ERCIM SNA WG *Algorithmic Aspects of Social Network Analysis*

Guillaume-Jean Herbiet

March, 11th 2010, UCD, Dublin

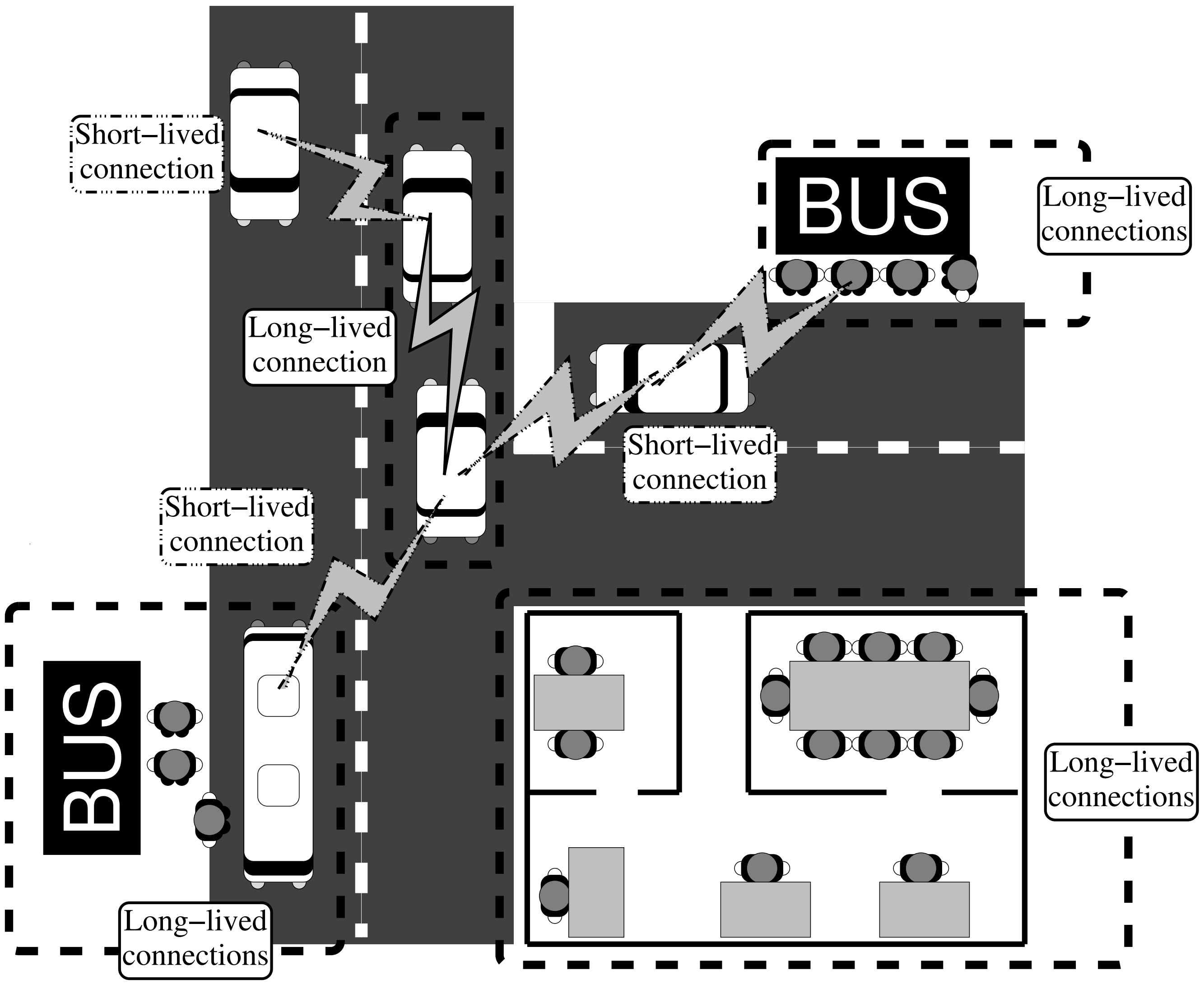
Distributed community detection over dynamic networks using neighborhood similarity

UNIVERSITÉ DU LUXEMBOURG

Rationale for this approach

Rationale for this approach

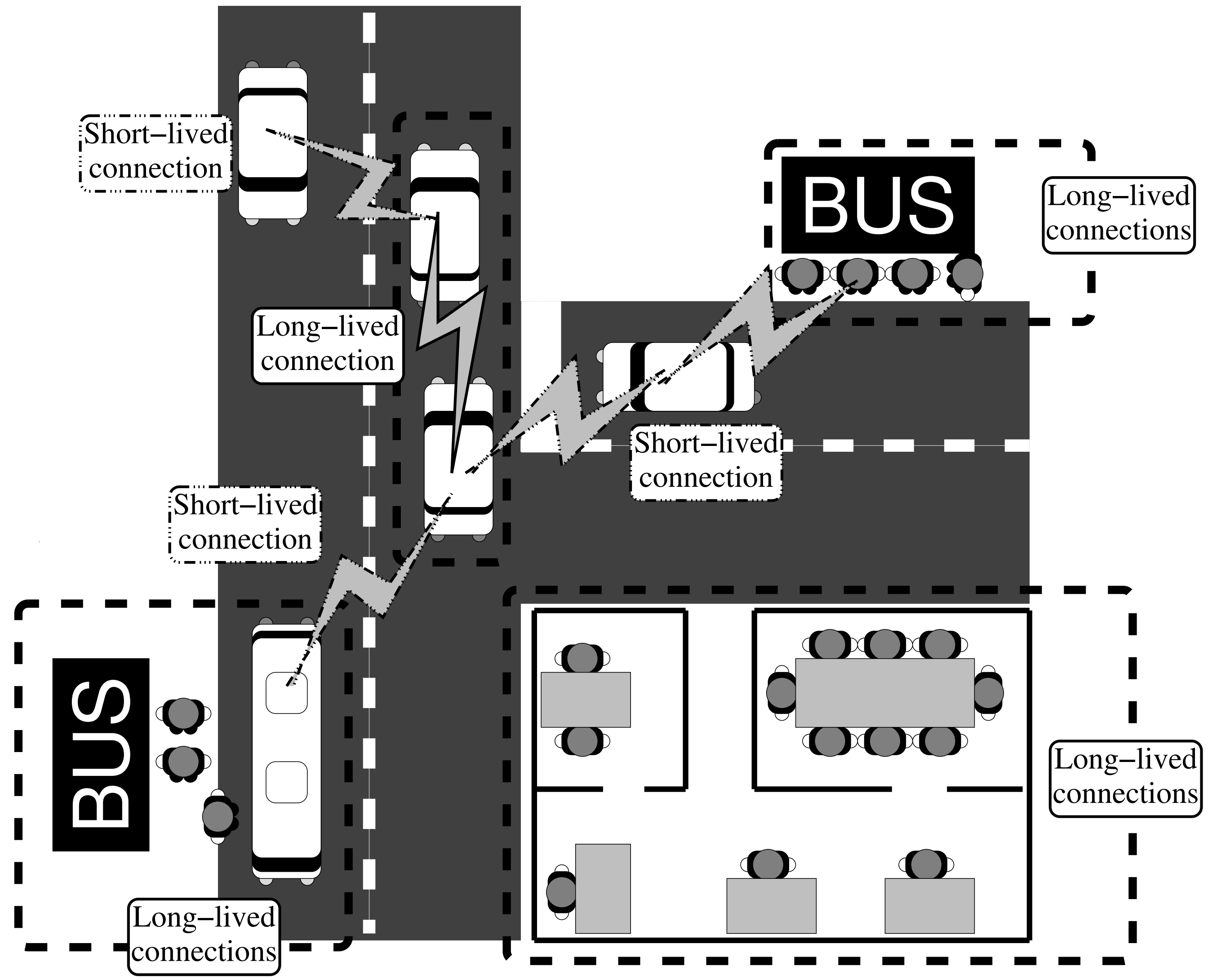
- In wireless mobile ad hoc networks **network heterogeneity** exists
  - Propagation/environment
  - Human (i.e. social) mobility patterns
  - Mutual or group shared interest



Rationale for this approach

- In wireless mobile ad hoc networks **network heterogeneity** exists
  - Propagation/environment
  - Human (i.e. social) mobility patterns
  - Mutual or group shared interest

- Community structures are helpful
  - Subgraphs of interactive and high-end communications
  - Use *weak ties* for delay-tolerant global information

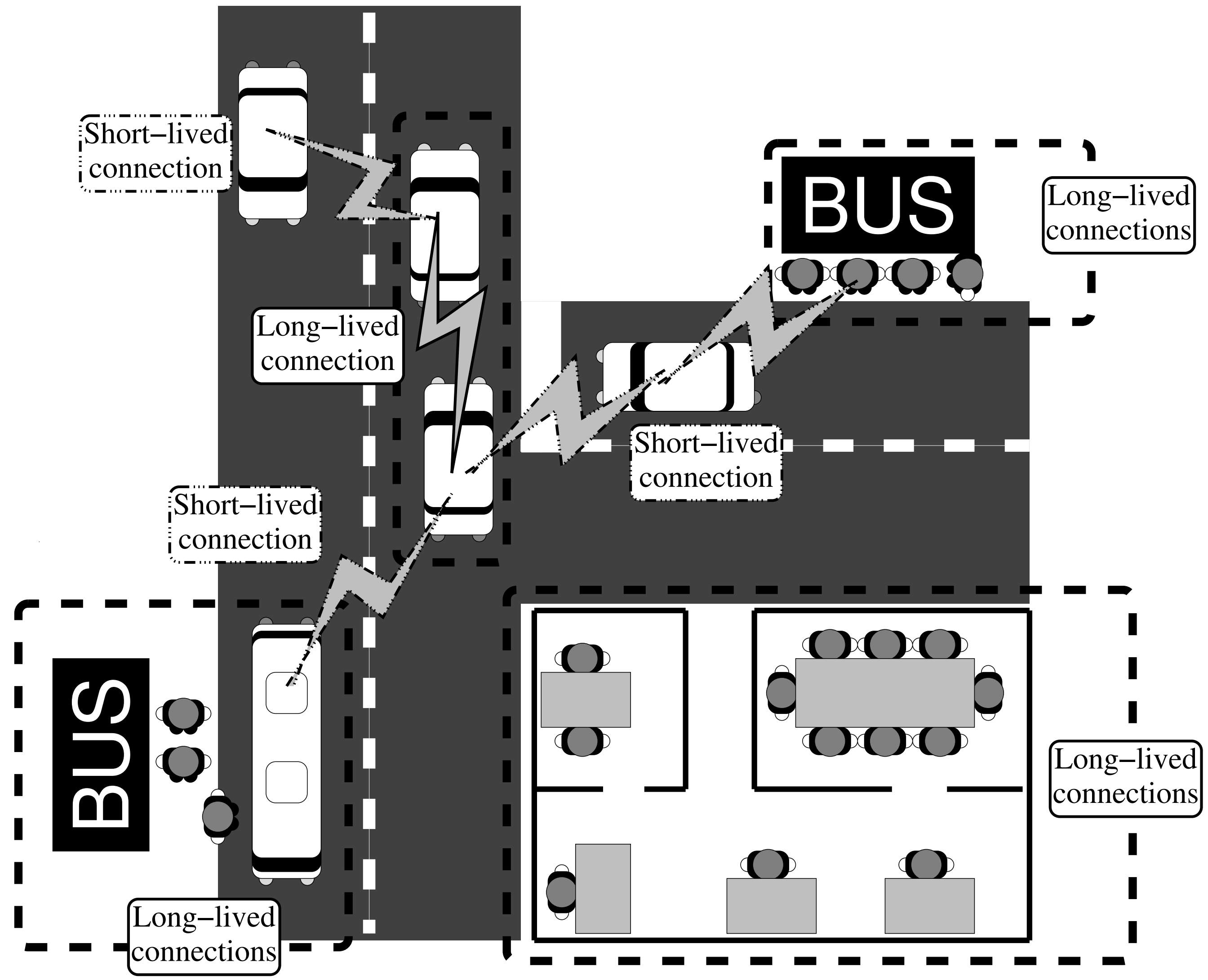


Rationale for this approach

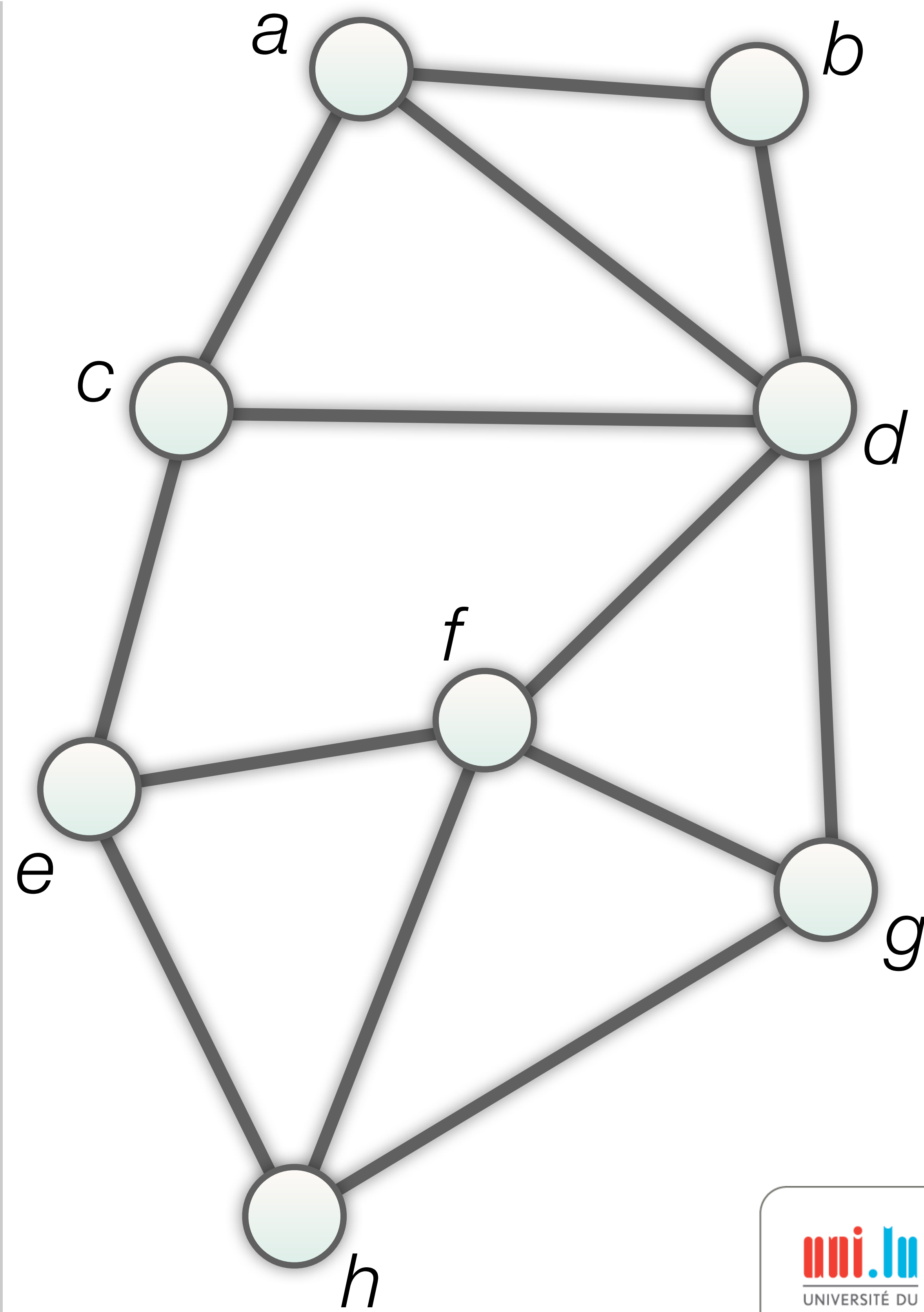
- In wireless mobile ad hoc networks **network heterogeneity** exists
  - Propagation/environment
  - Human (i.e. social) mobility patterns
  - Mutual or group shared interest

- Community structures are helpful
  - Subgraphs of interactive and high-end communications
  - Use *weak ties* for delay-tolerant global information

