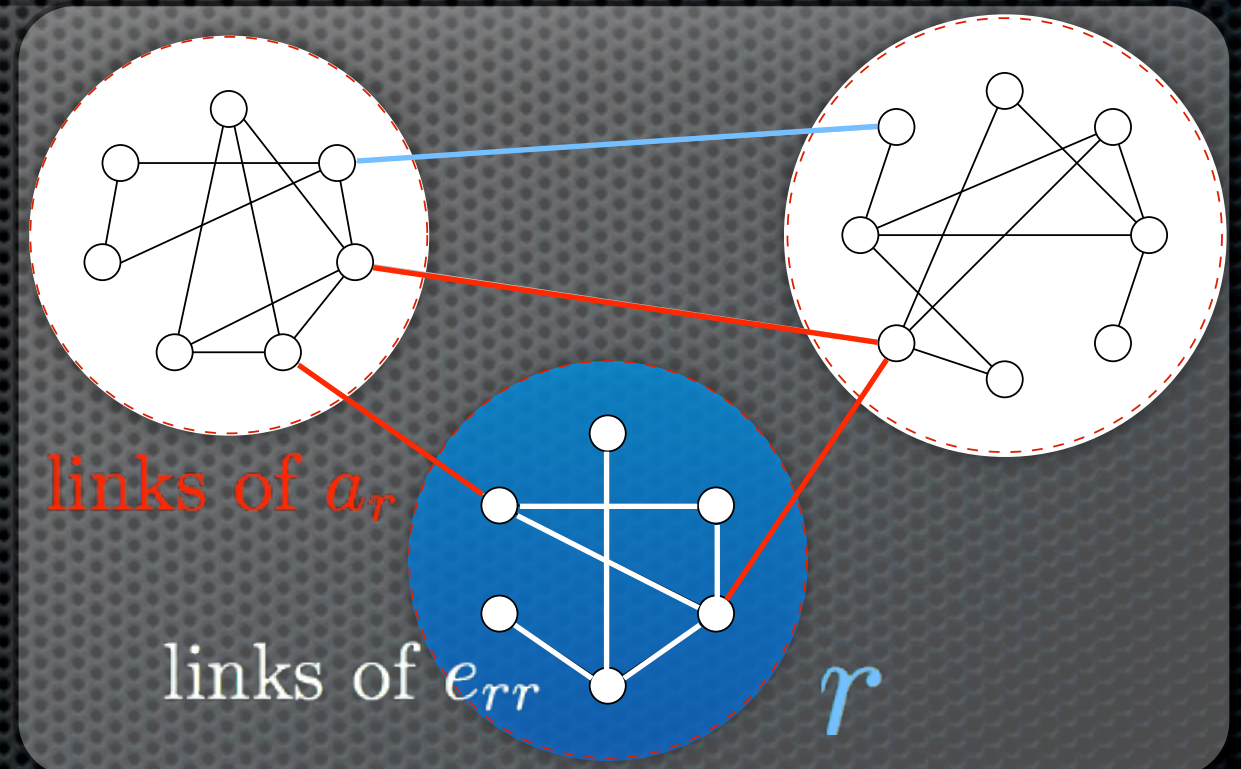


# Modularity: an example of metric for community detection

$$Q = \sum_r \left( e_{rr} - a_r^2 \right)$$

$r$ : a community  
 $e_{rr}$ : fraction of links between two nodes of the community  $r$   
 $a_r$ : fraction of links going in/out community  $r$