- **Challenges:**
  - Dynamic network
  - Distributed approach



Principle and limitation of distributed algorithms

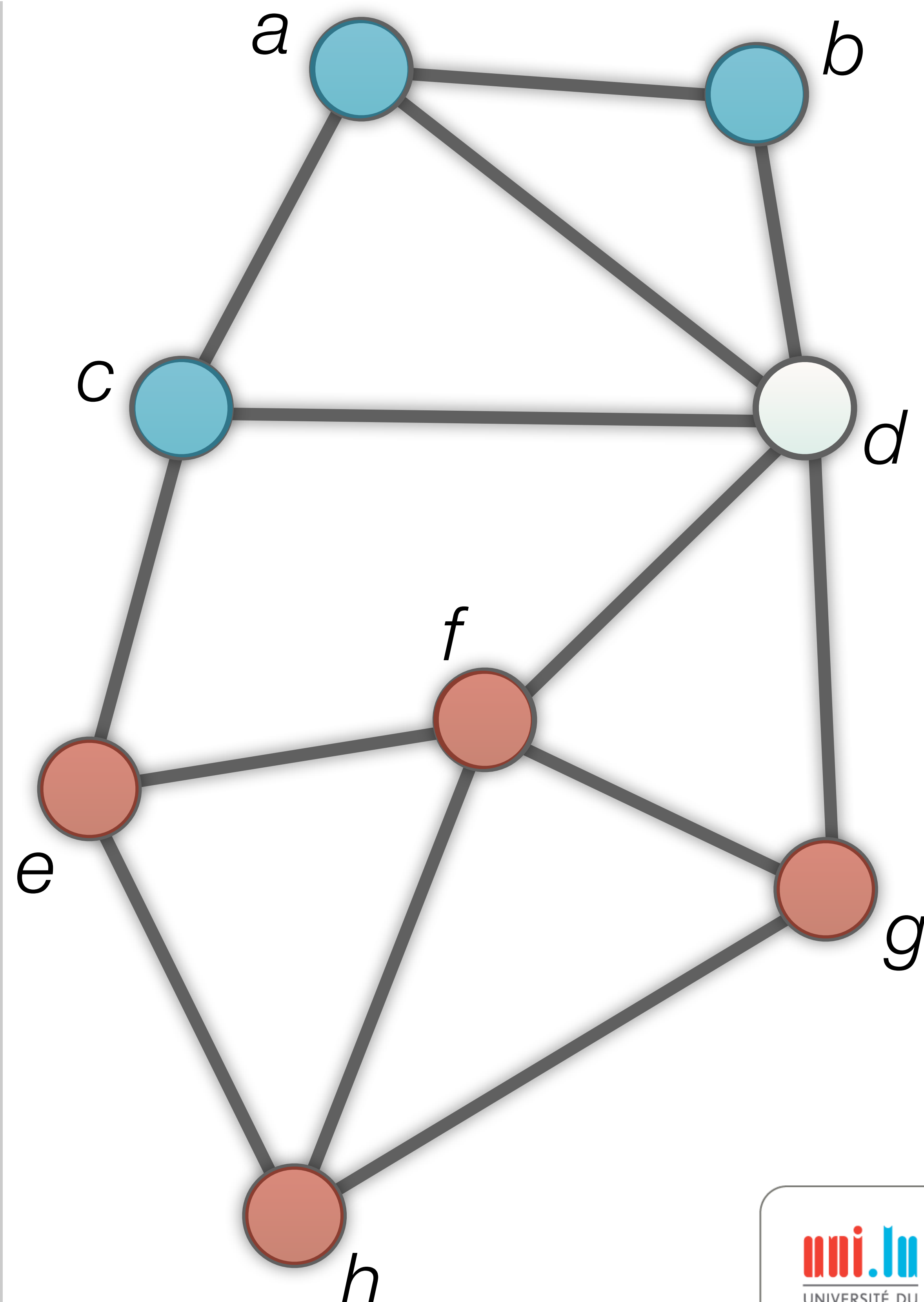


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation

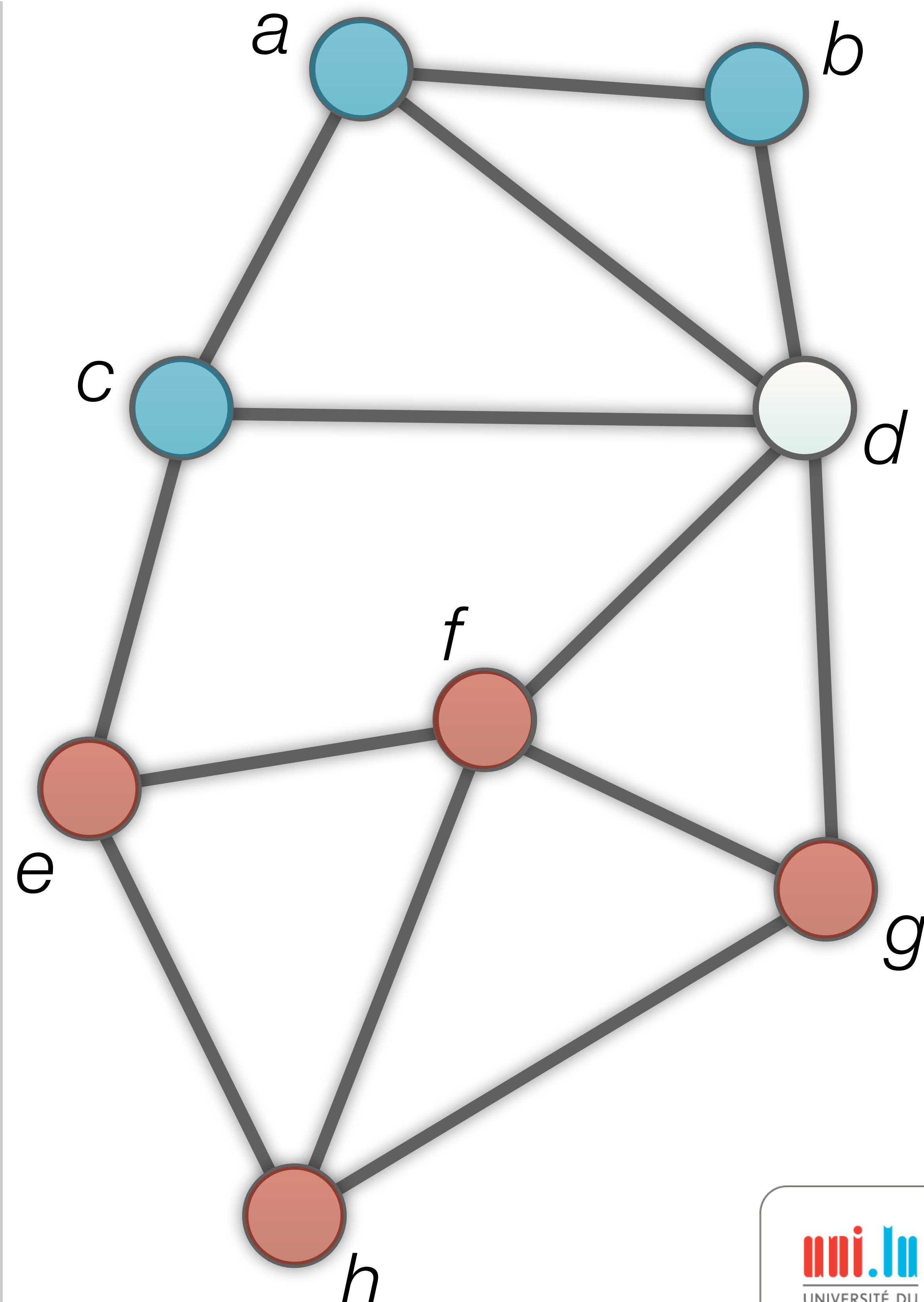


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors

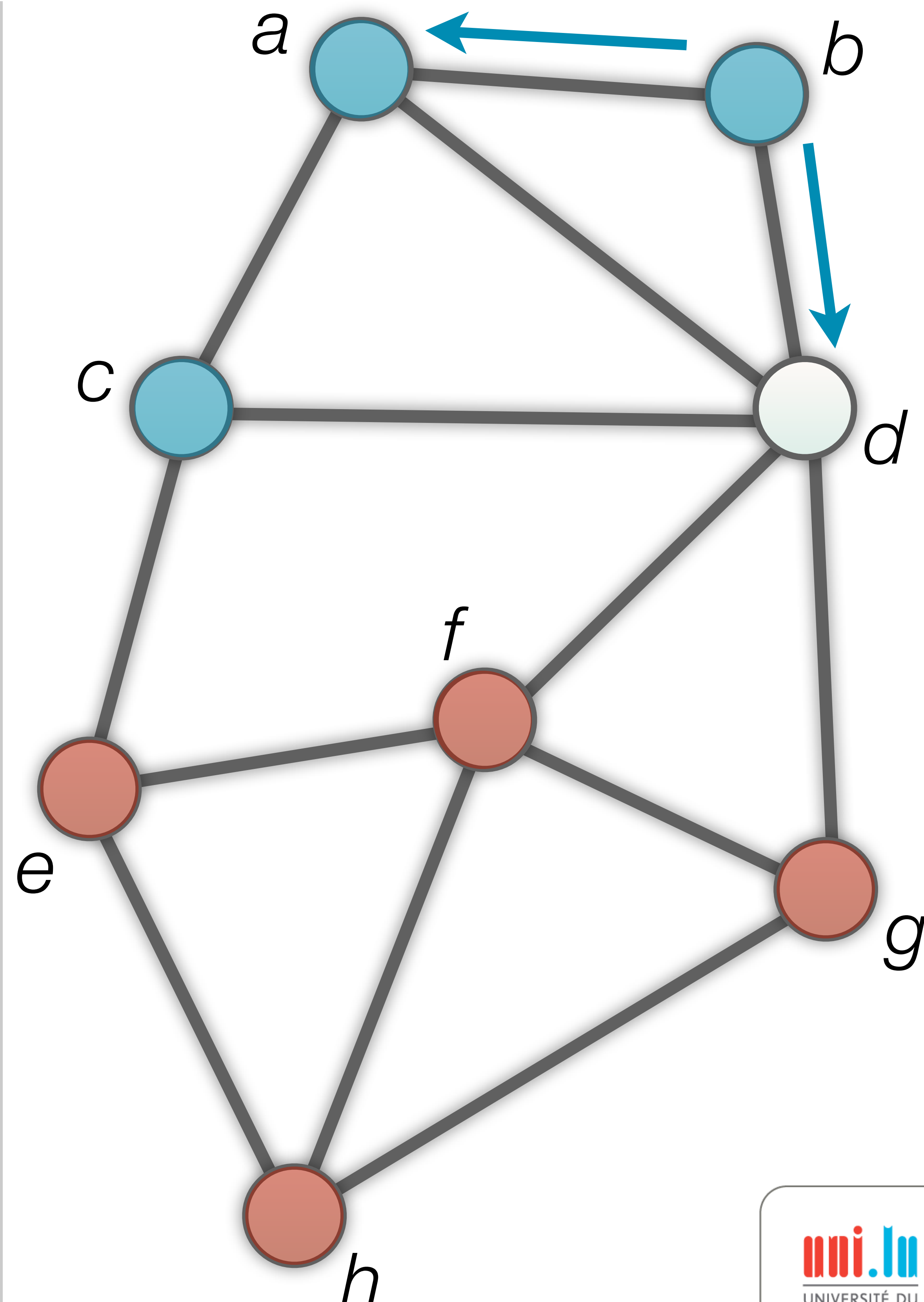


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors

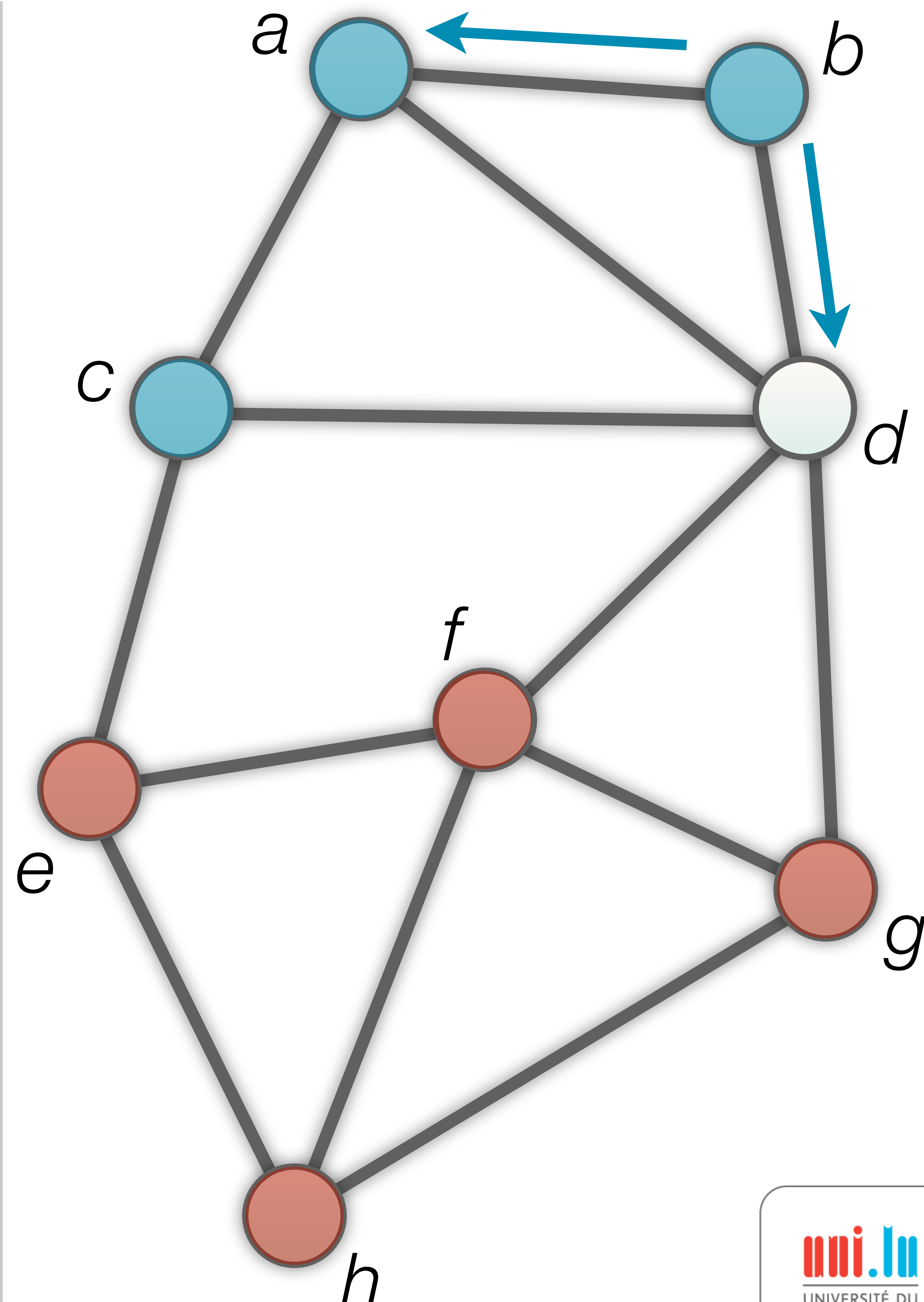


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

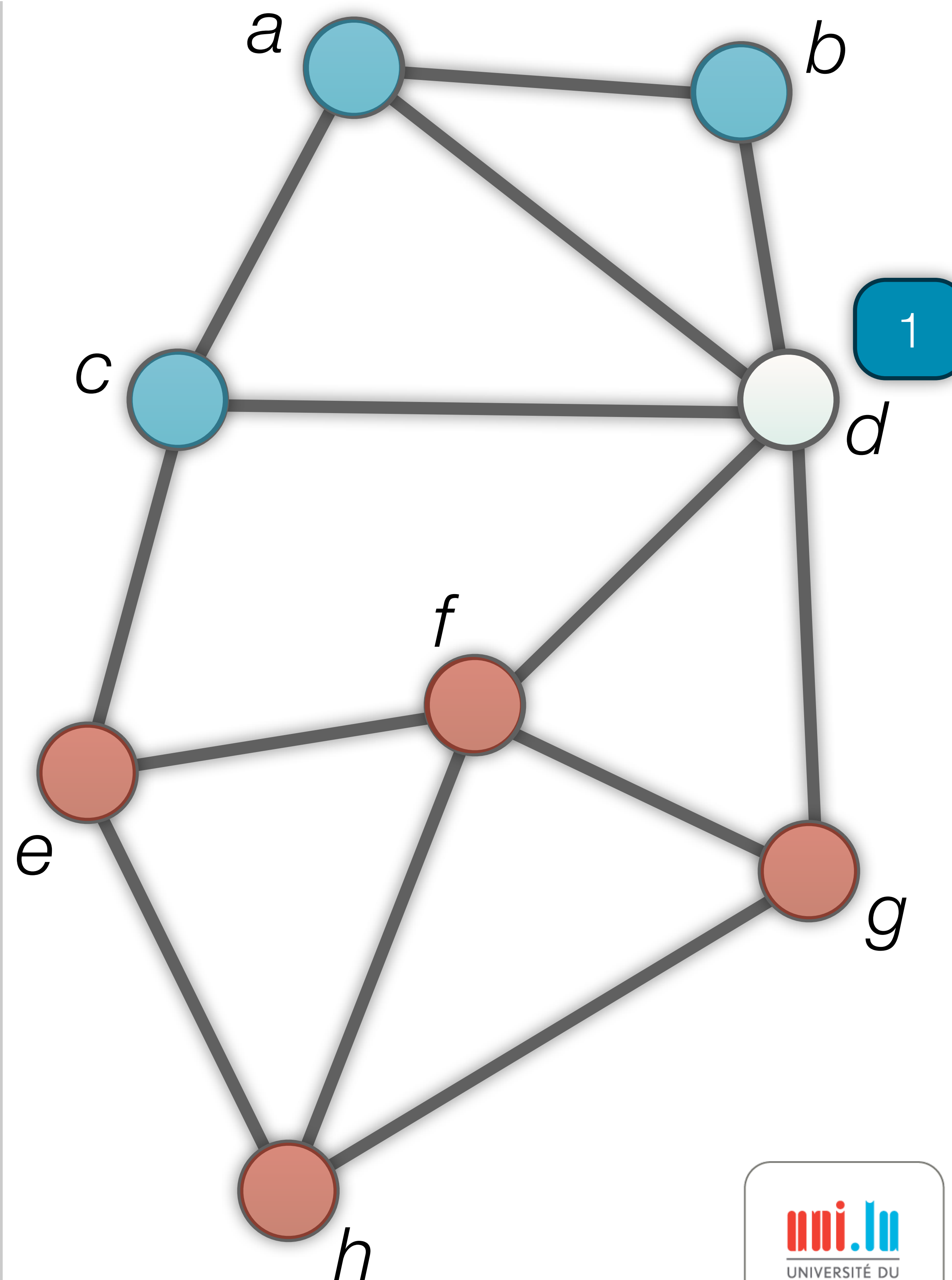


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

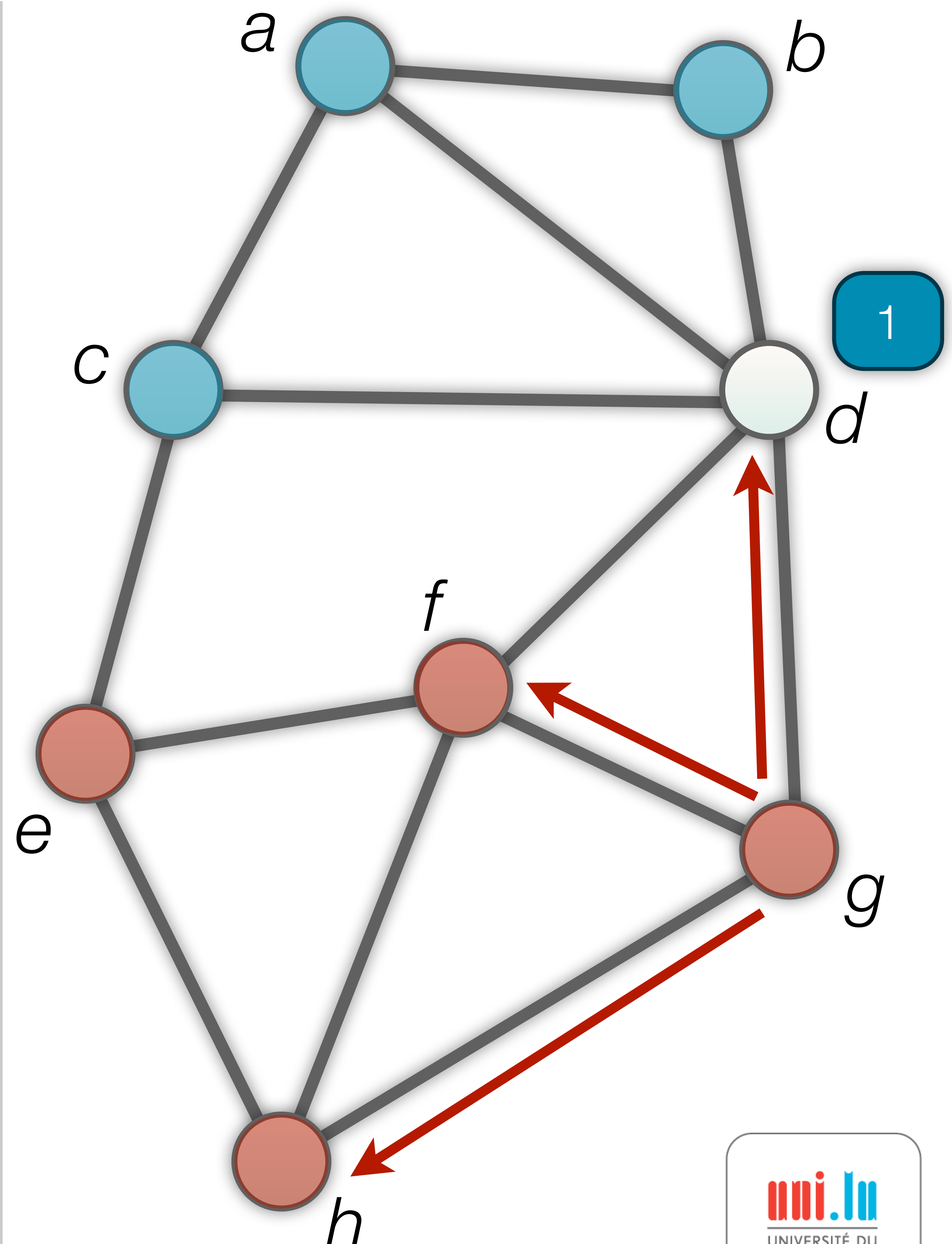


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

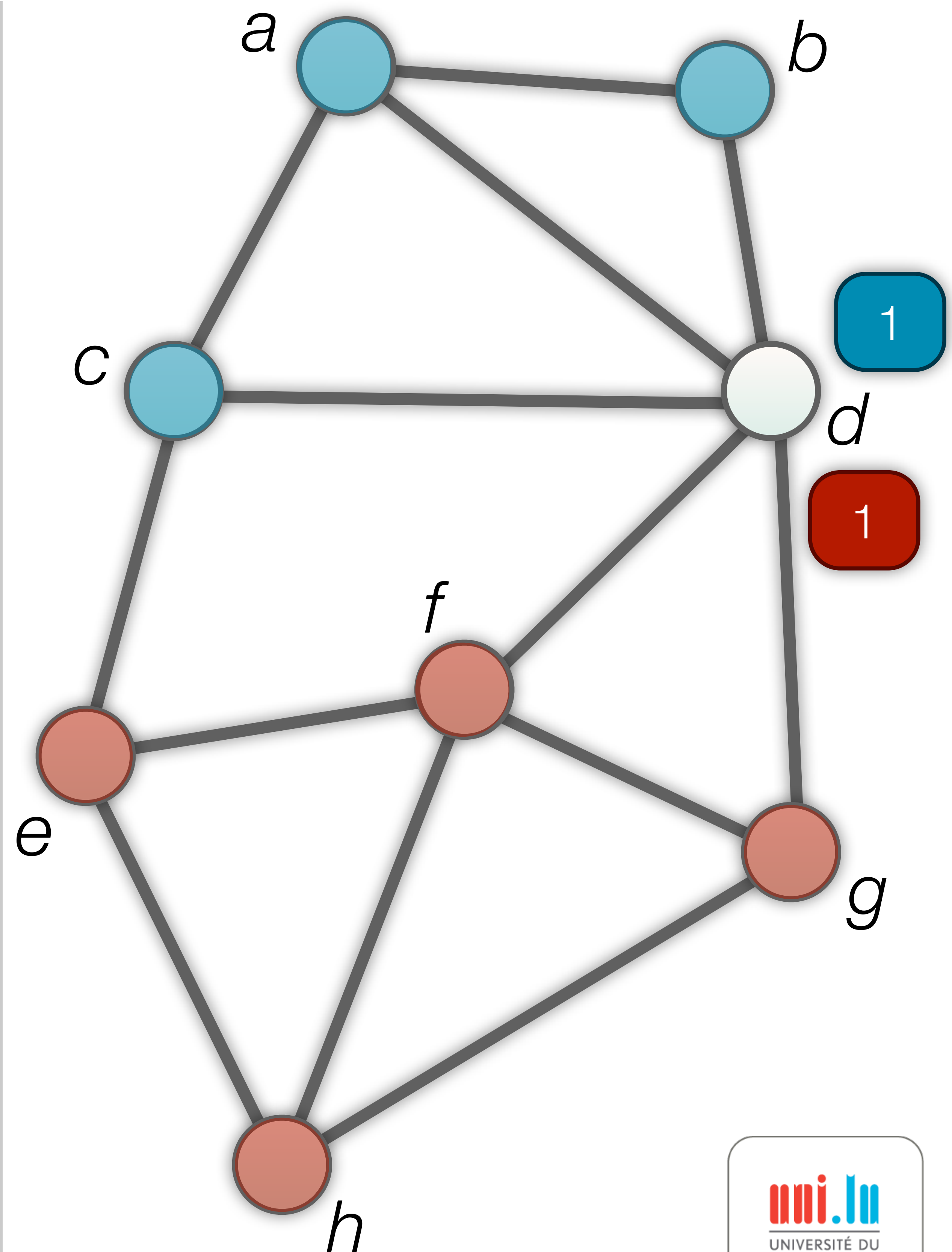


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

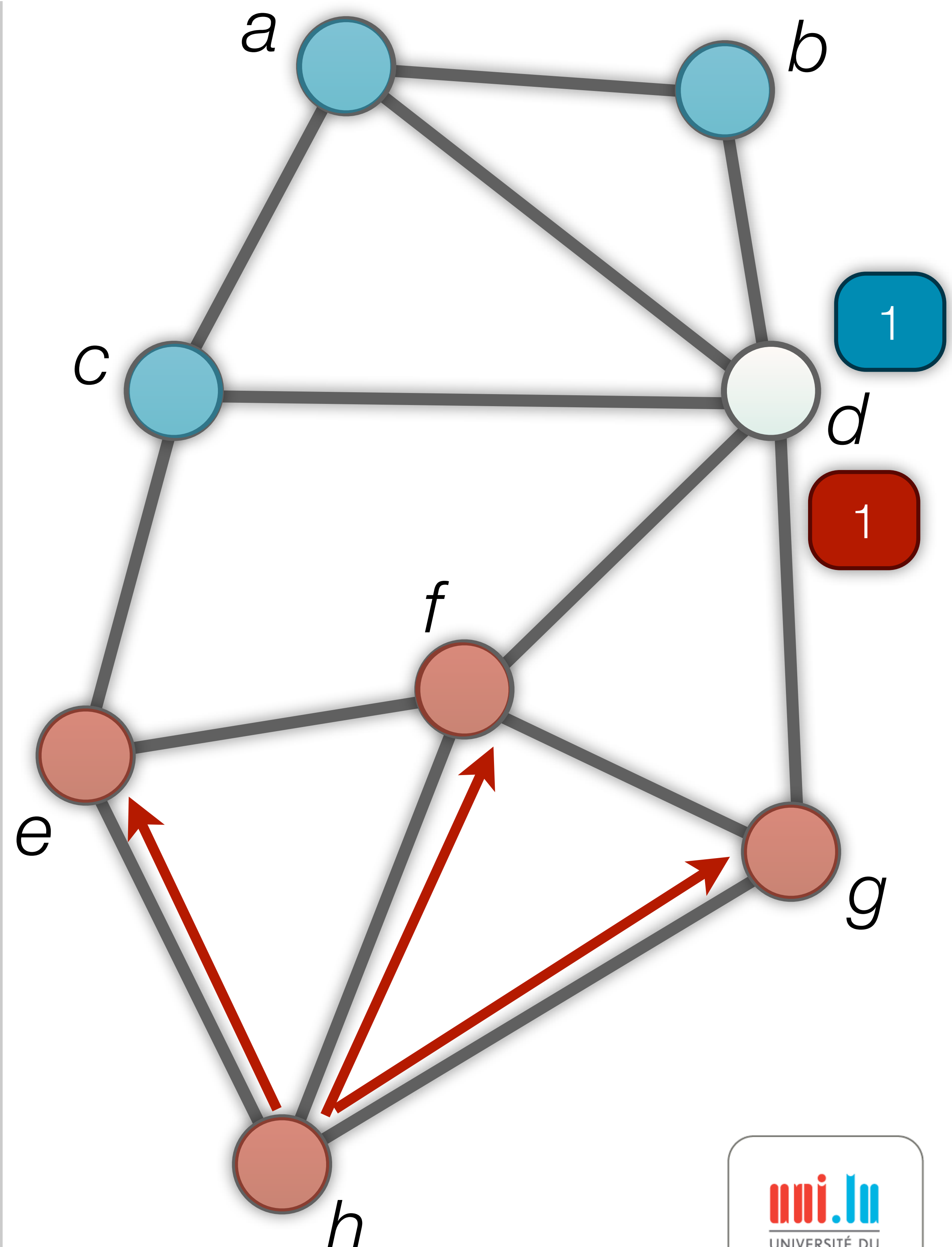


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

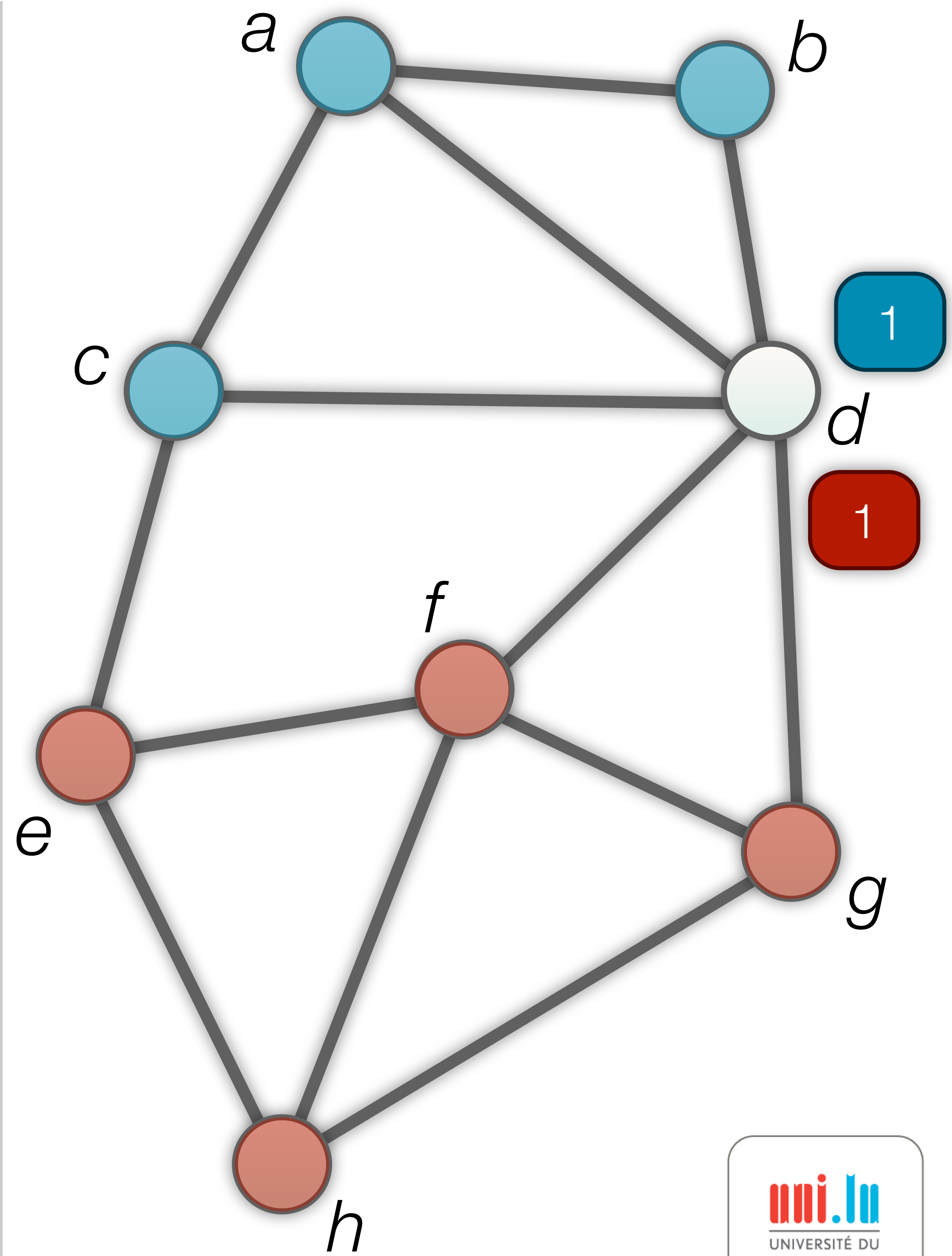


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

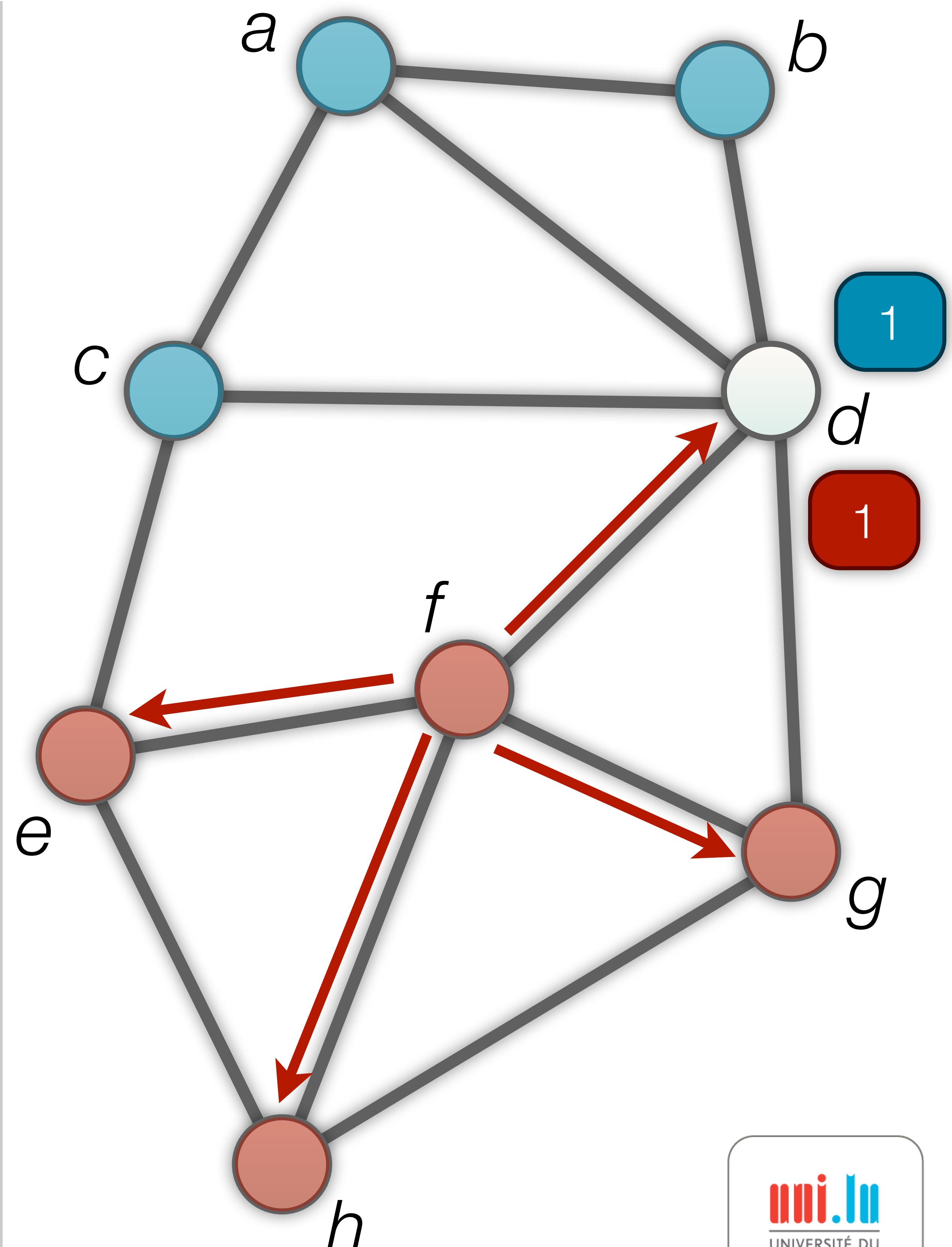


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

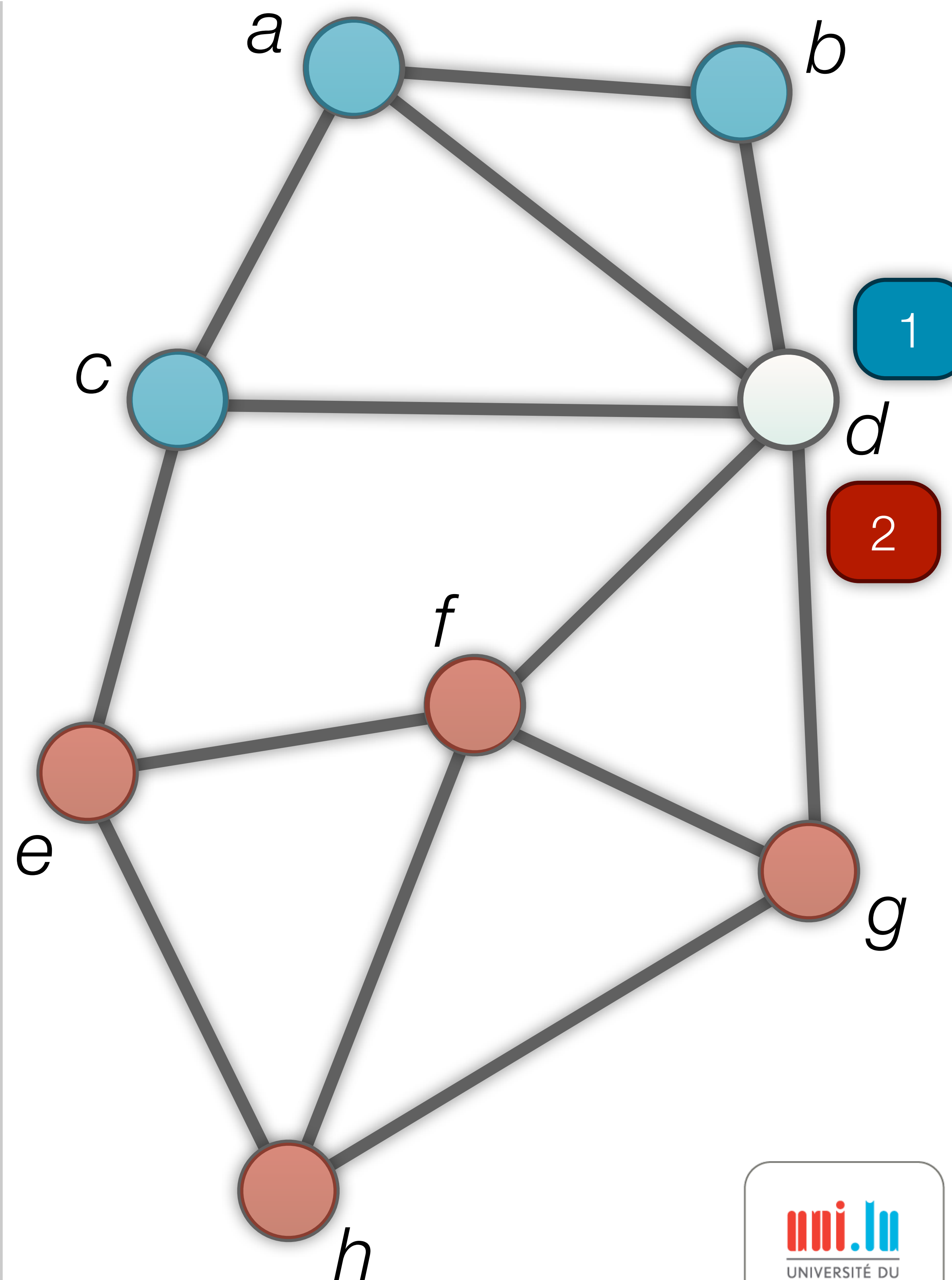


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

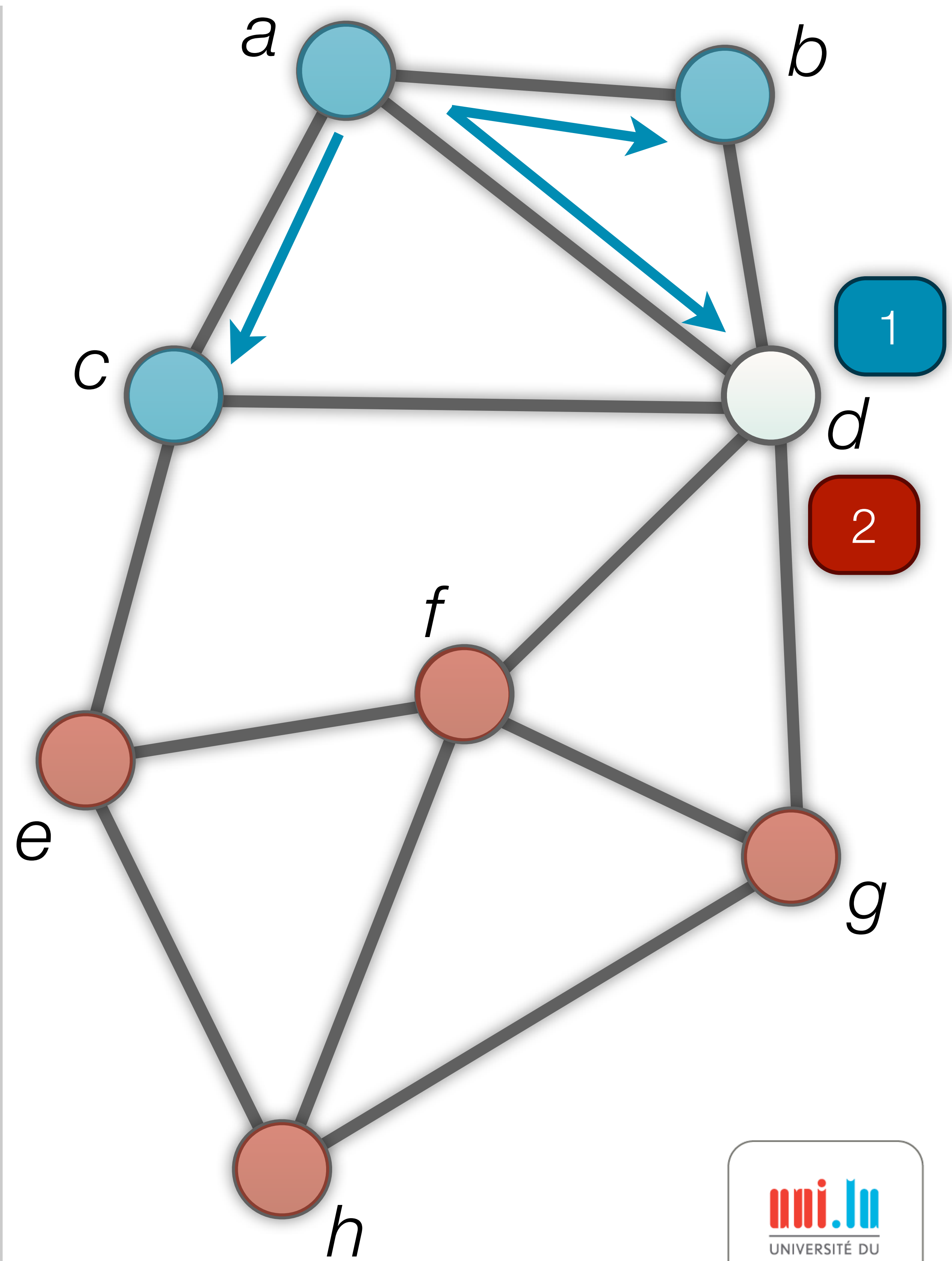


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

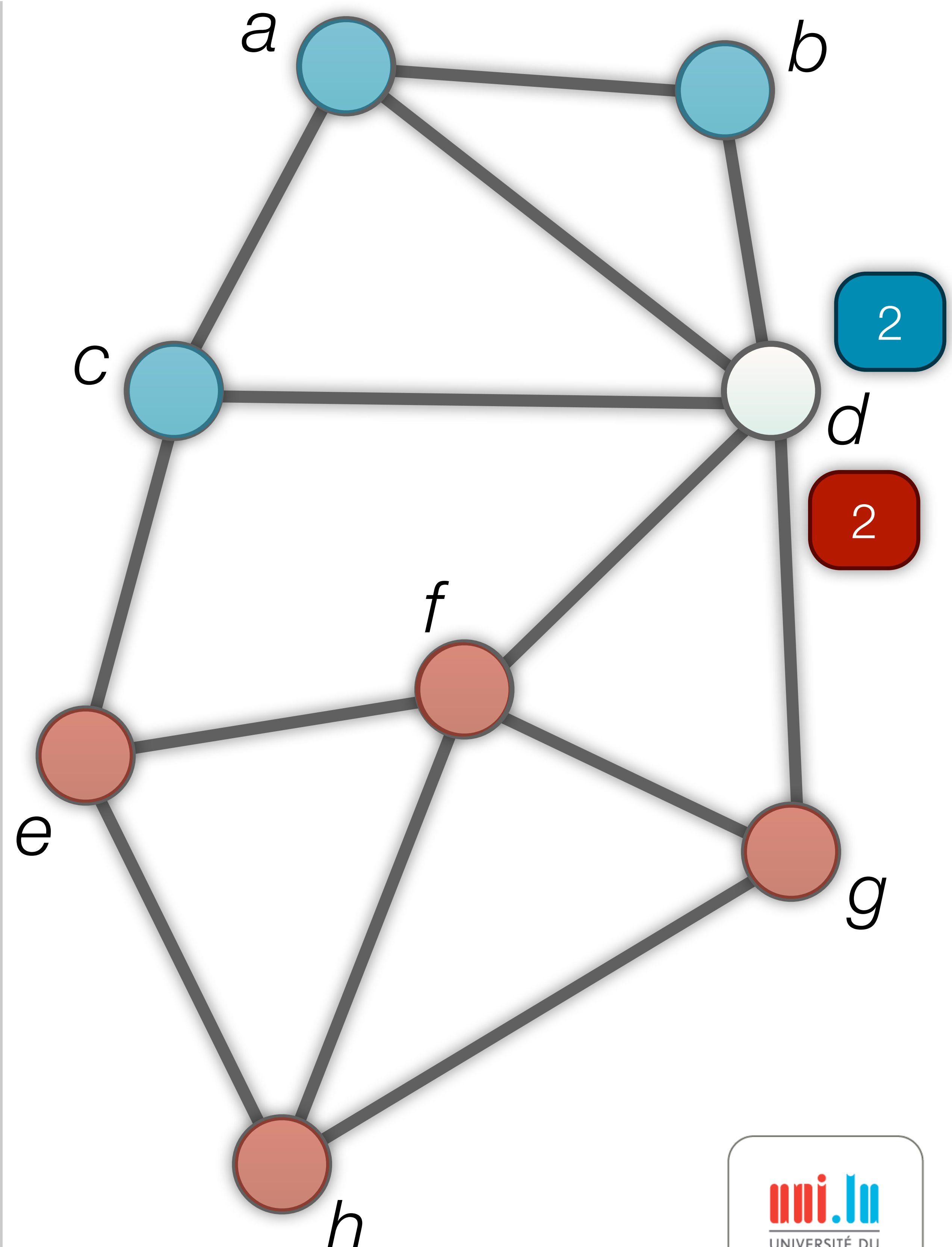


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

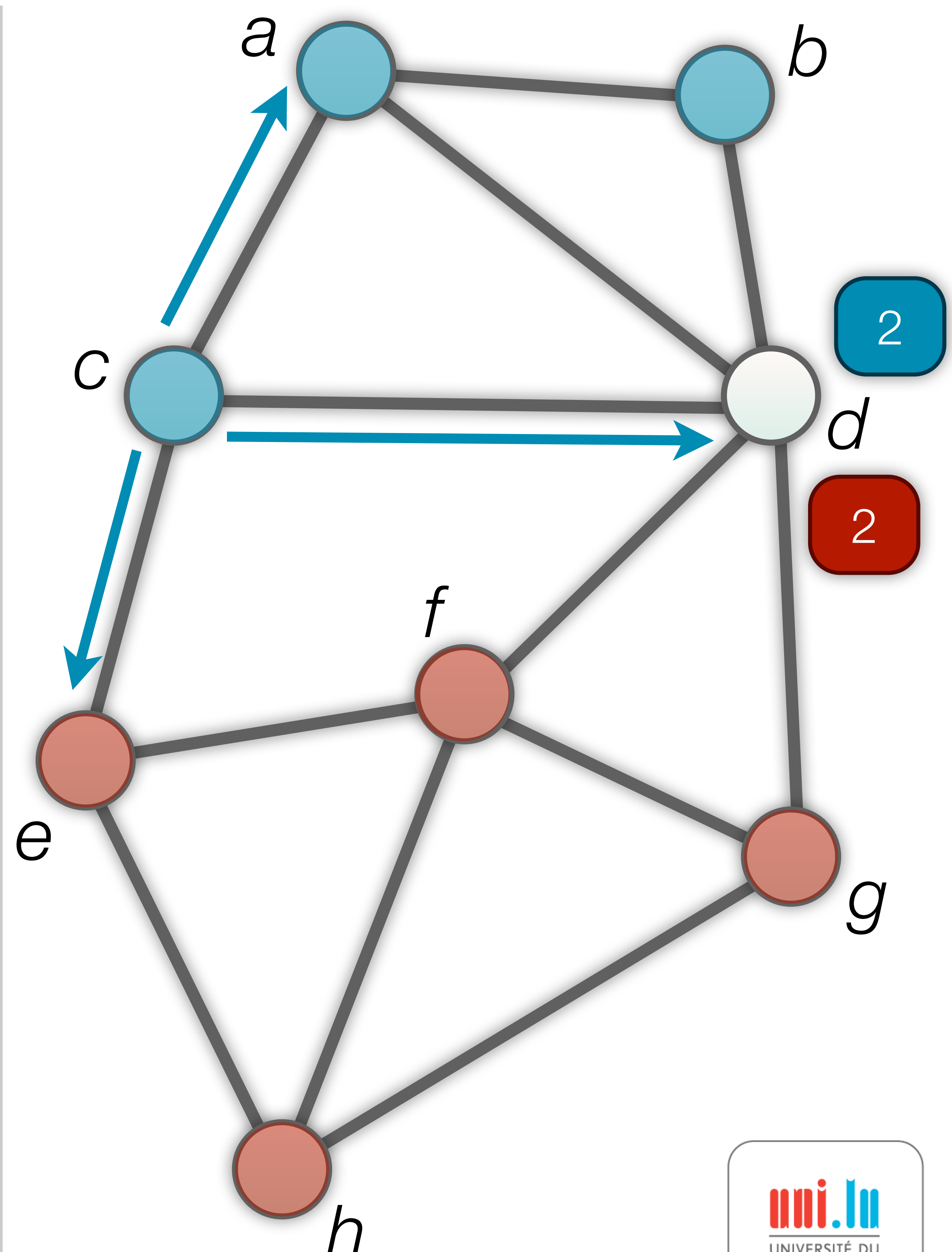


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

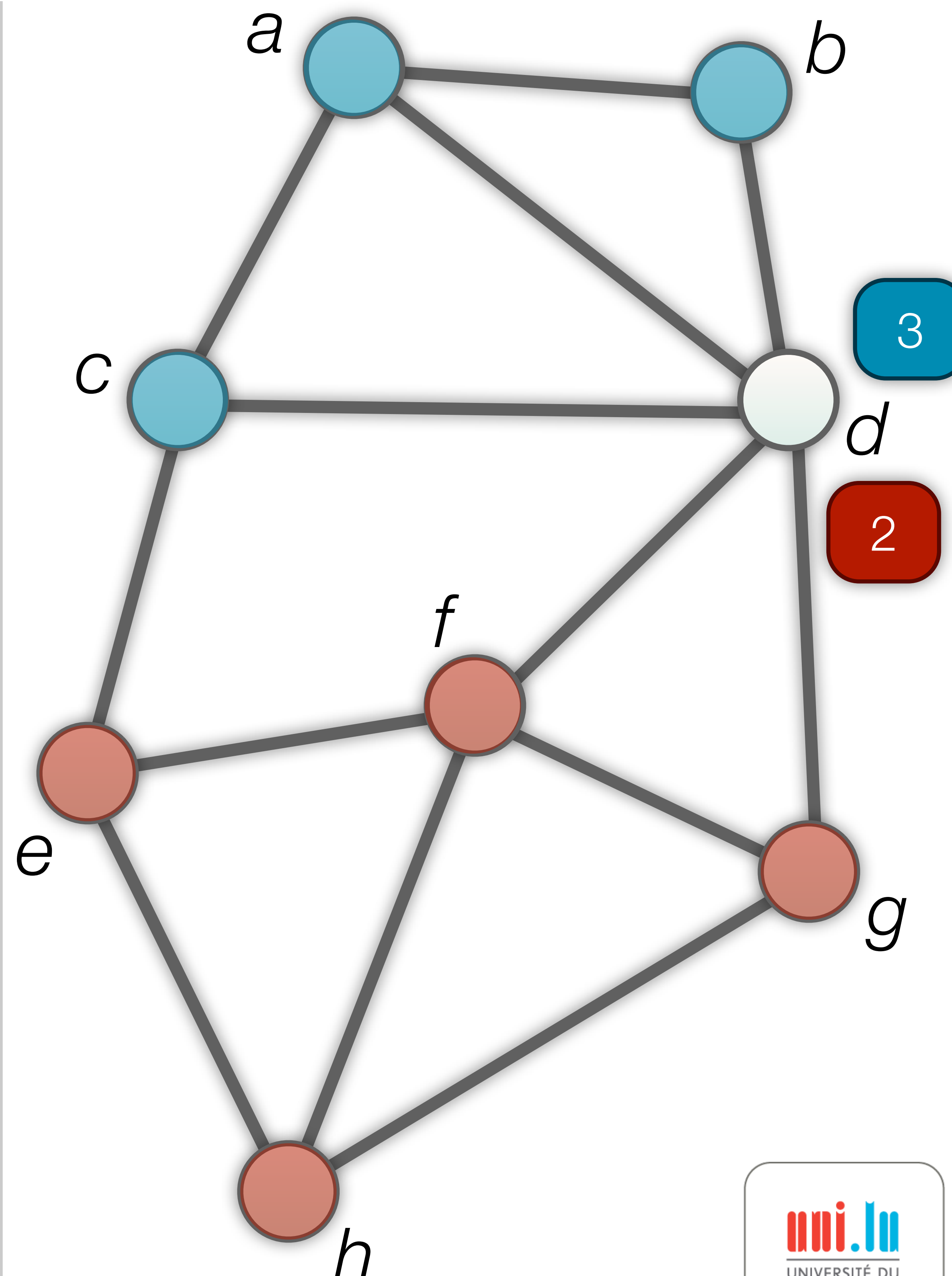


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted

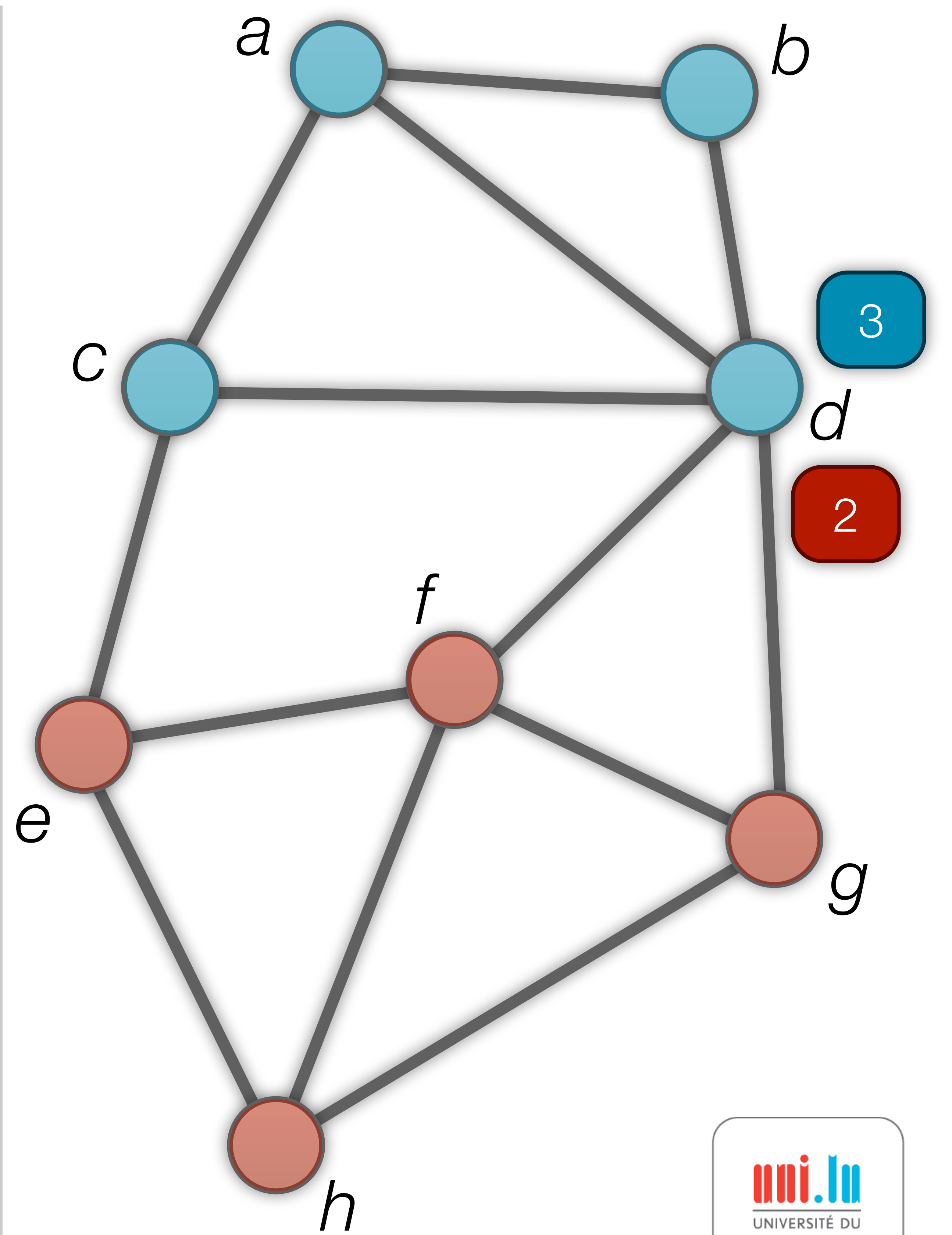


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted
  - Node changes to the label with highest count