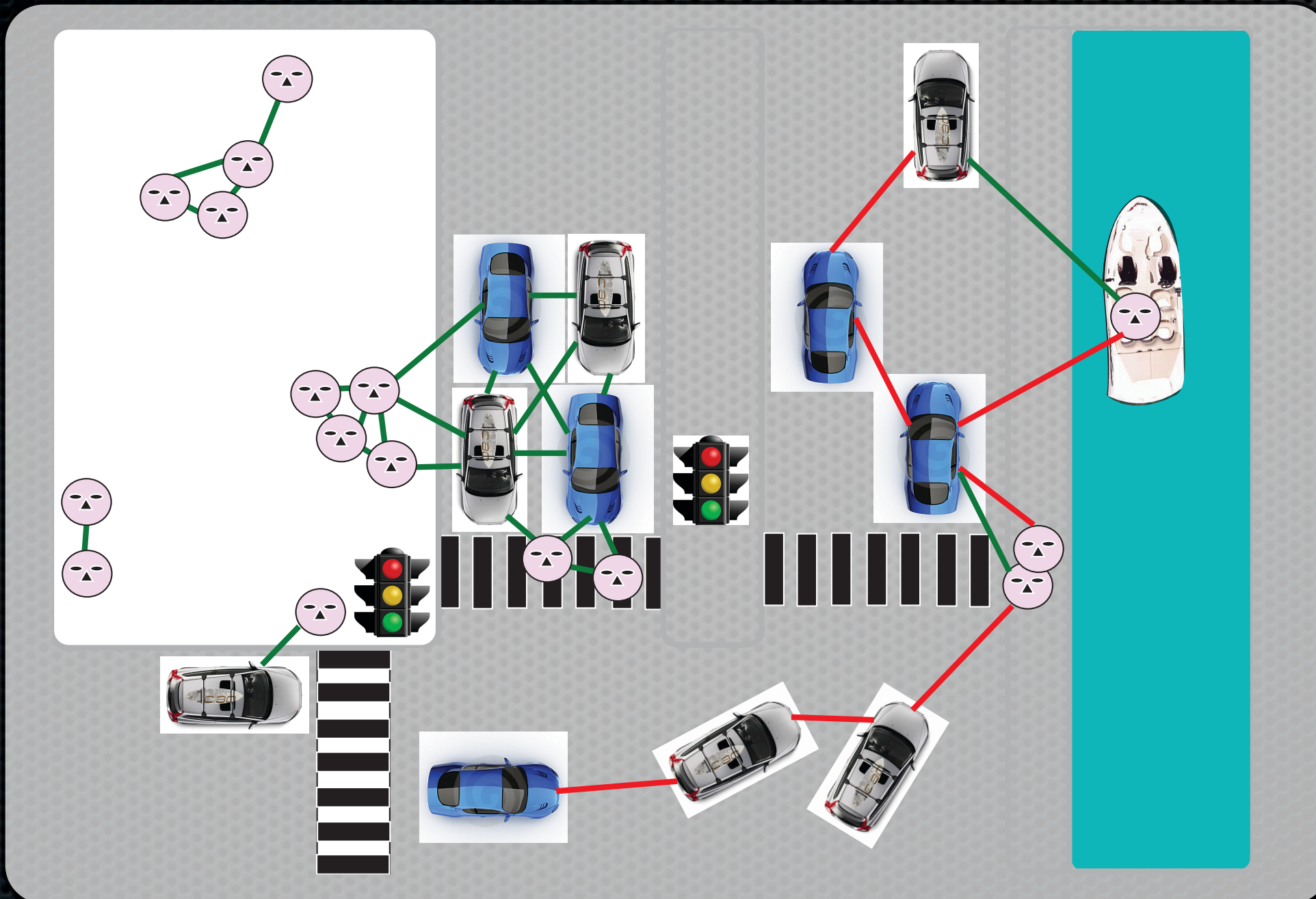
- ✦ Introduced by Newman and Girvan<sup>1</sup>
- ✦ Higher  $Q$  values correspond to accurate community partition
- ✦  $Q$  can be used in a fitness function of an optimization problem



<sup>1</sup> M. E. J. Newman and M. Girvan, *Phys. Rev. E* 69, 026113 (2004)