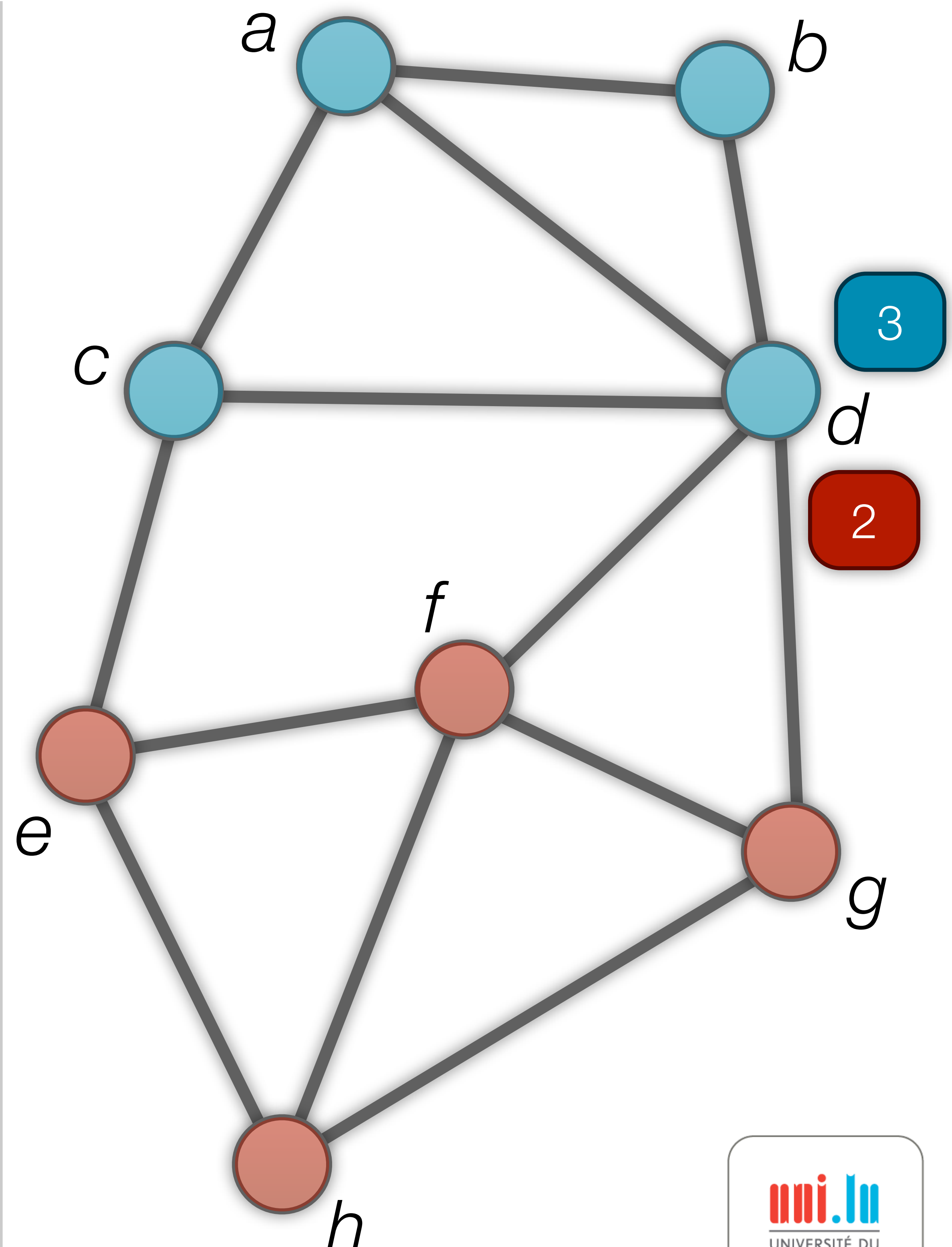


[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted
  - Node changes to the label with highest count
- Operation is (a)synchronous



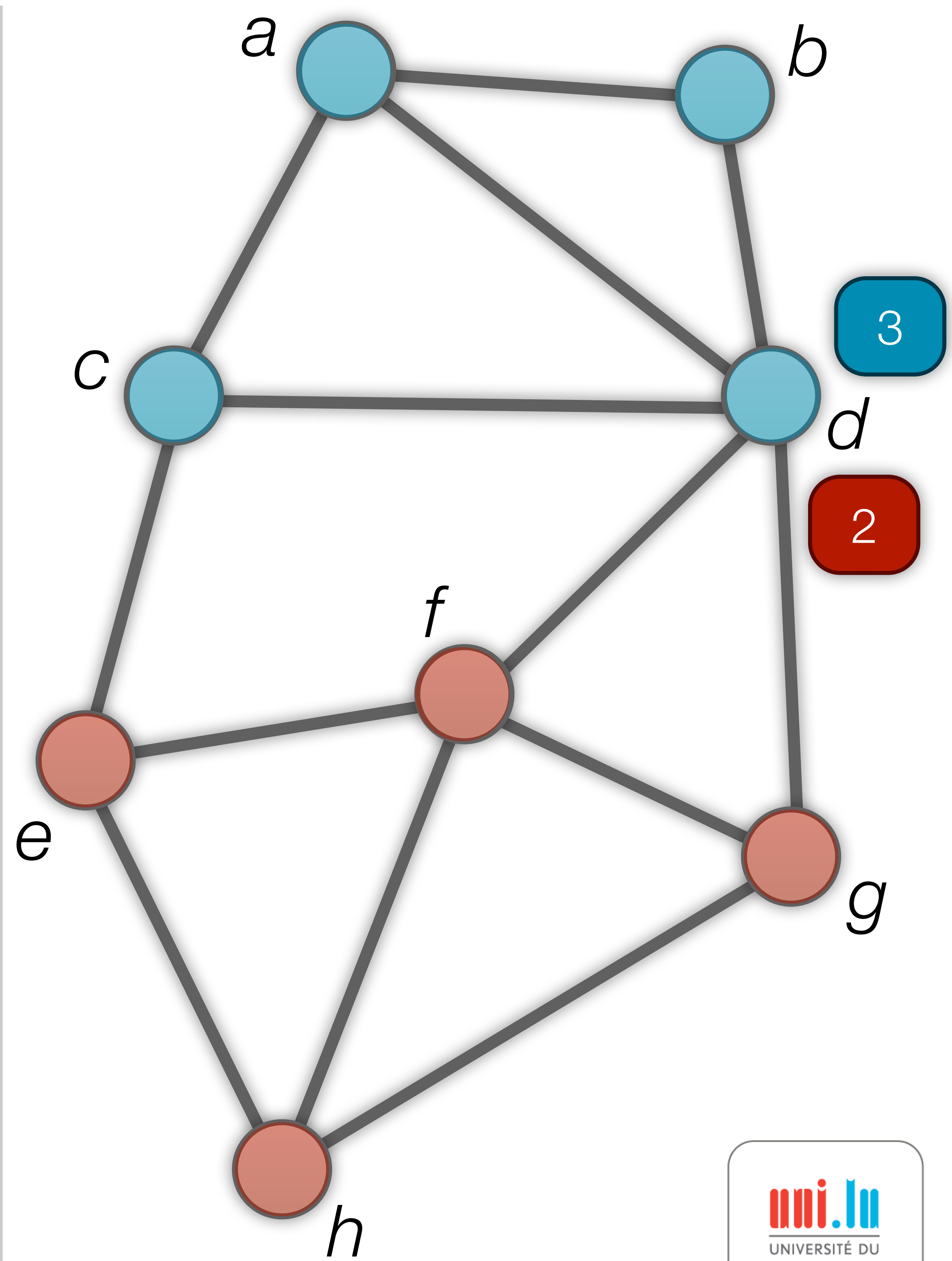
[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted
  - Node changes to the label with highest count
- Operation is (a)synchronous

- Known issues:



[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

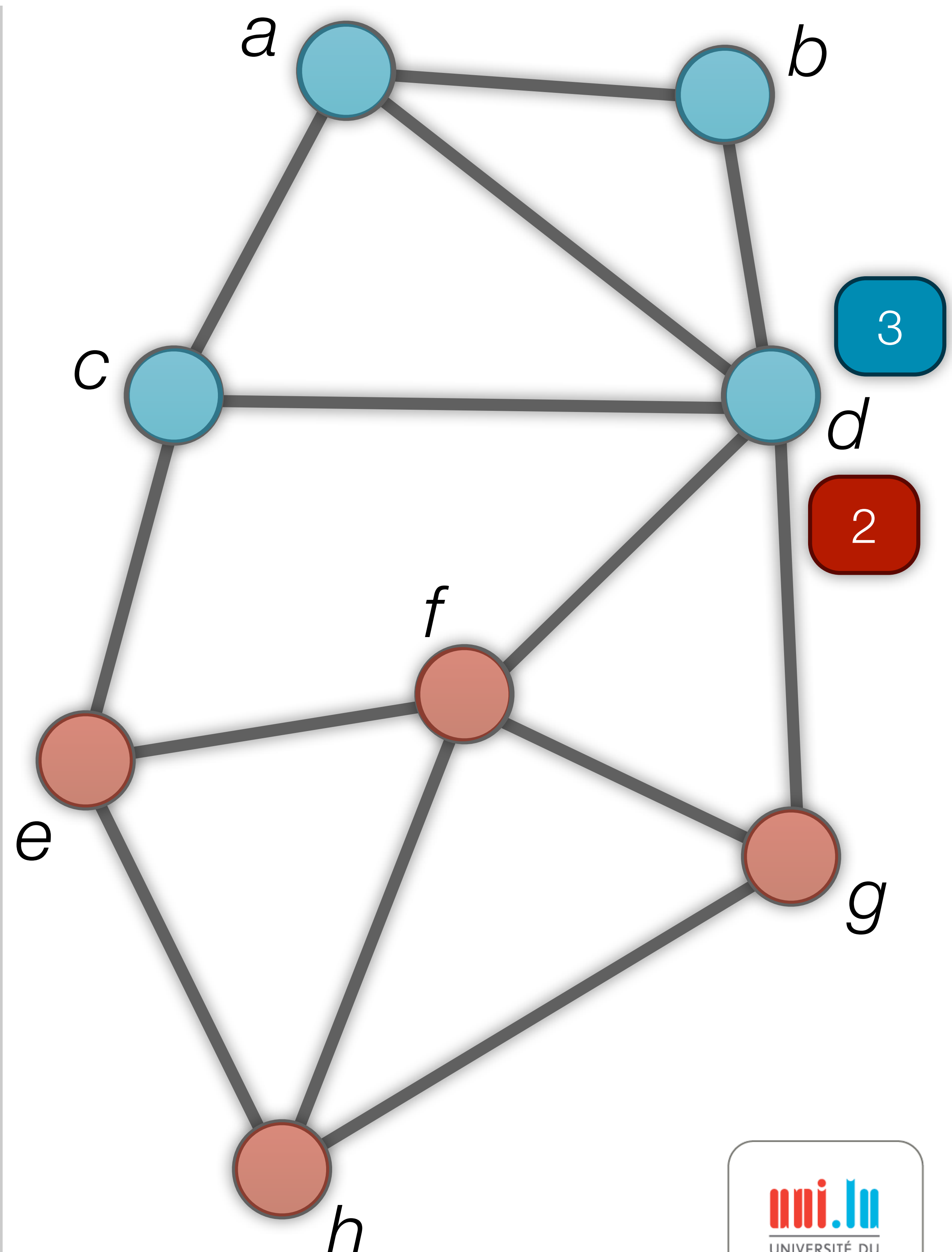
Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted
  - Node changes to the label with highest count
- Operation is (a)synchronous

- Known issues:
  - Label oscillation (synchronous)

[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

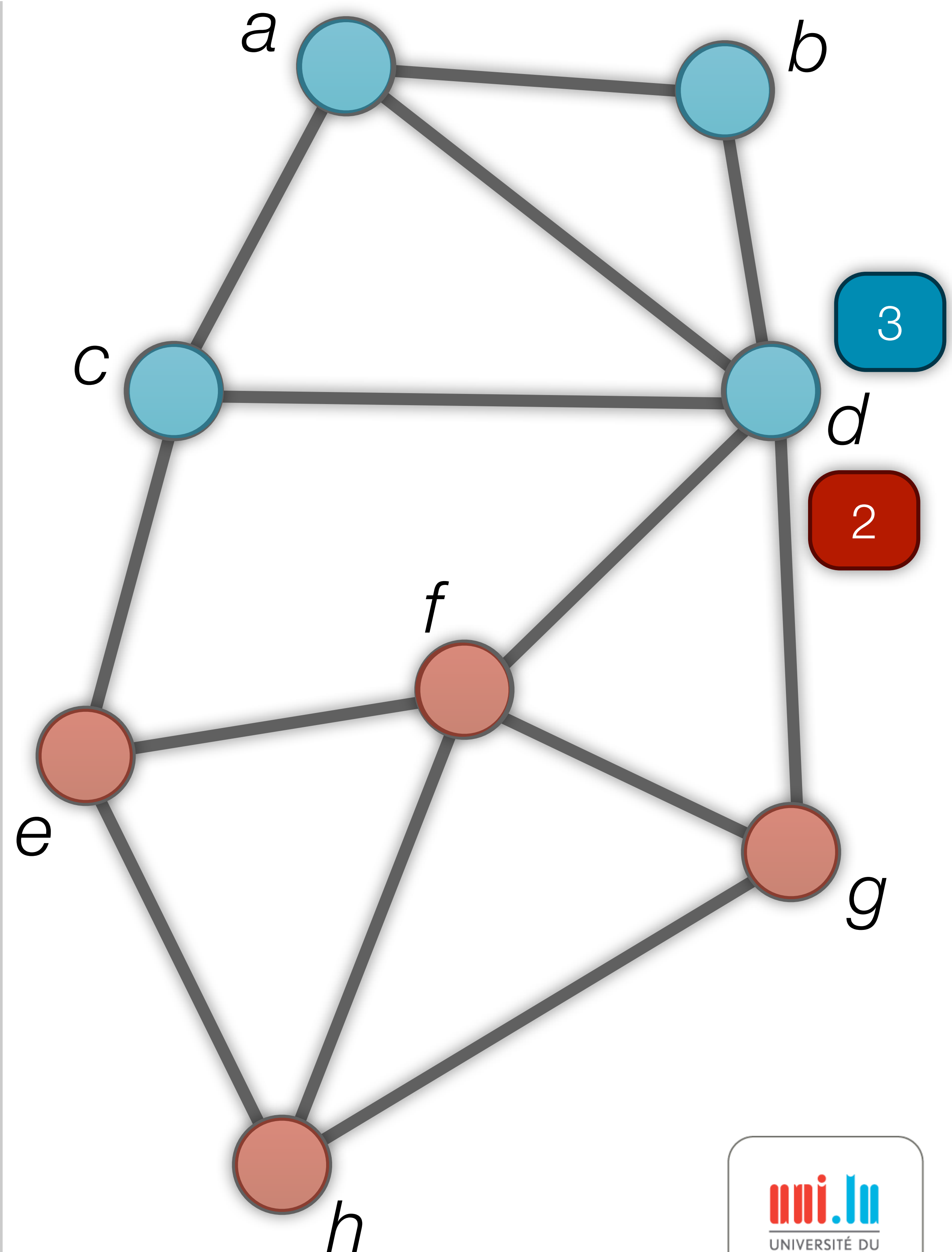


UNIVERSITÉ DU LUXEMBOURG

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted
  - Node changes to the label with highest count
- Operation is (a)synchronous

- Known issues:
  - Label oscillation (synchronous)
  - Community domination



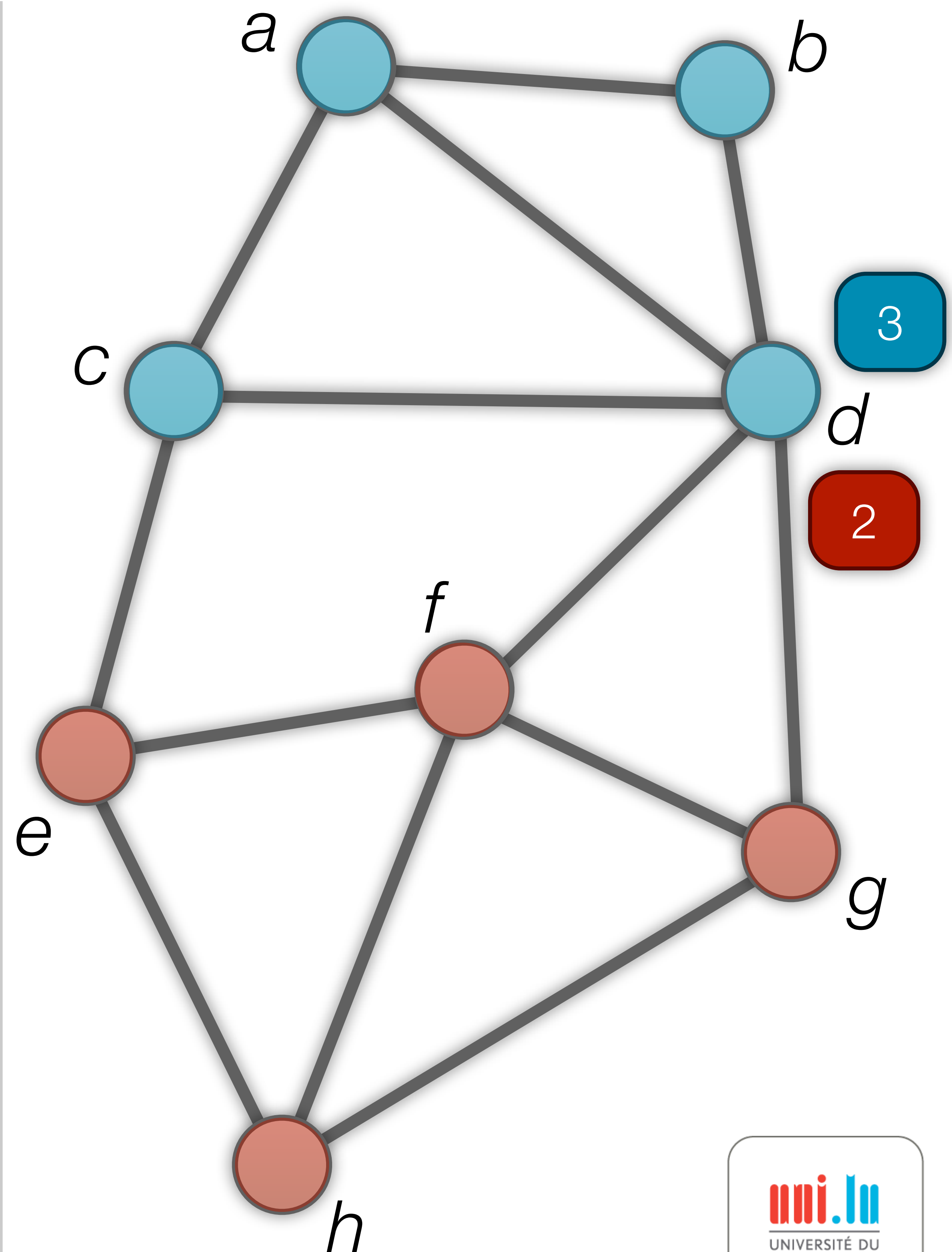
[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted
  - Node changes to the label with highest count
- Operation is (a)synchronous

- Known issues:
  - Label oscillation (synchronous)
  - Community domination
  - Parameter dependent



[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

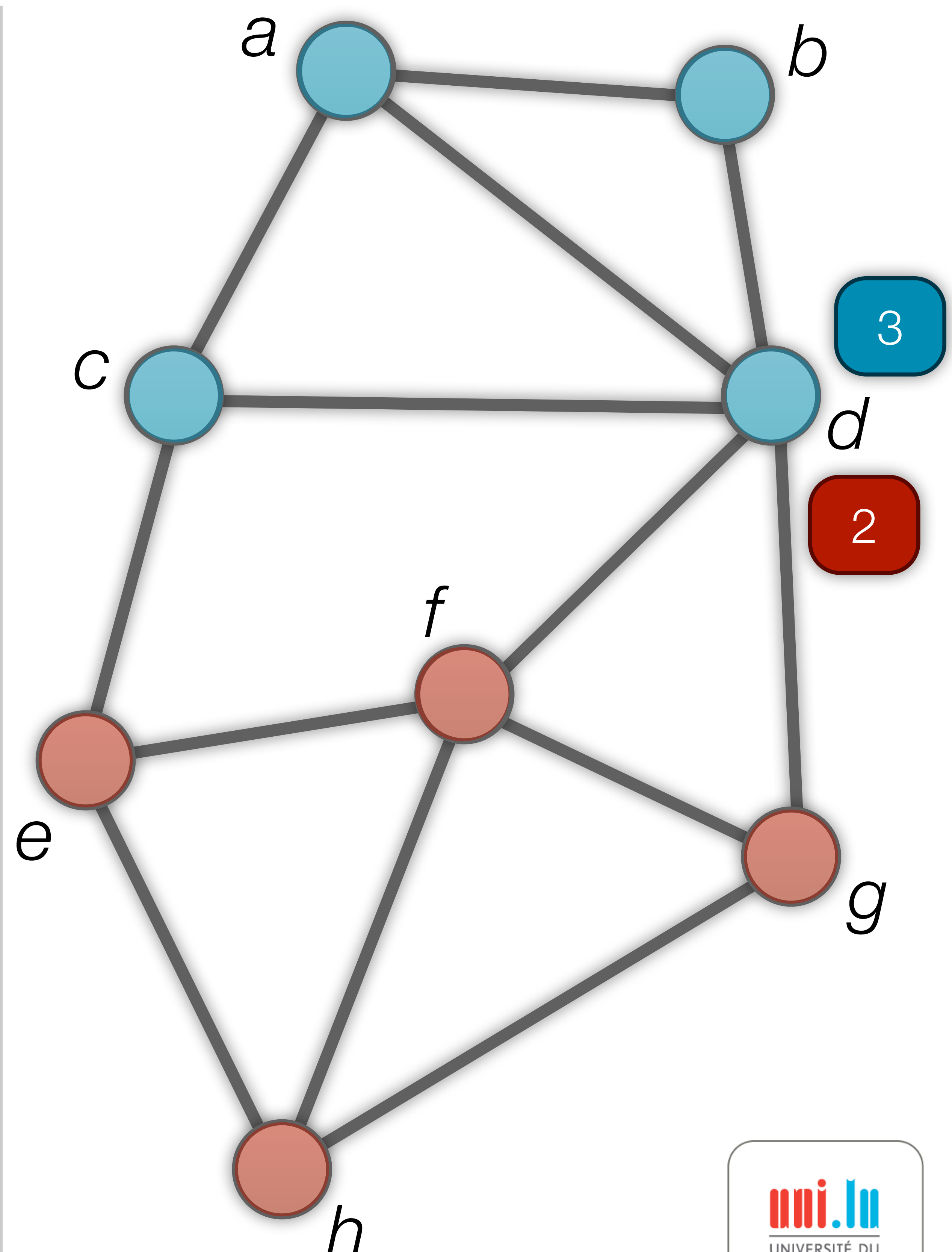
[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

Principle and limitation of distributed algorithms

- Alternative algorithms[1][2] use epidemic label propagation
  - Nodes broadcast their labels to their neighbors
  - Received labels are counted
  - Node changes to the label with highest count
- Operation is (a)synchronous

- Known issues:
  - Label oscillation (synchronous)
  - Community domination
  - Parameter dependent

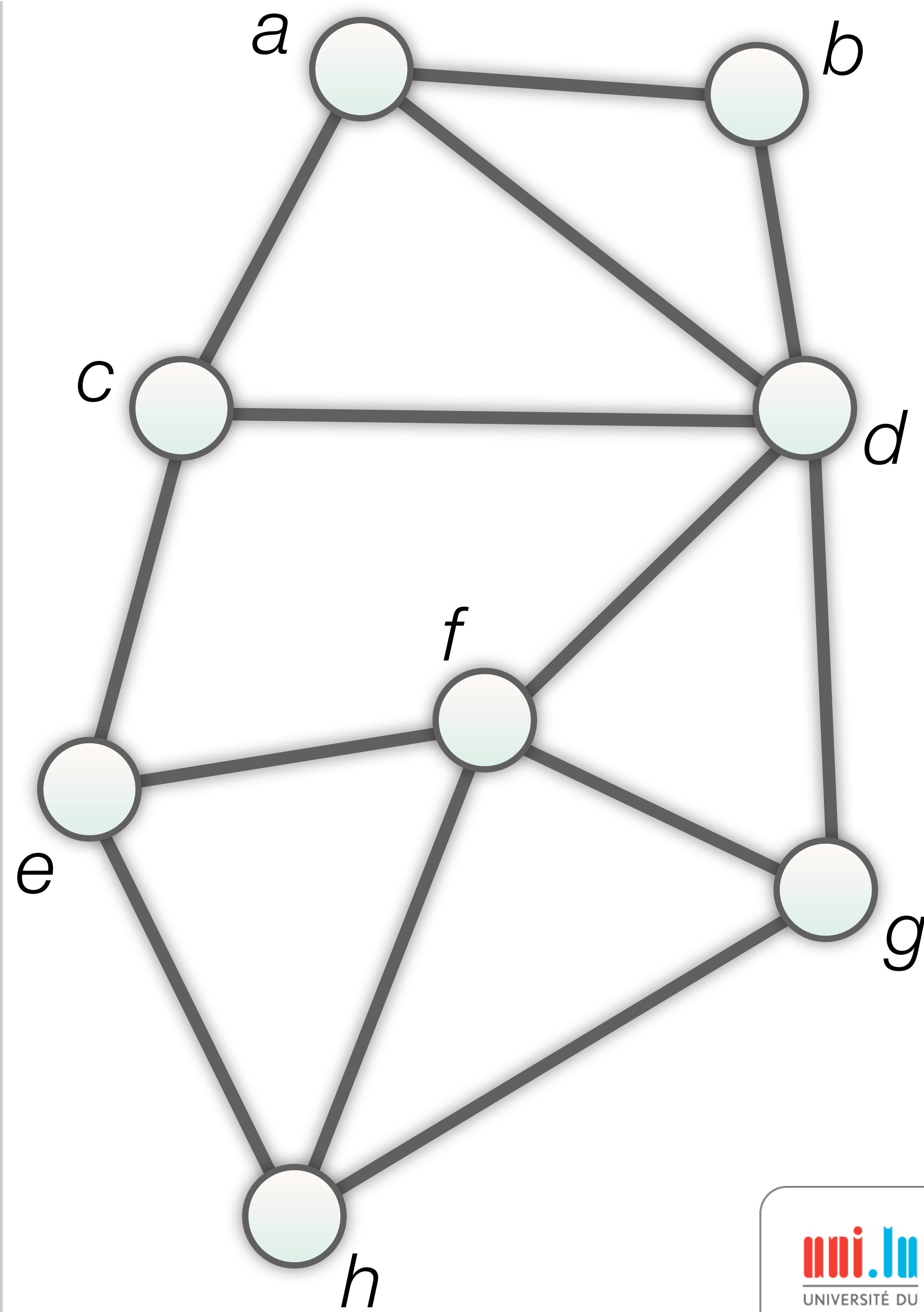
- Not really focusing on dynamic networks



[1] U.N. Raghavan et al., *Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks*, 2007

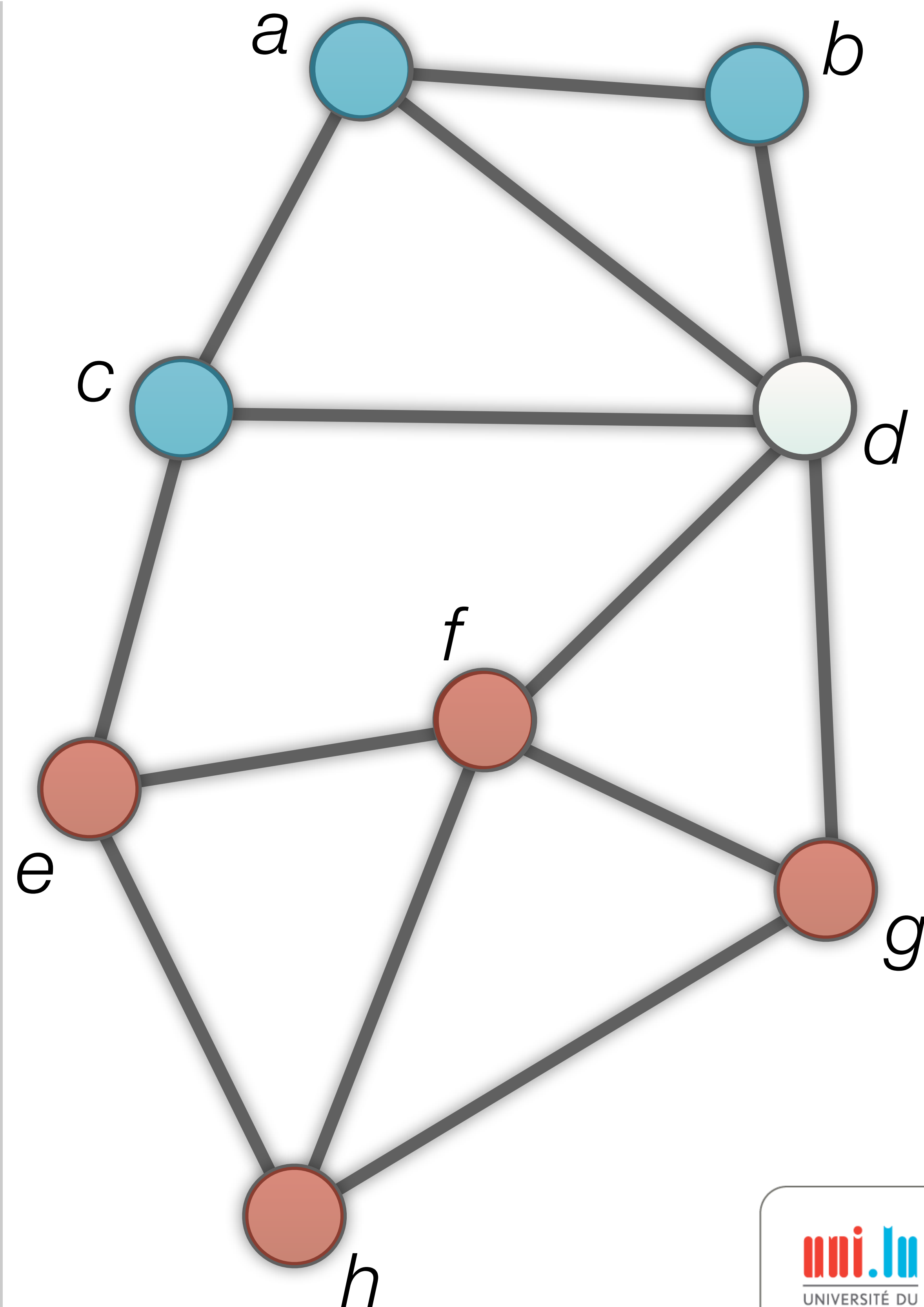
[2] I.X.Y. Leung and P. Hui and P. Lio and J. Crowcroft, *Towards Real-Time Community Detection in Large Networks)*, 2009

SHARC: Sharper Heuristic for Assignment of Robust Communities



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

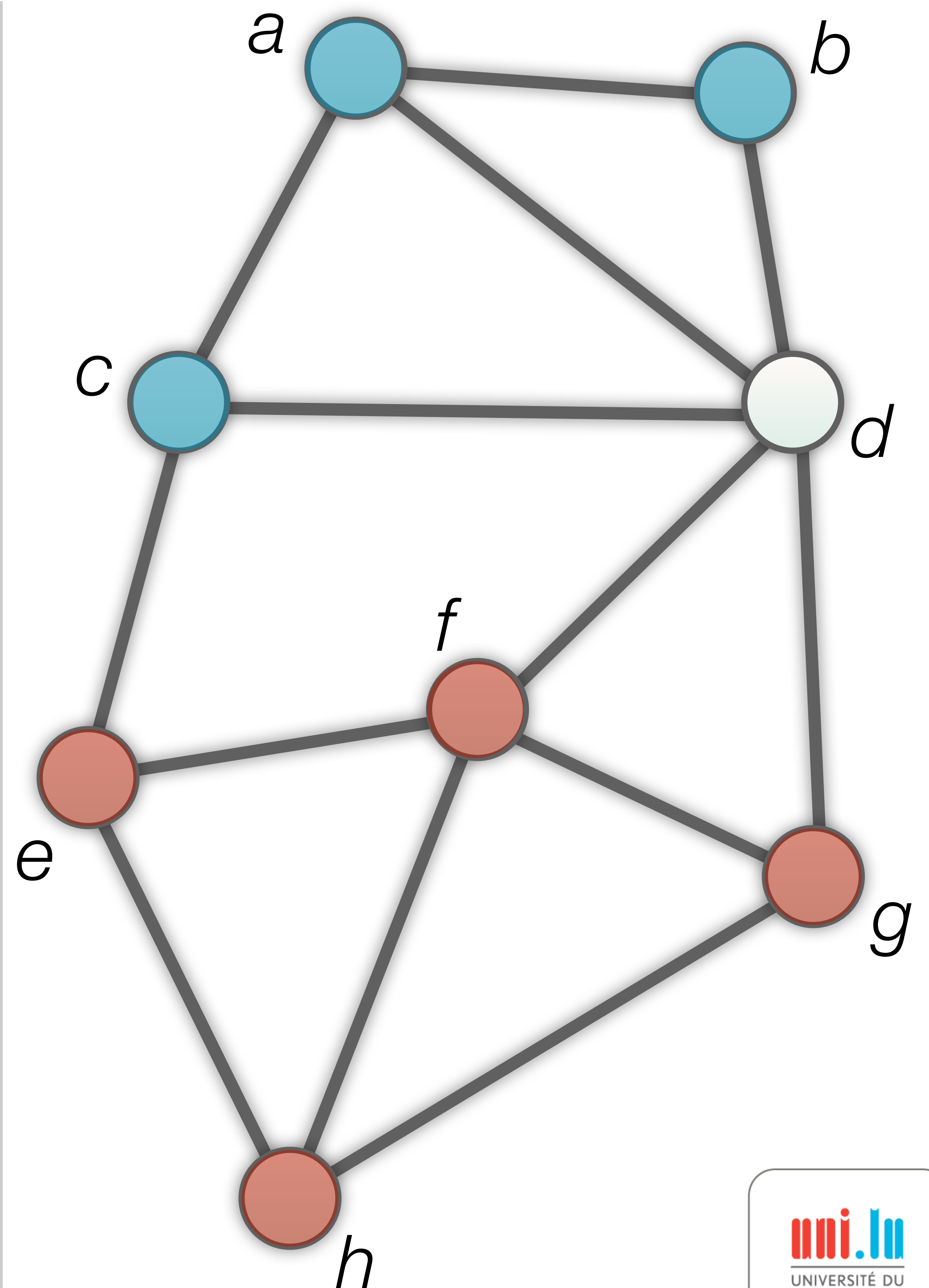
SHARC: Sharper Heuristic for Assignment of Robust Communities



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information

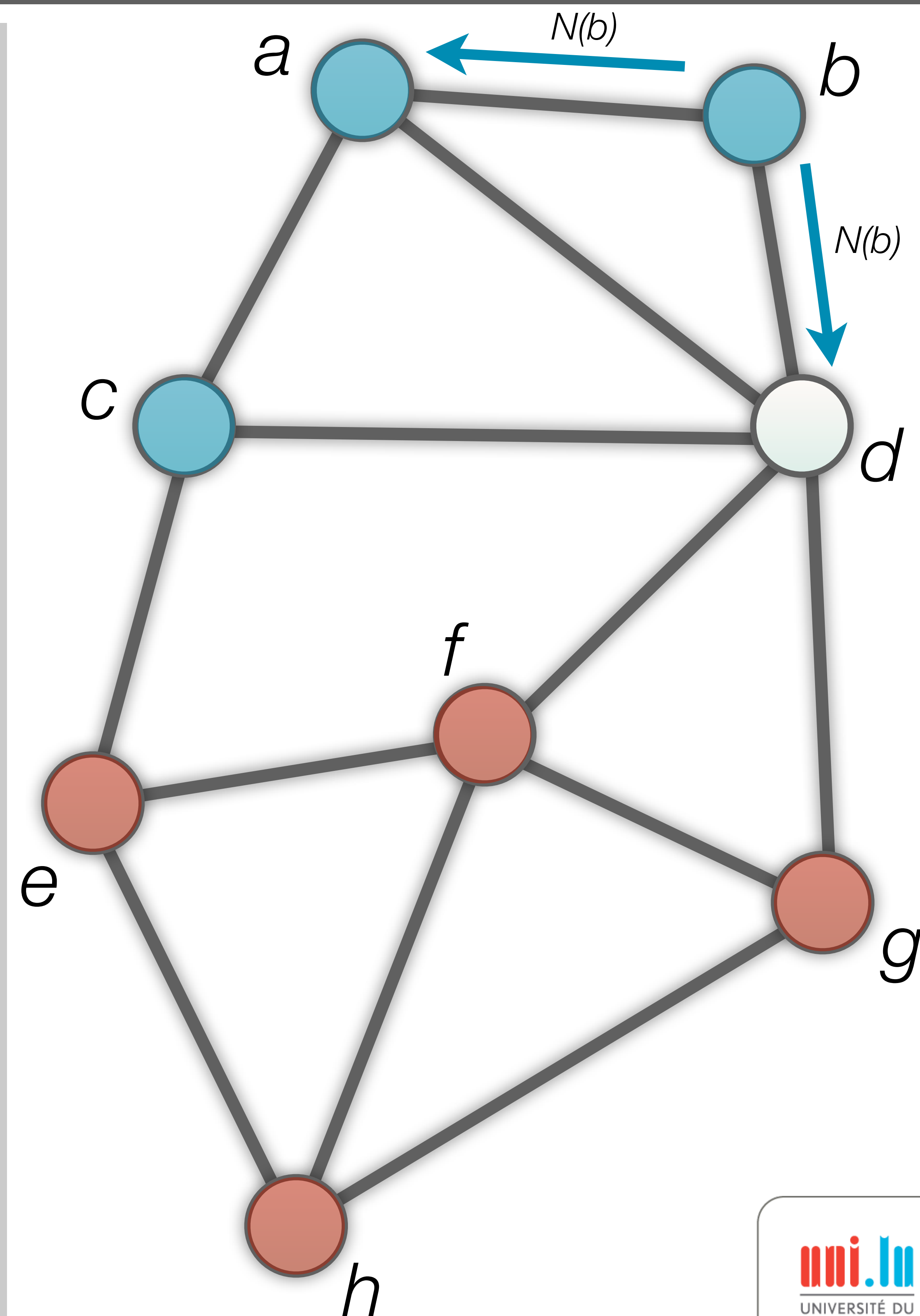


[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

UNIVERSITÉ DU LUXEMBOURG

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors



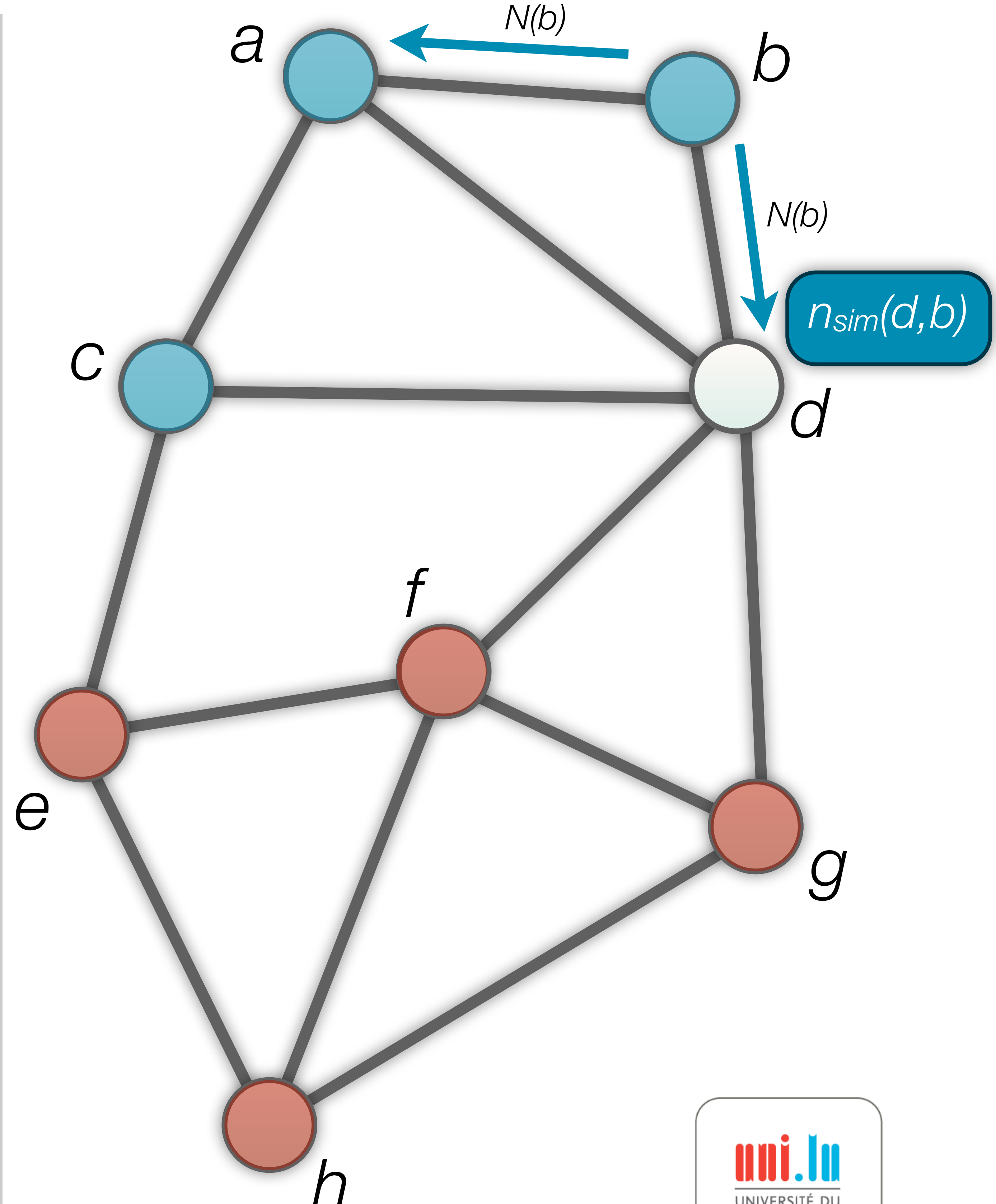
[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$



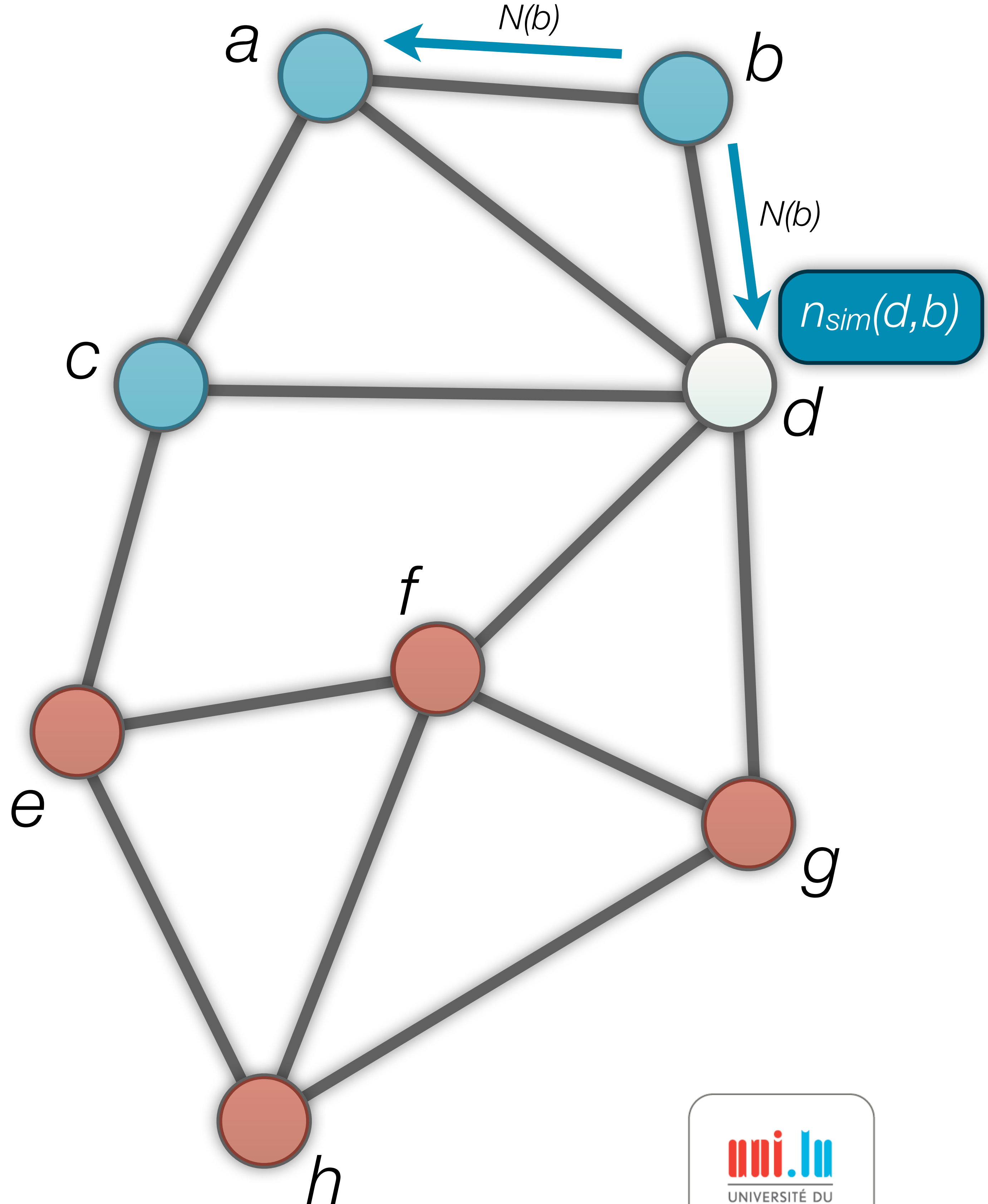
[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$
$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

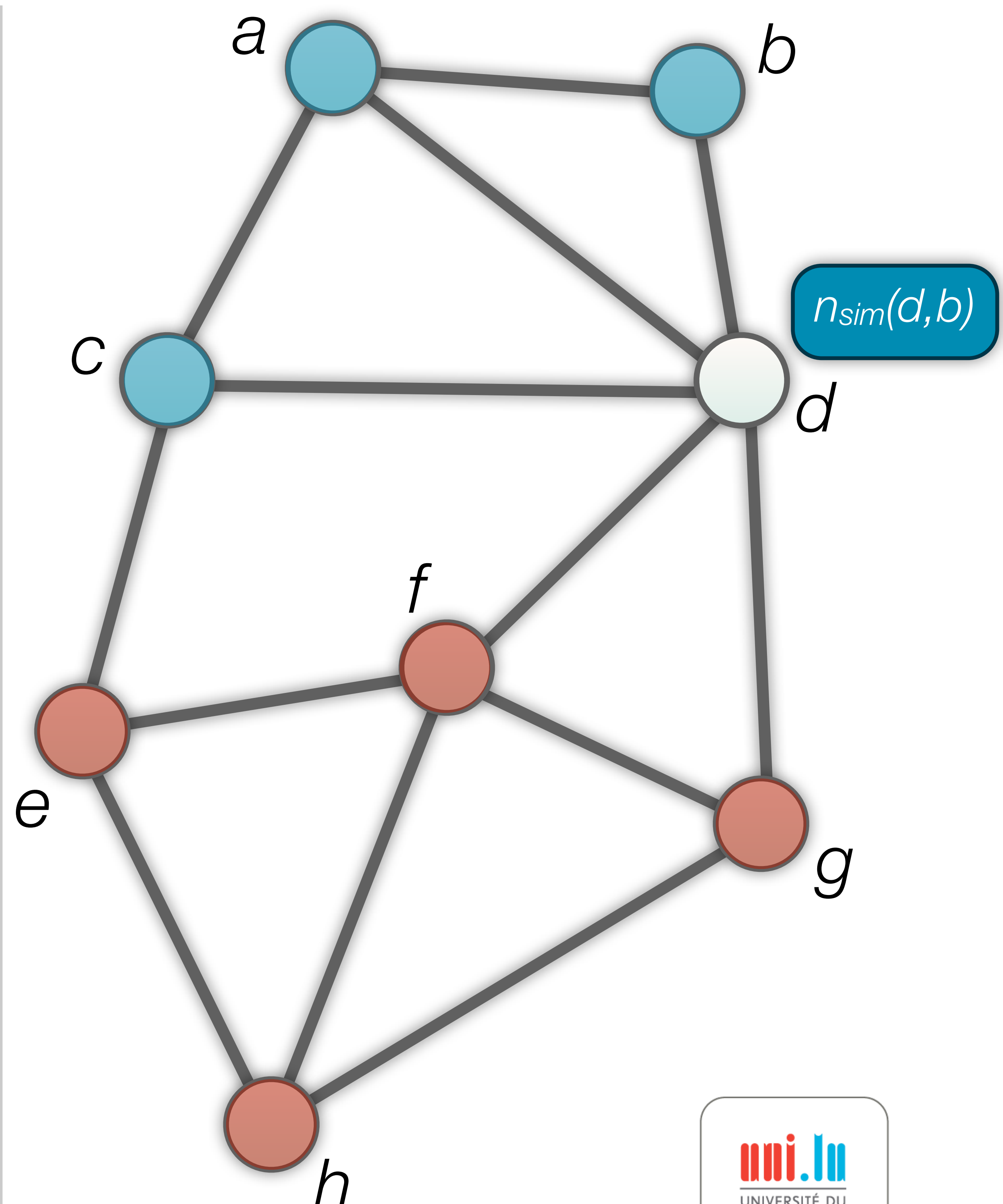
SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