# DA-GRS: creating spanning trees in distributed manner





# DA-GRS: creating spanning trees in distributed manner

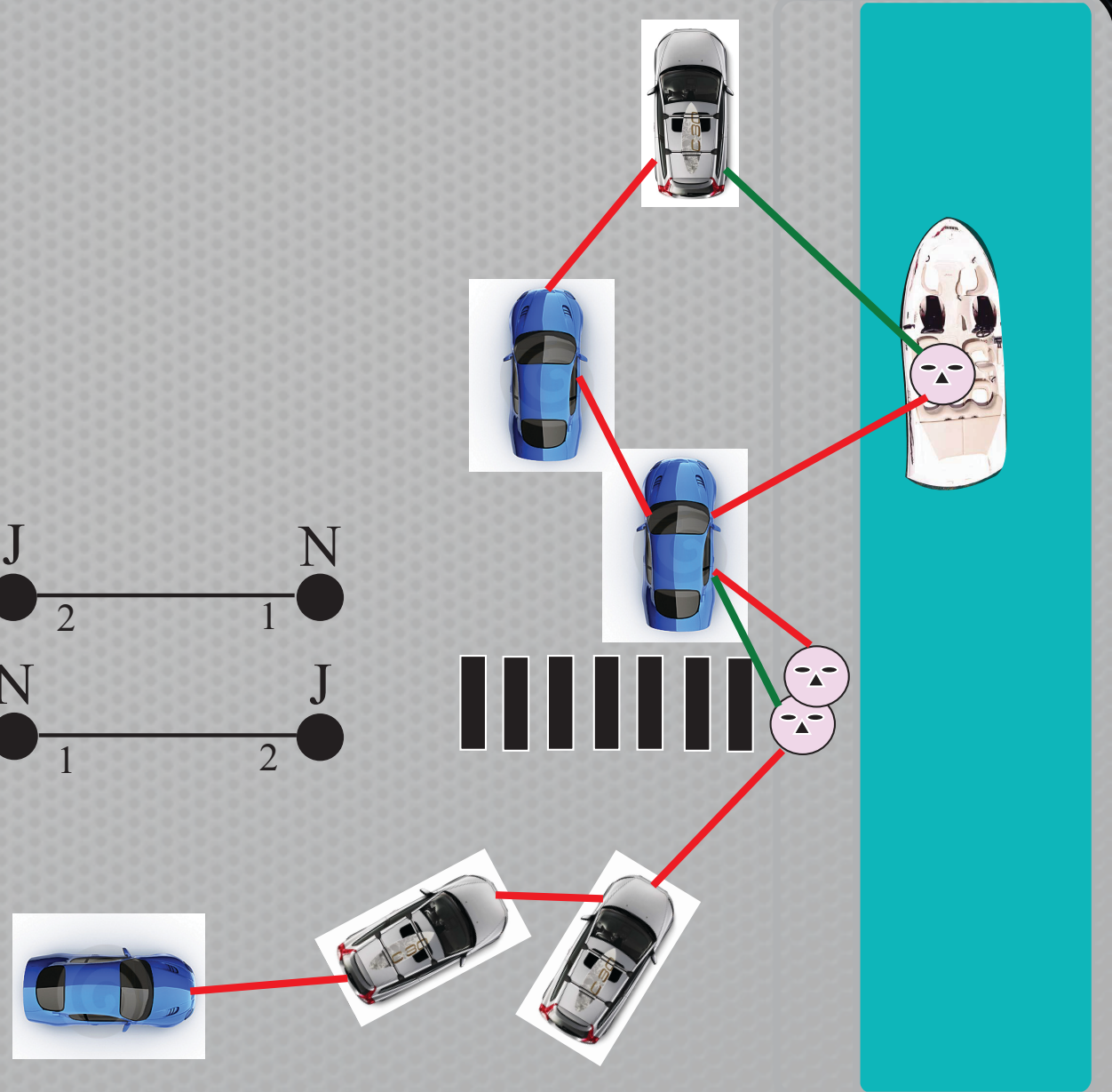
Initial label: ●<sup>J</sup>

rule1: ●<sup>N</sup><sub>1</sub><sup>off</sup> → ●<sup>J</sup>

rule2: ●<sup>Any</sup><sub>2</sub><sup>off</sup> → ●<sup>Any</sup>

rule3: ●<sup>J</sup><sub>0</sub> — ●<sup>J</sup><sub>0</sub> → ●<sup>J</sup><sub>2</sub> — ●<sup>N</sup><sub>1</sub>

rule4: ●<sup>J</sup><sub>2</sub> — ●<sup>N</sup><sub>1</sub> → ●<sup>N</sup><sub>1</sub> — ●<sup>J</sup><sub>2</sub>





# Project objectives

- ✦ Study the relation between constructed spanning tree (like DA-GRS) and network community structures.
  - ✦ Does the tree structure match/represent the community topology?
- ✦ Optimize the matching between community structures and spanning tree construction
  - ✦ Propose a heuristic for DA-GRS to optimize the matching



# Prerequisites

- ✦ JAVA programming
  - ✦ Basic background in algorithmic
  - ✦ Basic background in graph theory
  - ✦ Basic background in networking
- 
- ✦ Project requires use of a network simulator and/or graph simulator:
    - ✦ Tool usage will be taught during project



# Contacts



- ✦ Apivadee Piyatumrong  
<[apivadee.piyatumrong@uni.lu](mailto:apivadee.piyatumrong@uni.lu)>
- ✦ Office E009



- ✦ Guillaume-Jean Herbiet  
<[guillaume.herbiet@uni.lu](mailto:guillaume.herbiet@uni.lu)>
- ✦ Office E008