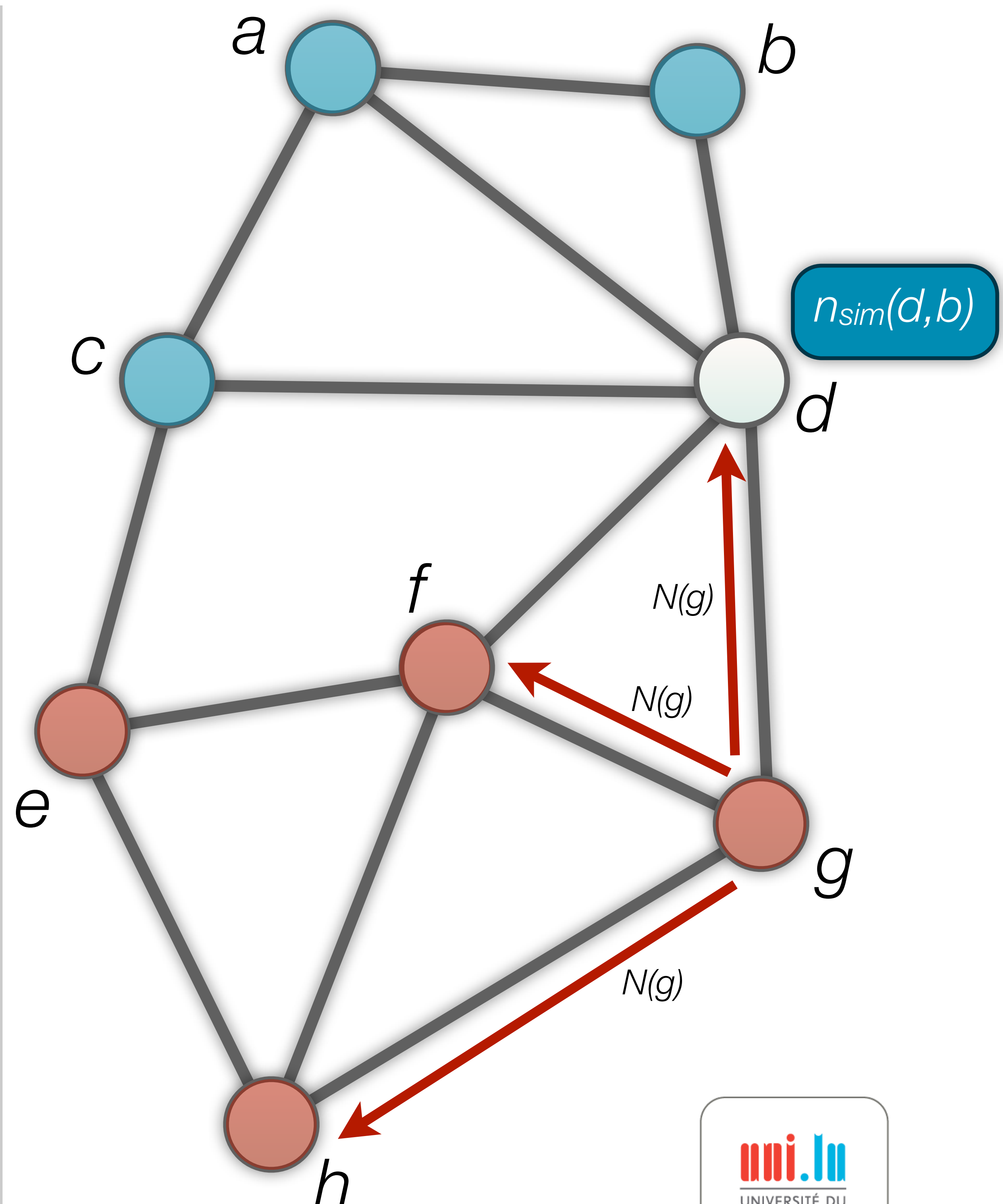
[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$
$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

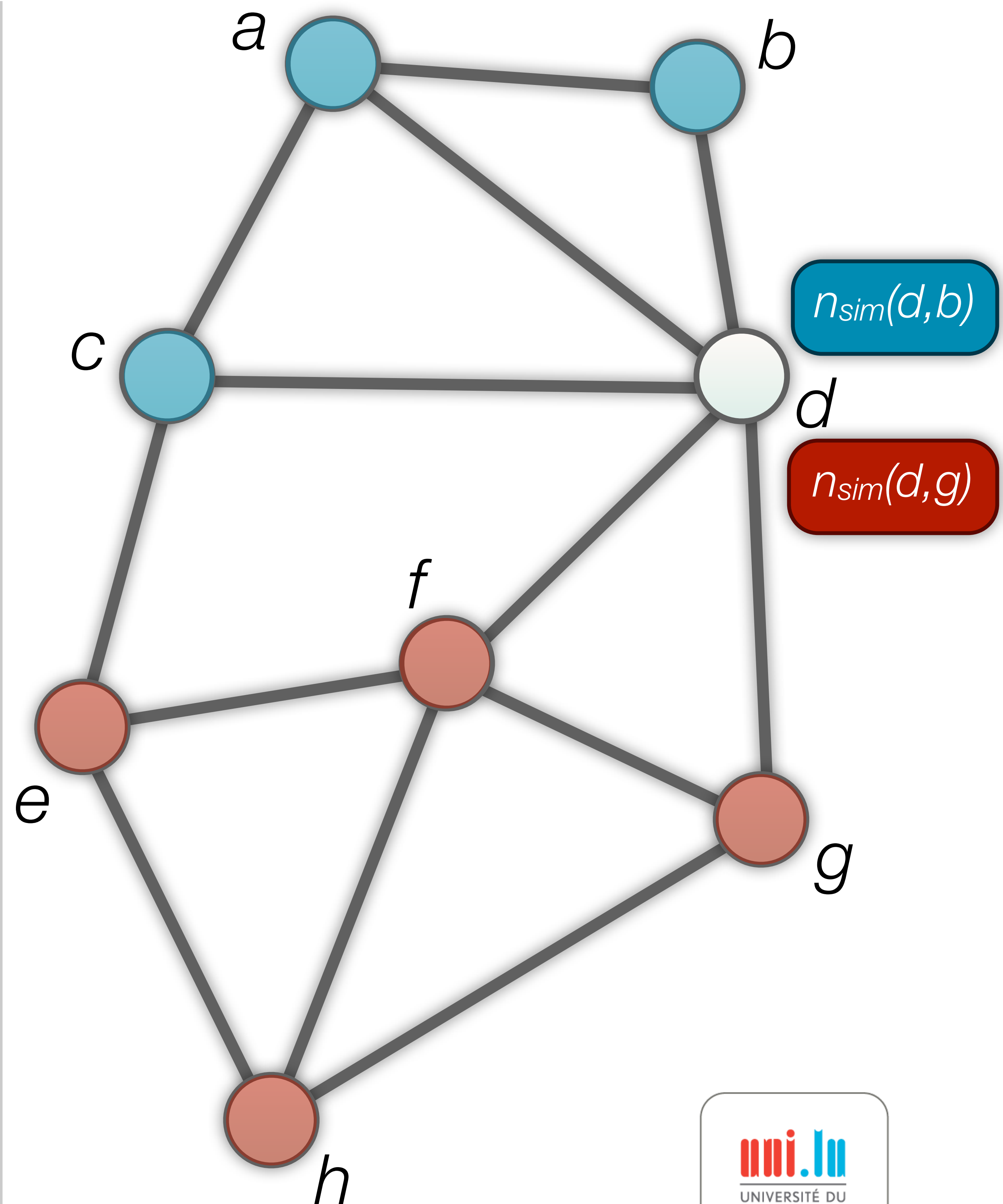
SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

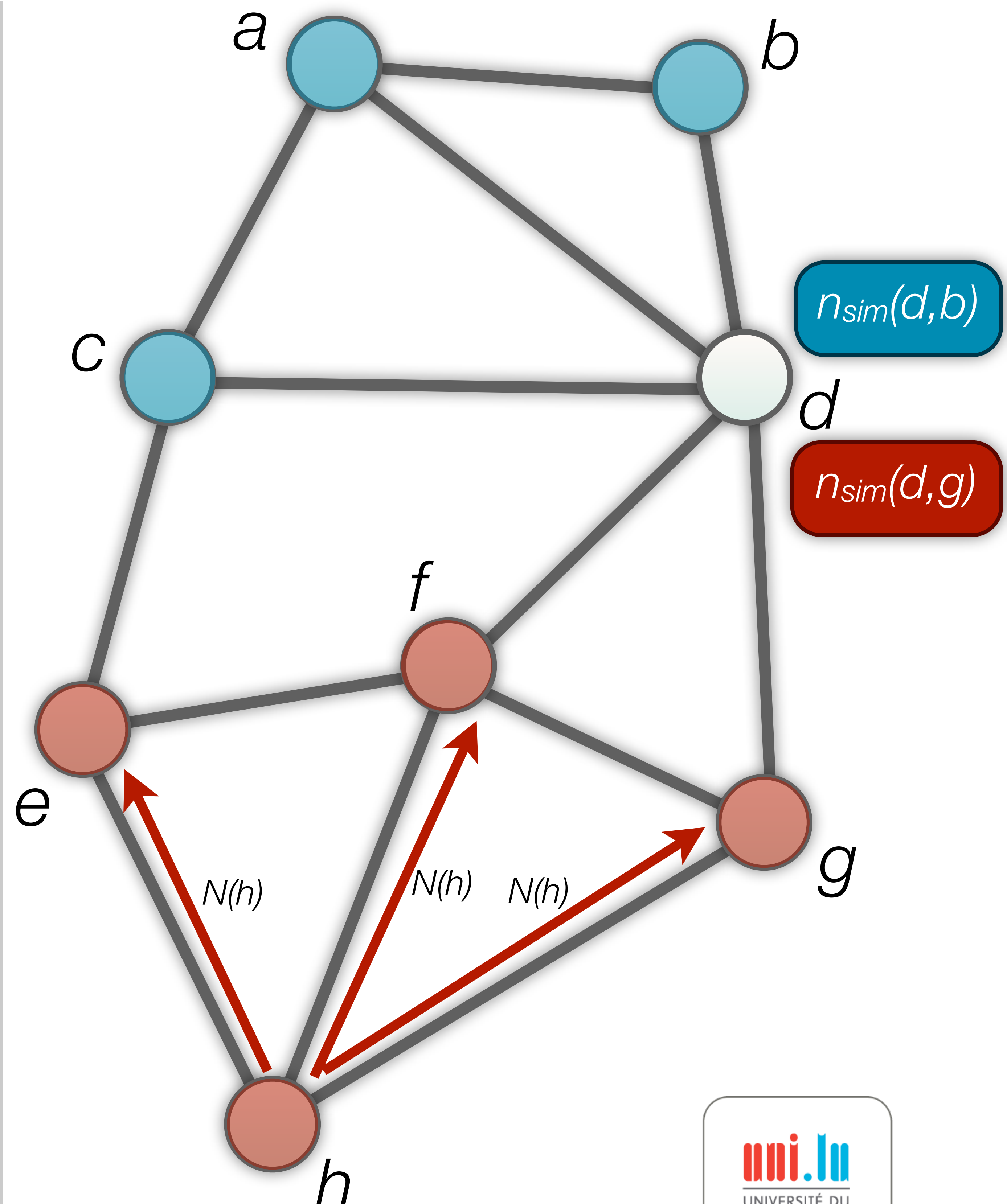
SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

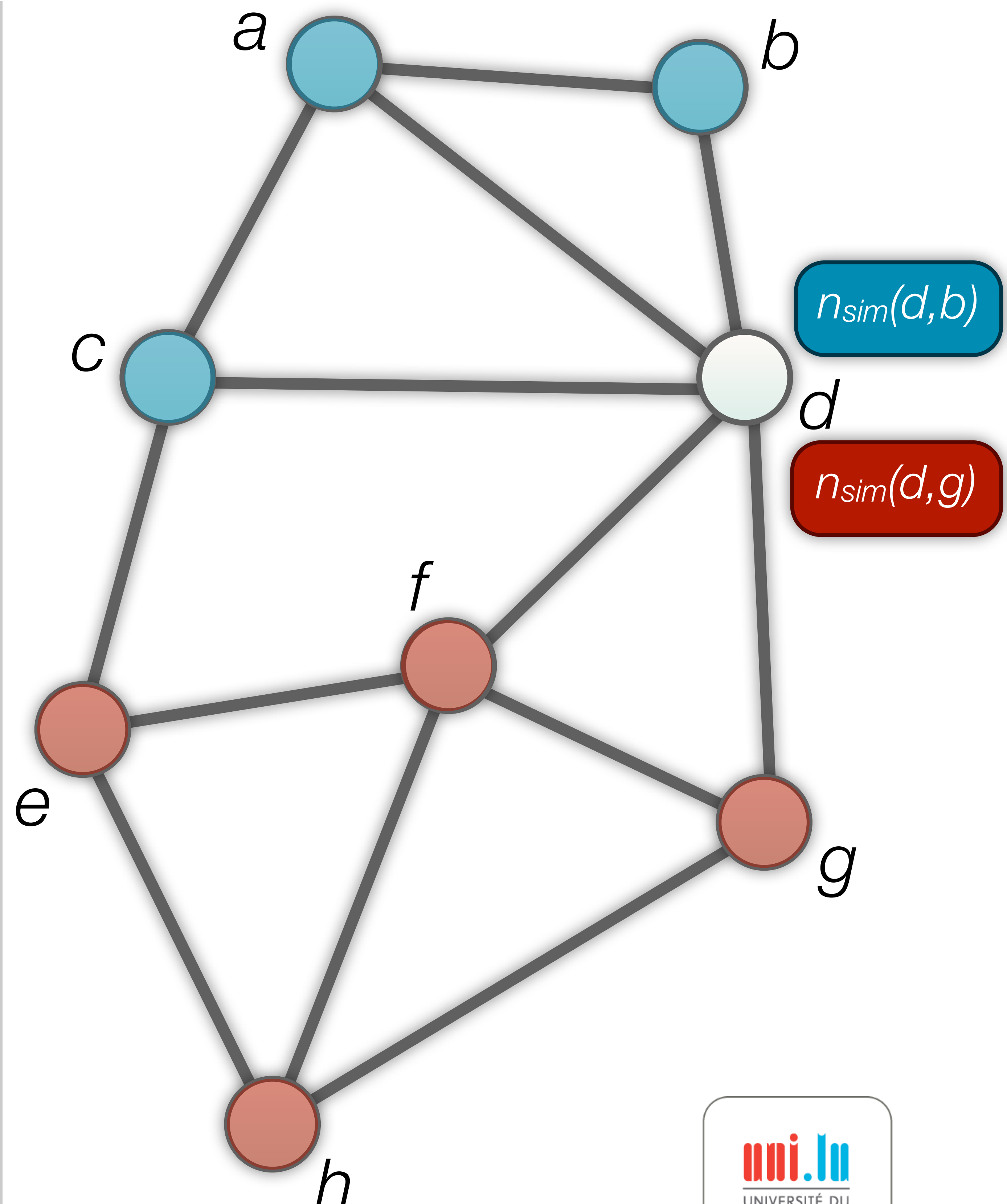
SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



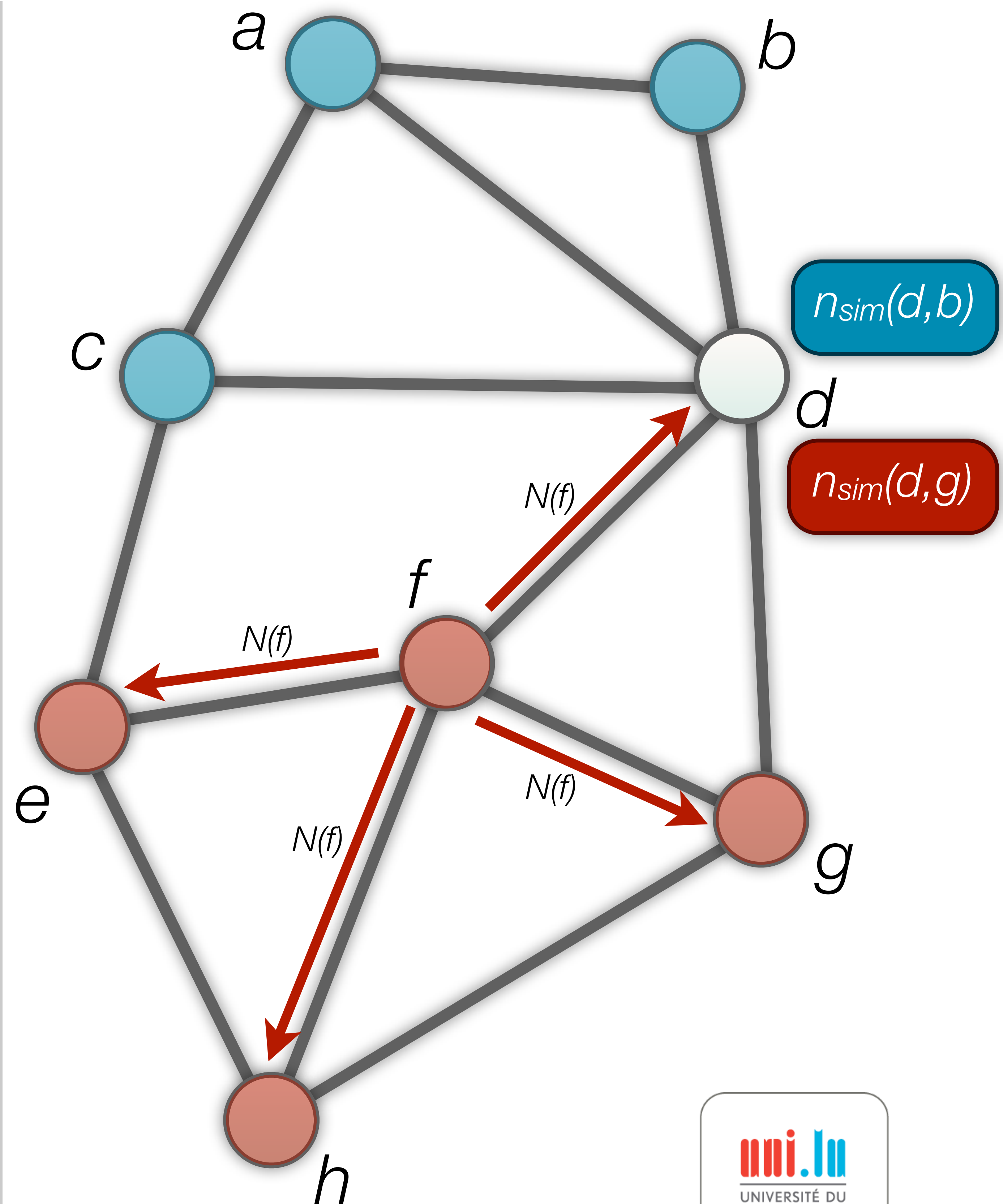
[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$
$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

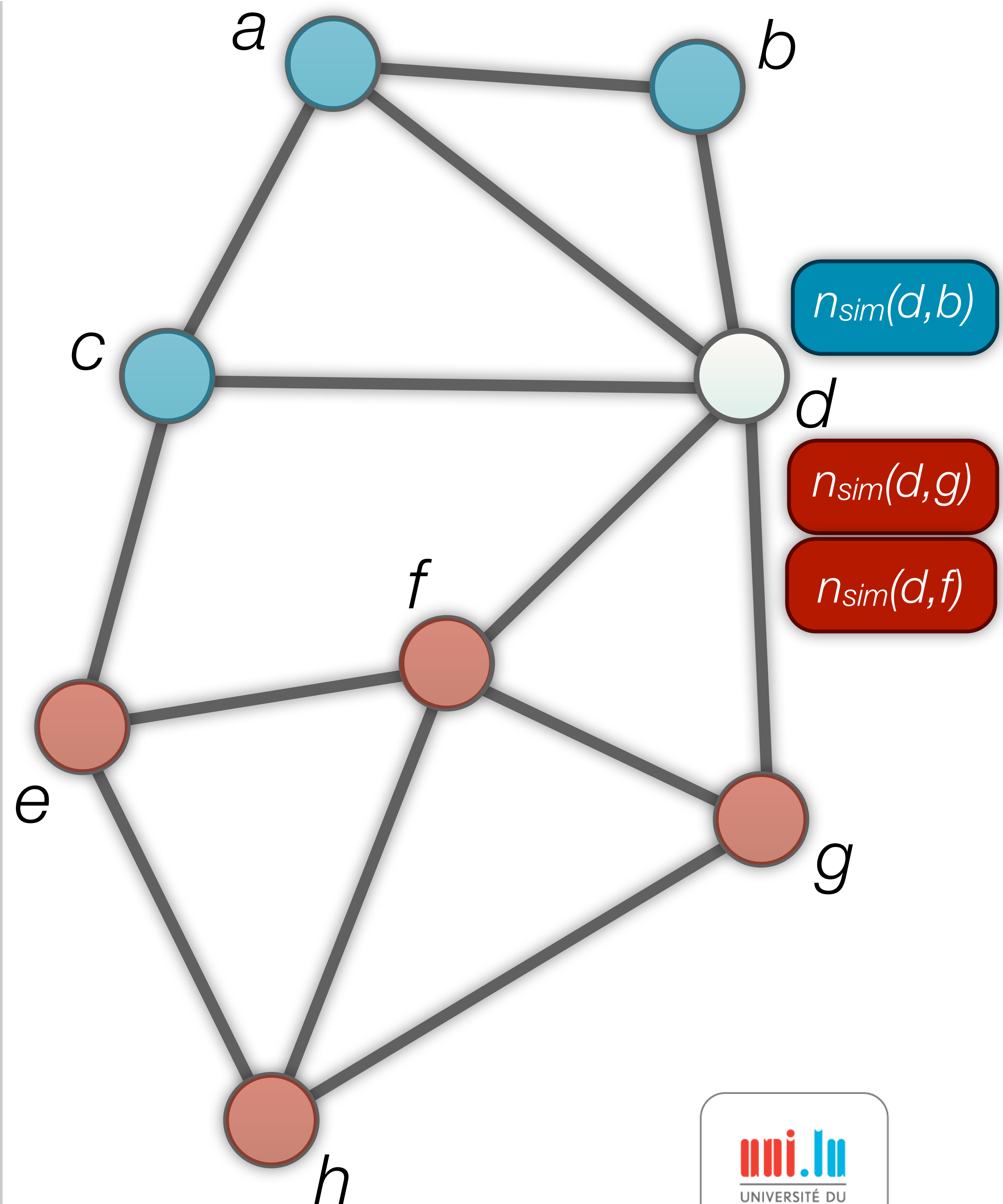
SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



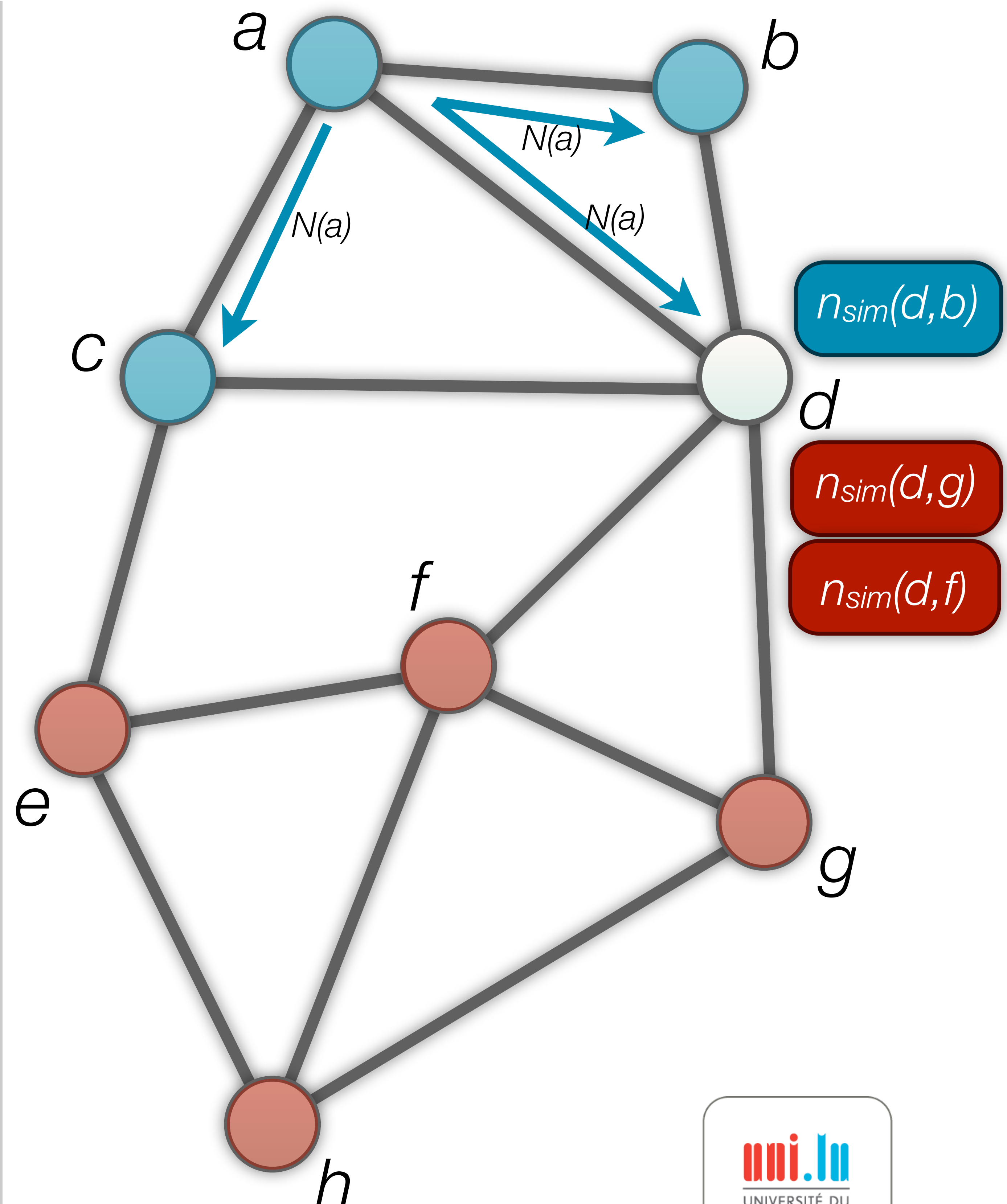
[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$
$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

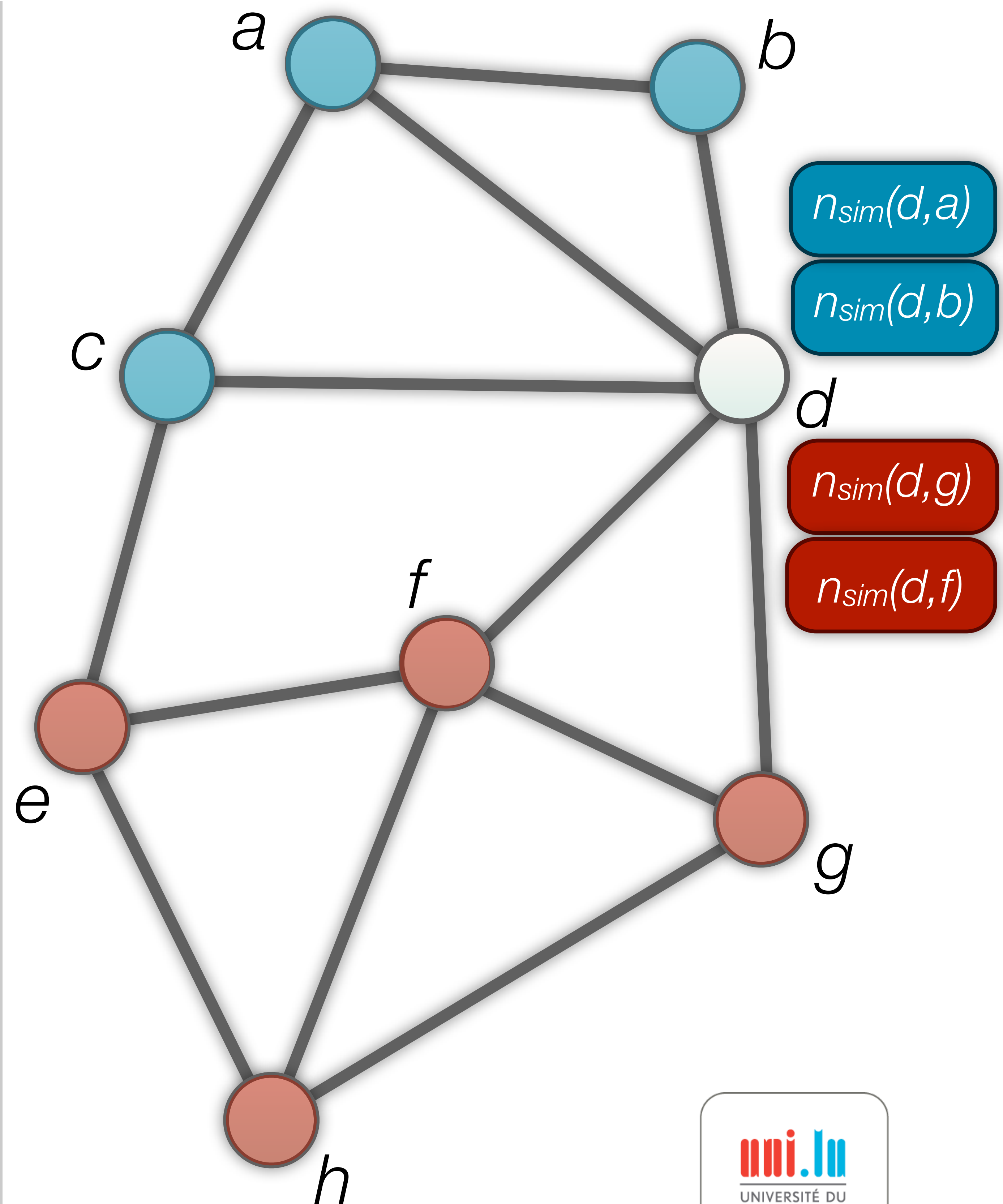
SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



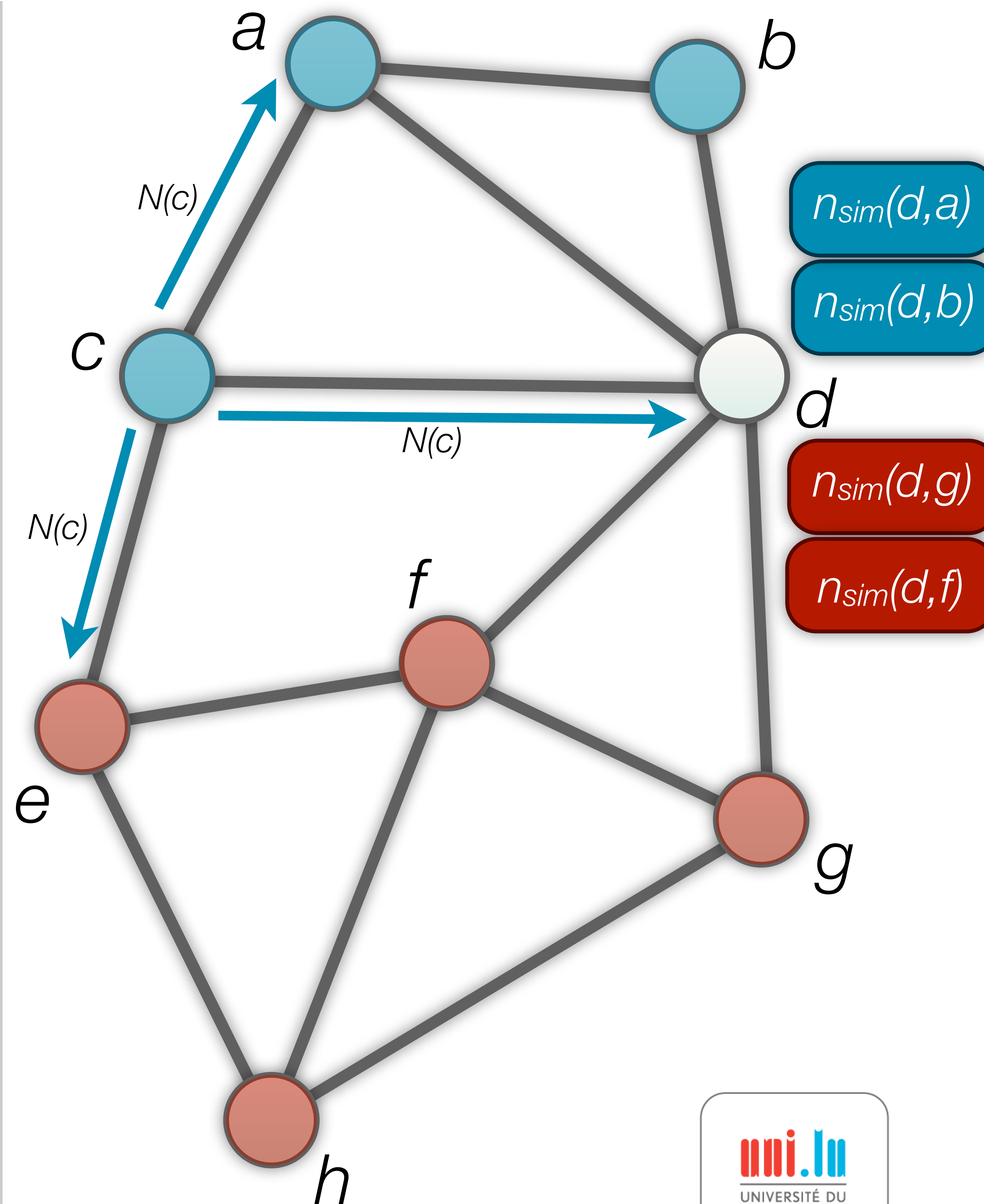
[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$
$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



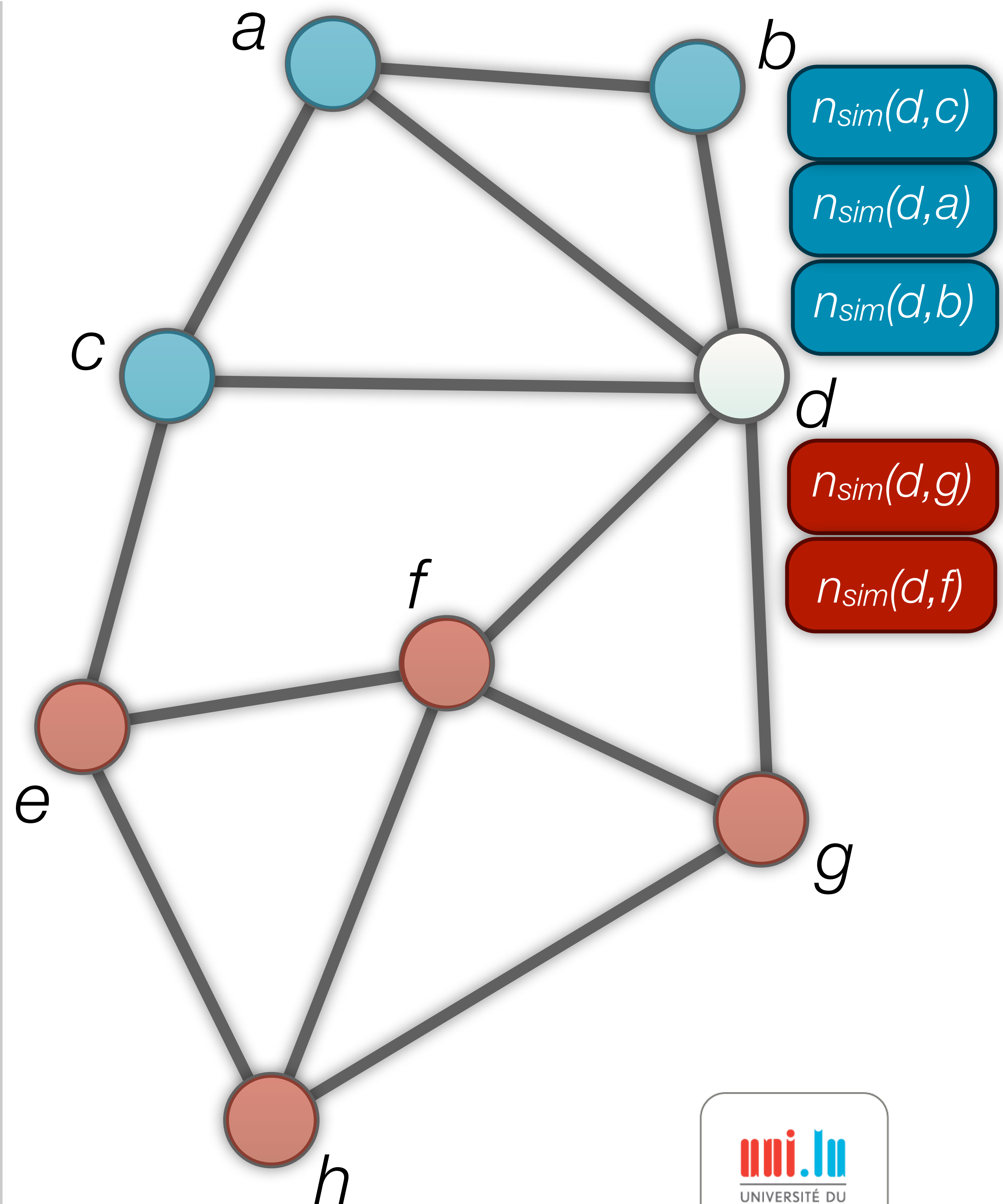
[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$
$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

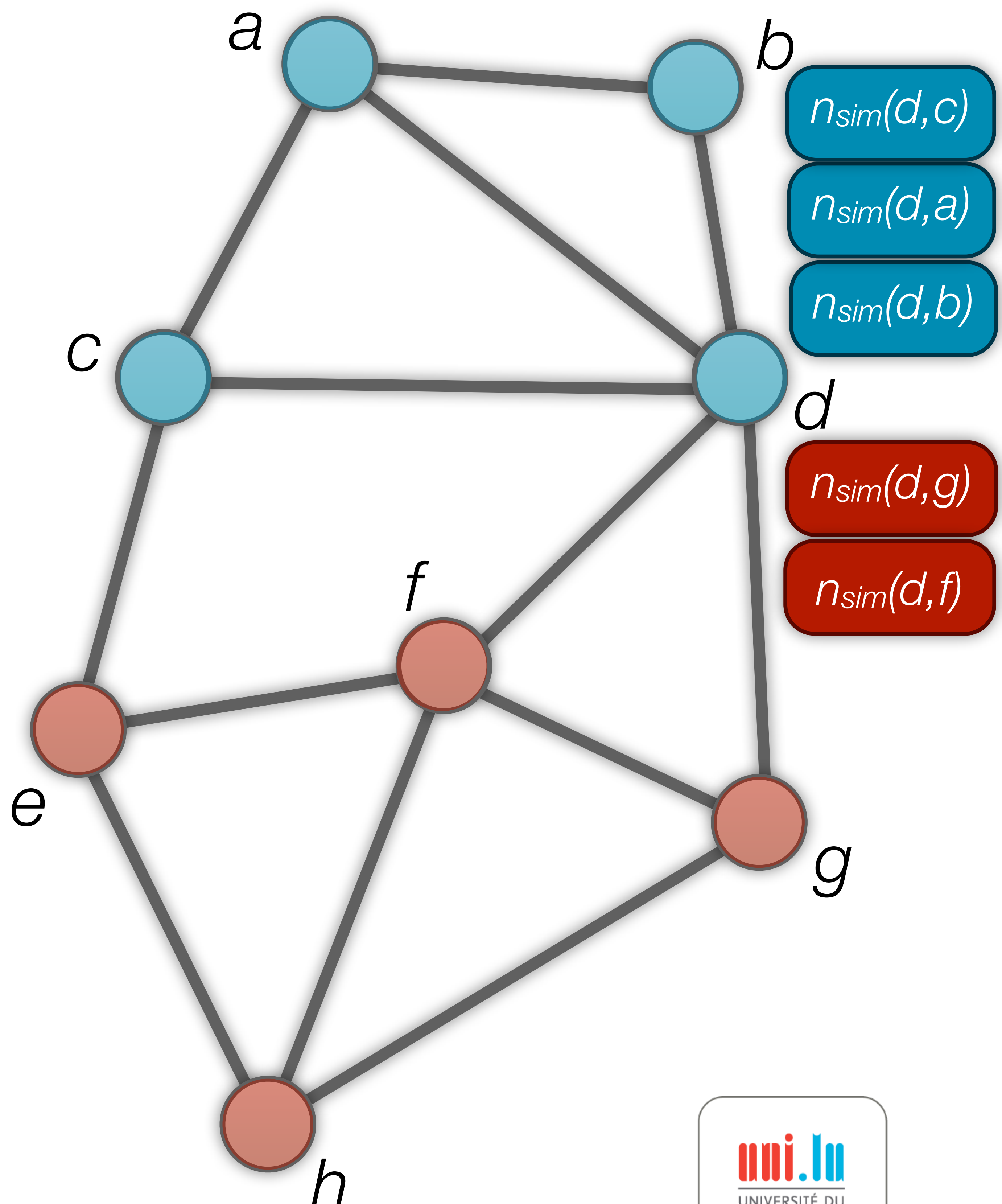
- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks

$$\boxed{C(a) = \arg\max_{c \in \mathcal{C}} \sum_{\substack{b \in N(a)/\\ C(b)=c}} n_{sim}(a,b)}$$



[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

SHARC: Sharper Heuristic for Assignment of Robust Communities

- SHARC uses two-hop information
  - Nodes broadcast their labels **and** their list of neighbors
  - Used in the computation of a **neighborhood similarity** measure

$$\forall a \in V(\mathcal{N}), b \in N(a)$$

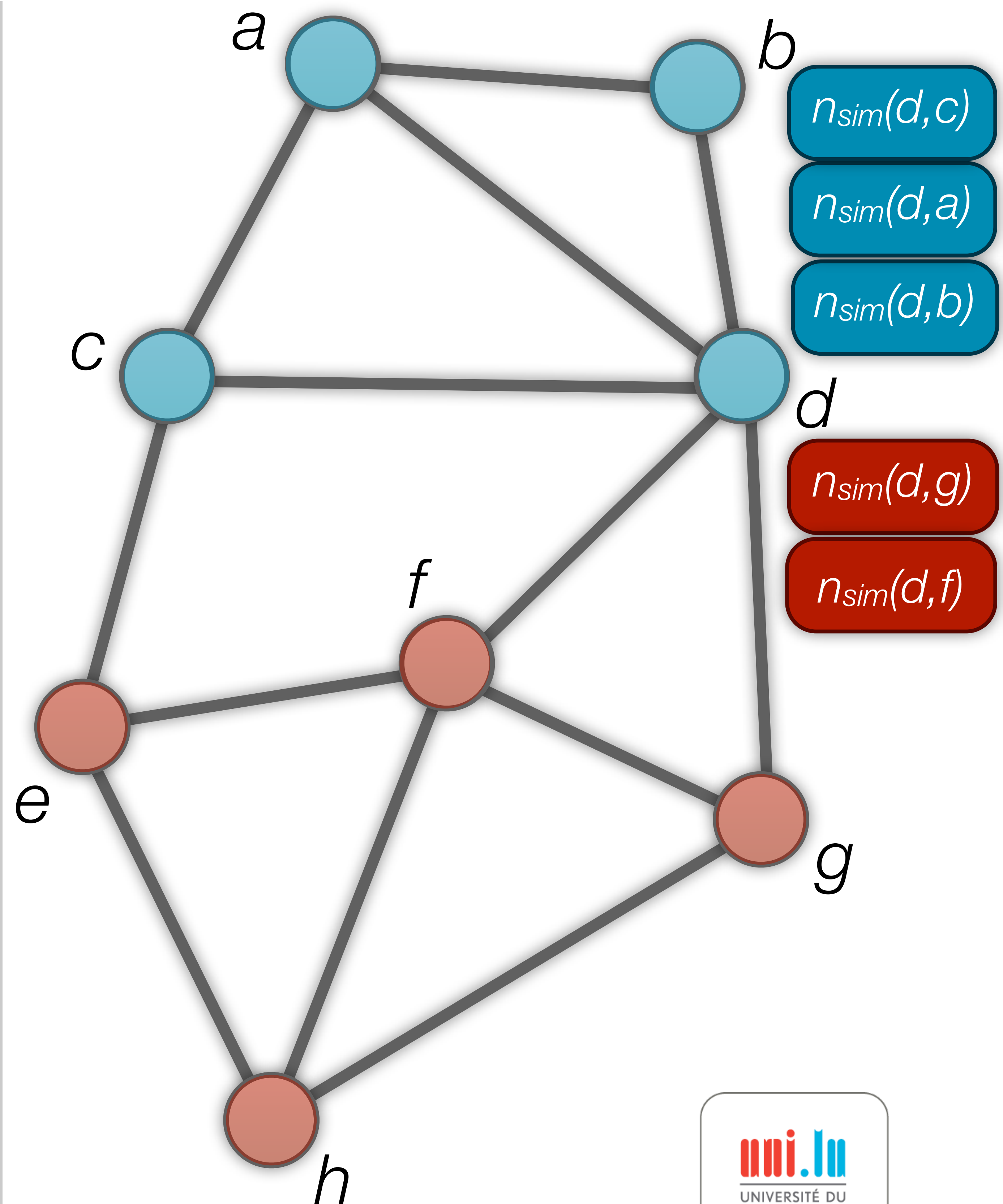
$$n_{sim}(a,b) = 1 - \frac{|(N(a) \setminus N(b)) \cup (N(b) \setminus N(a))|}{|N(a)| + |N(b)|}$$

  - inspired by the **triadic closure**[1] property of social networks

$$\boxed{C(a) = \arg\max_{c \in \mathcal{C}} \sum_{\substack{b \in N(a)/ \\ C(b)=c}} n_{sim}(a,b)}$$

- Objectives:
  - Strengthen the community boundaries
  - Be more deterministic

[1] Marc S. Granovetter. *The Strength of weak ties.* American Journal of Sociology, Vol. 78 Num. 6 p.1360

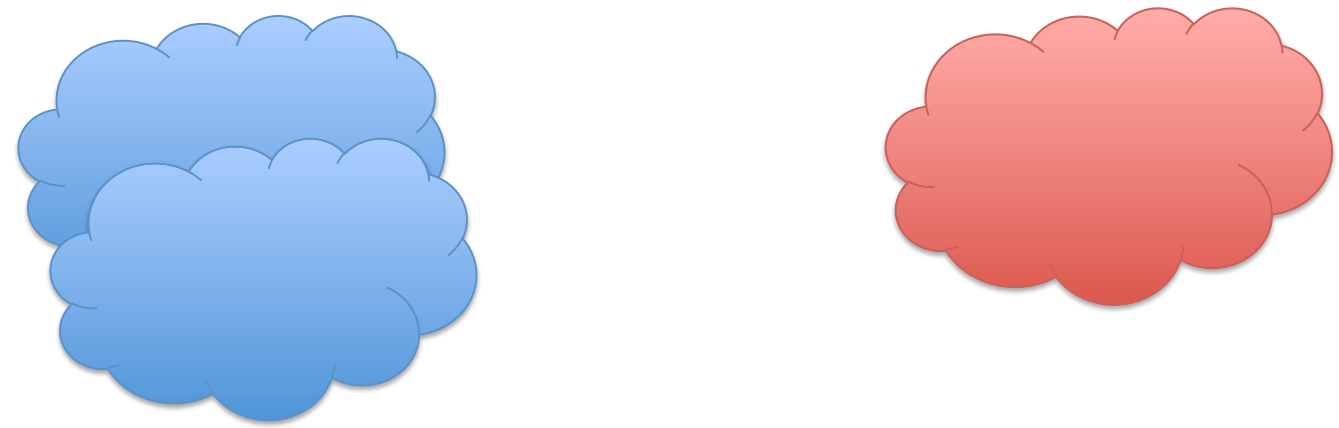


Additional SHARC improvements

Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect

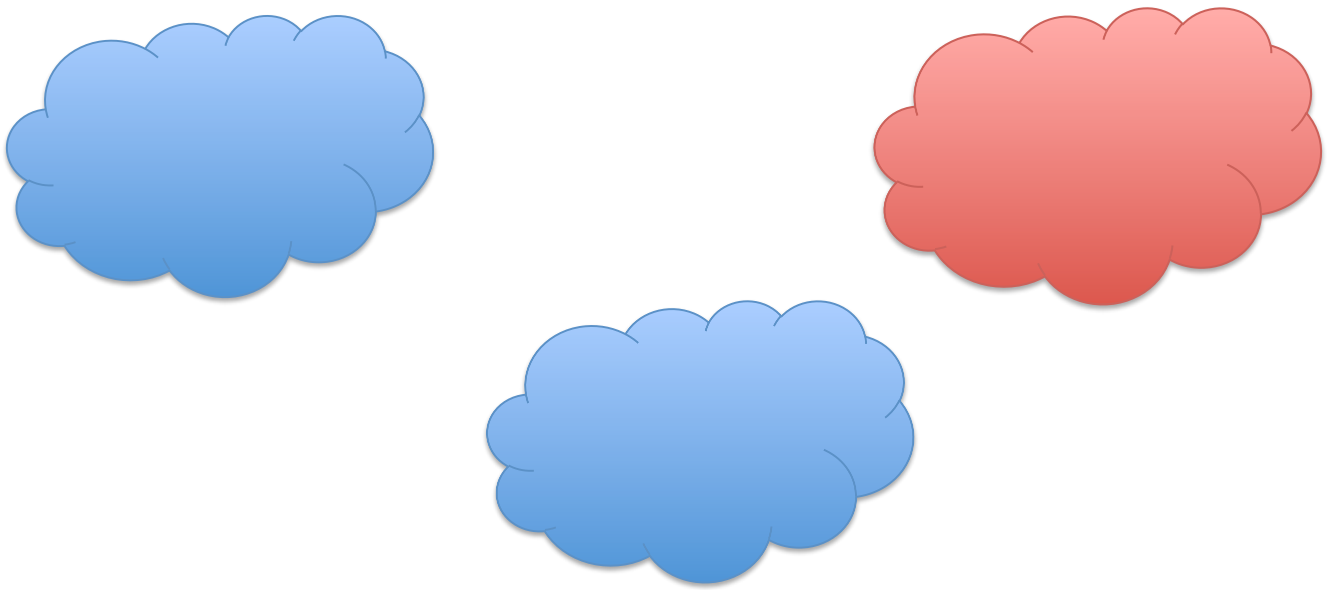
Wandering community effect



Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect

Wandering community effect



Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect

Wandering community effect

Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect

Wandering community effect

Additional SHARC improvements

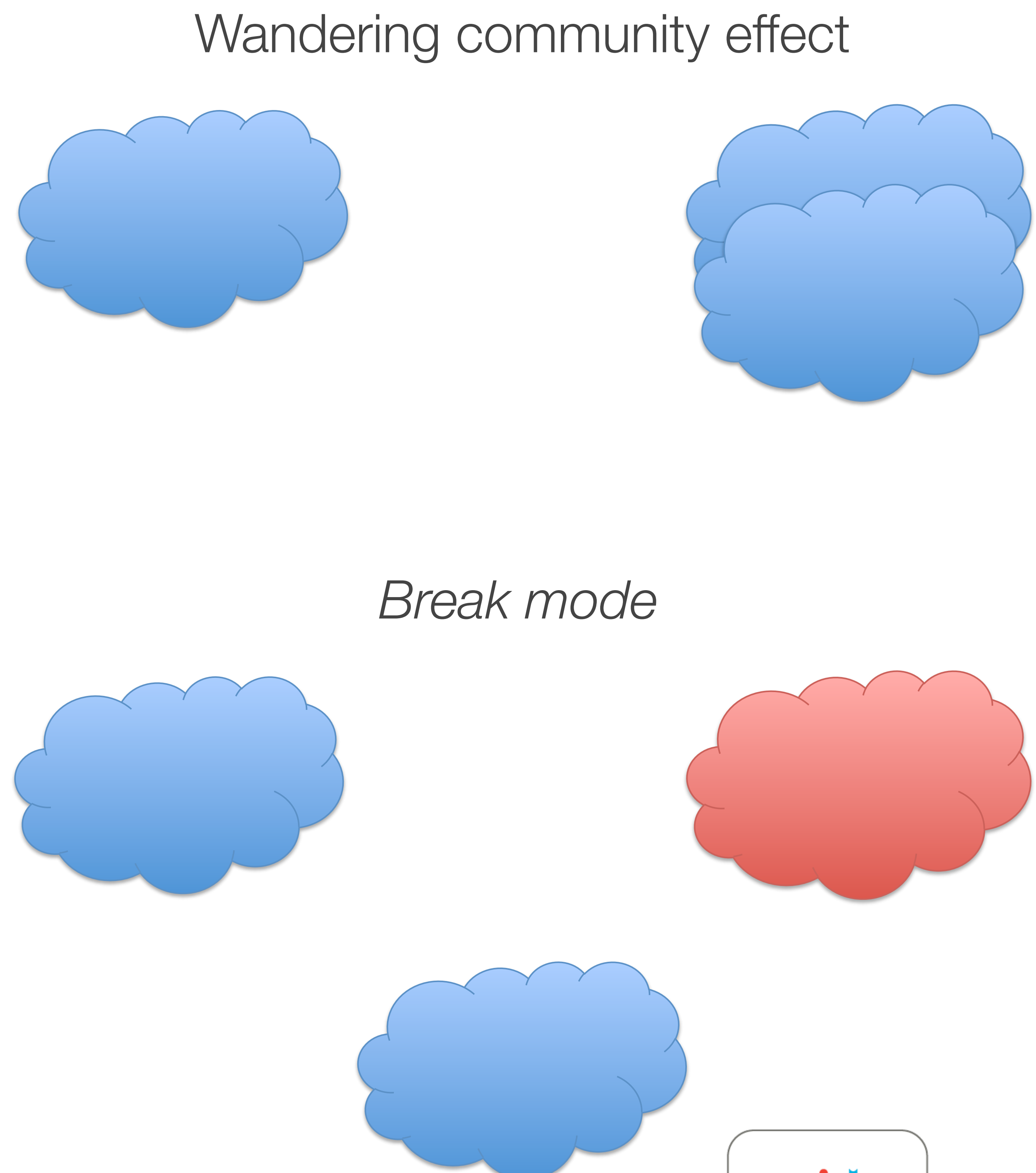
- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect

Wandering community effect

Break mode

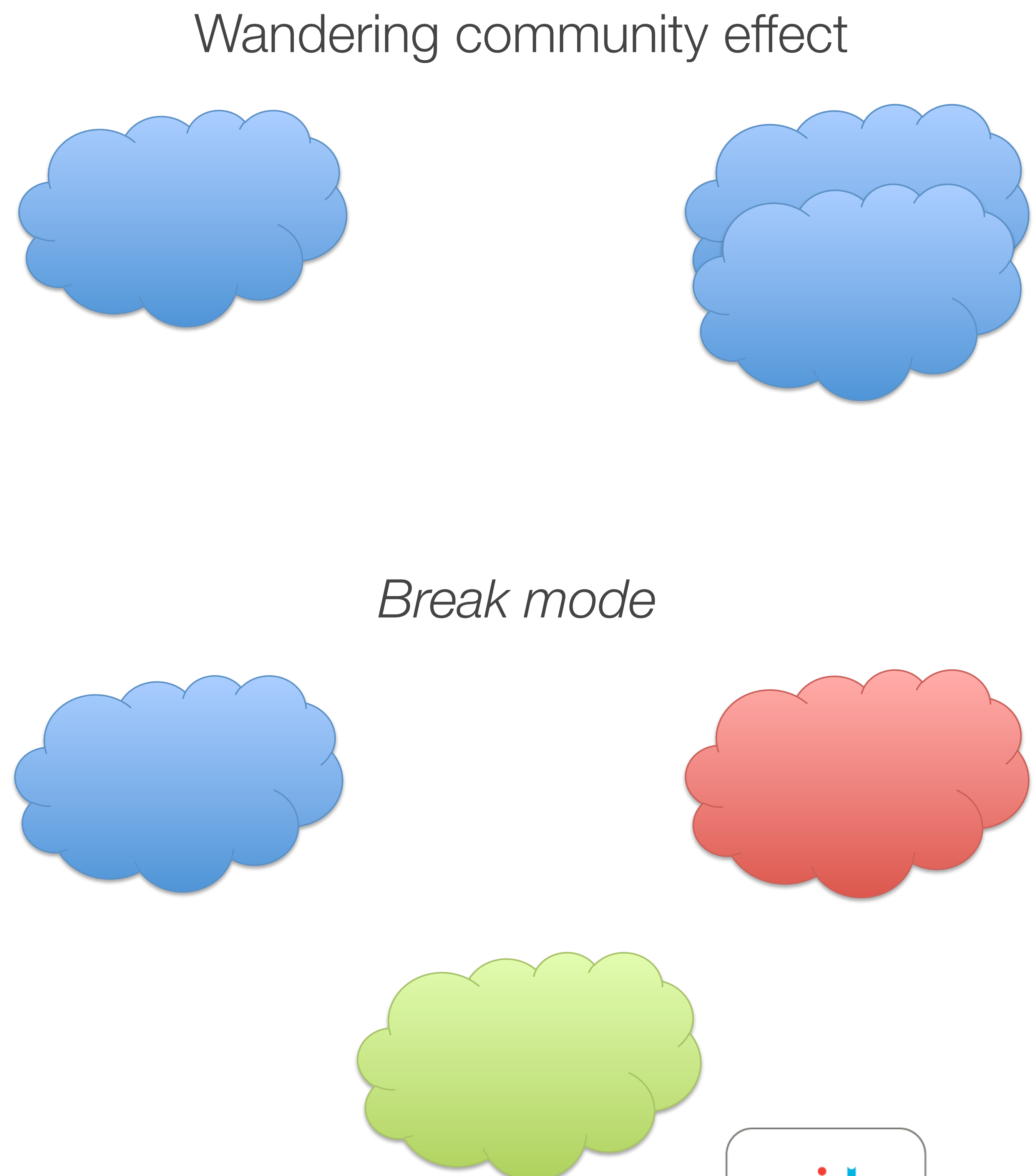
Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect



Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect



Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect

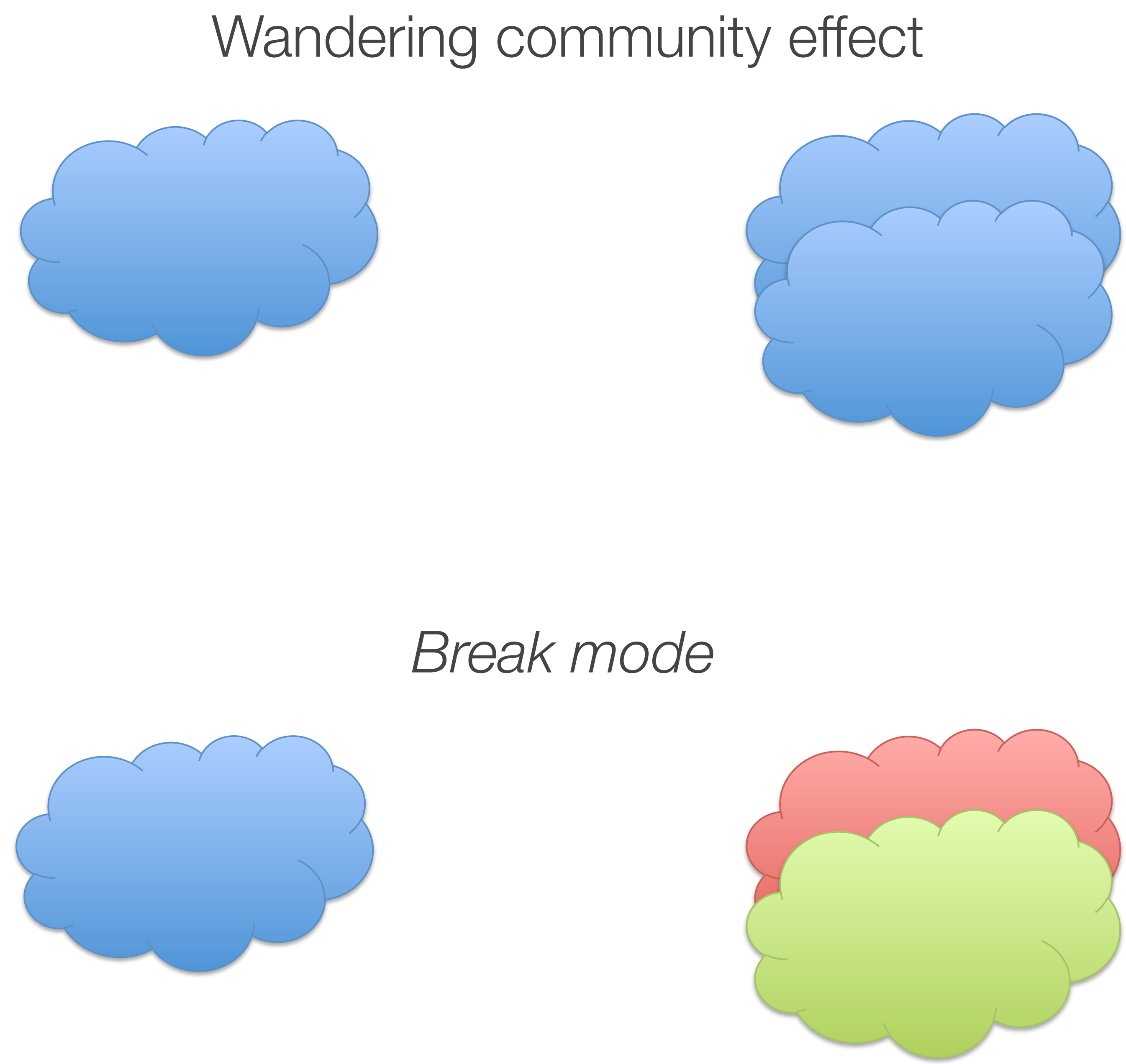
Wandering community effect

Break mode

Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect

- Community size limitation: nodes farther away than *d* hops create a new community
  → can be helpful for delay-constrained applications
  (e.g. VANET safety)



Additional SHARC improvements

- Focus on dynamic networks:
  - **Break mode**: detects split of a community in two disconnected subset
    → prevent the *wandering community* effect

- Community size limitation: nodes farther away than *d* hops create a new community
  → can be helpful for delay-constrained applications (e.g. VANET safety)

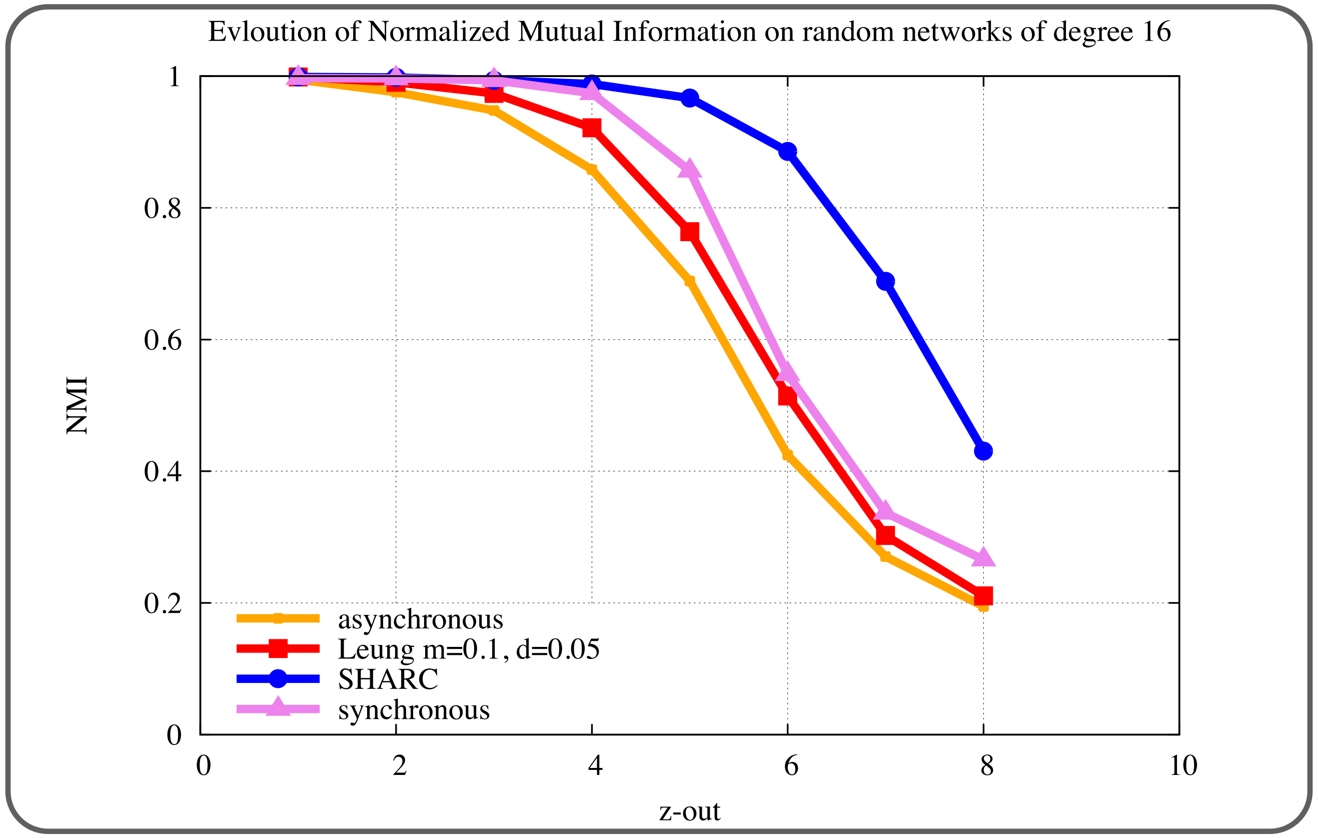
- Shift from pure formal considerations to applicative constraints

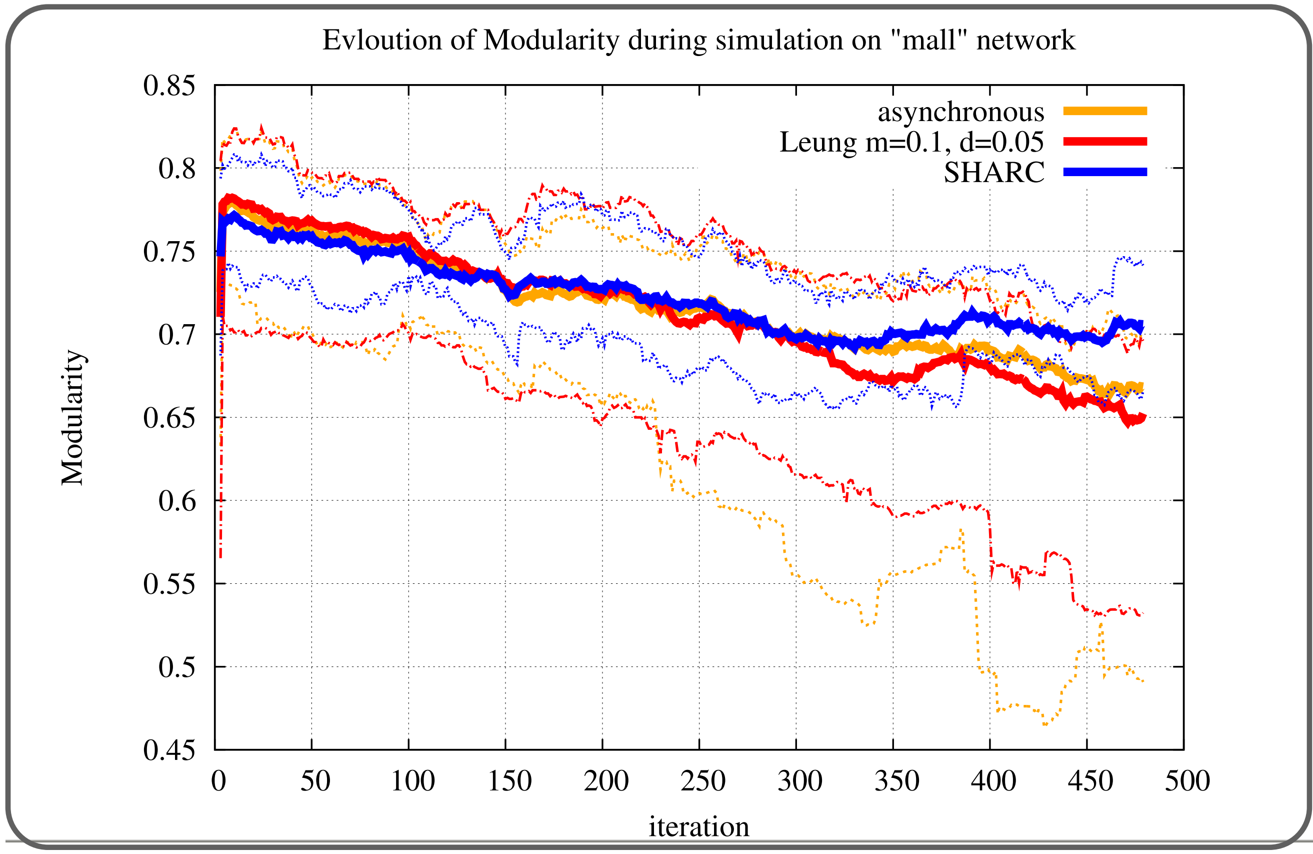
Wandering community effect

Break mode

UNIVERSITÉ DU LUXEMBOURG

SHARC performances





- Sharper assignment :
  - Higher modularity Q and NMI
  - Limits *jumbo* community effect by bounding community size
  - Important for highly dynamic networks
- More robust assignment :
  - Less standard deviation in results (topology, protocol seed)
- Results submitted as conference paper in Dec. 2009 (WoWMoM)
  - Acceptance pending

SAw-SHARC: Stability Aware SHARC

SAw-SHARC: Stability Aware SHARC

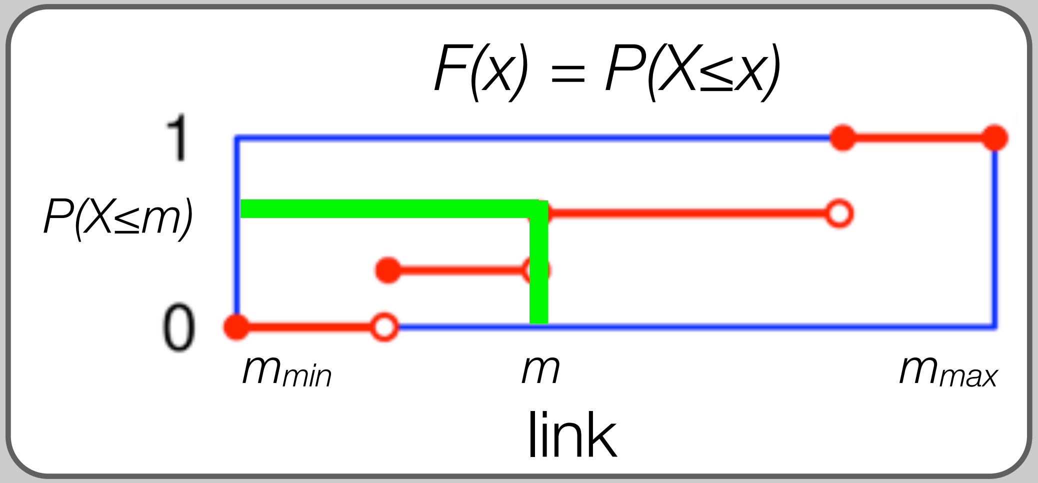
- Consider **link-stability as edge weight** in community assignment process
  - Normalized **stability estimator** (used to modulate the n_{sim} metric)
  - Auto-adaptive to current network conditions
  - Independent from underlying metric: age, average age, SNR (if X-layer), ...

UNIVERSITÉ DU LUXEMBOURG

SAw-SHARC: Stability Aware SHARC

- Consider **link-stability as edge weight** in community assignment process
  - Normalized **stability estimator** (used to modulate the n_{sim} metric)
  - Auto-adaptive to current network conditions
  - Independent from underlying metric: age, average age, SNR (if X-layer), ...

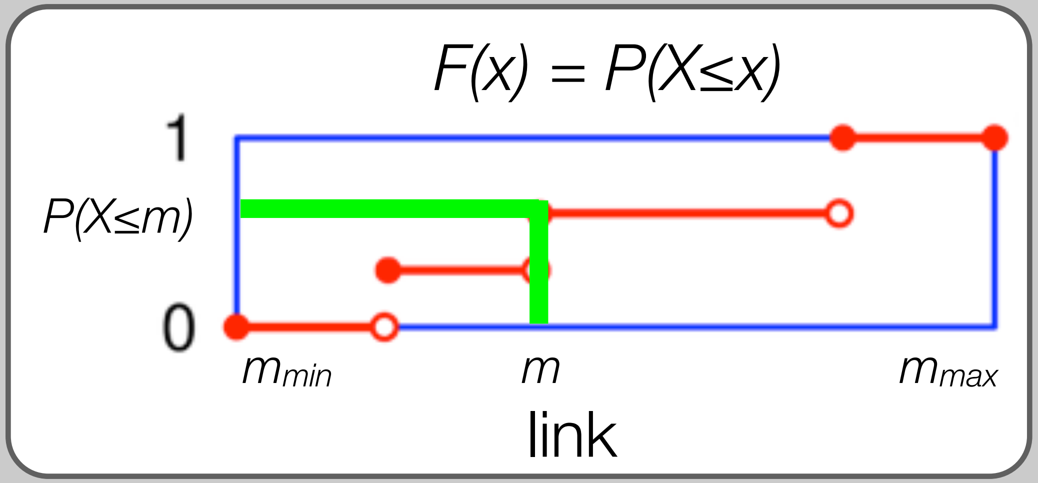
- Construct a CDF of link metrics amongst edges and use it as stability estimator
  - Gives the ratio of edges whose stability is equal or below the current edge value



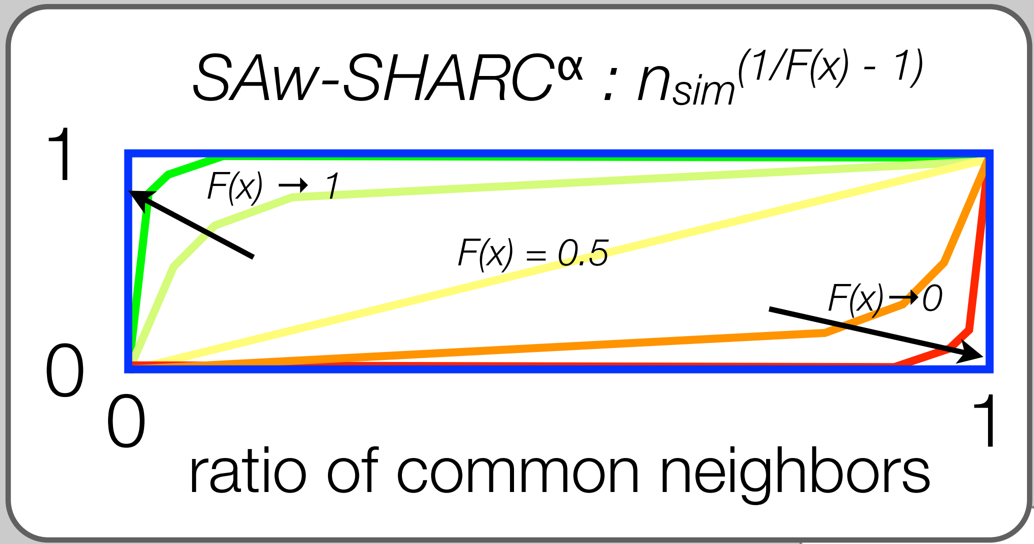
SAw-SHARC: Stability Aware SHARC

- Consider **link-stability as edge weight** in community assignment process
  - Normalized **stability estimator** (used to modulate the n_{sim} metric)
  - Auto-adaptive to current network conditions
  - Independent from underlying metric: age, average age, SNR (if X-layer), ...

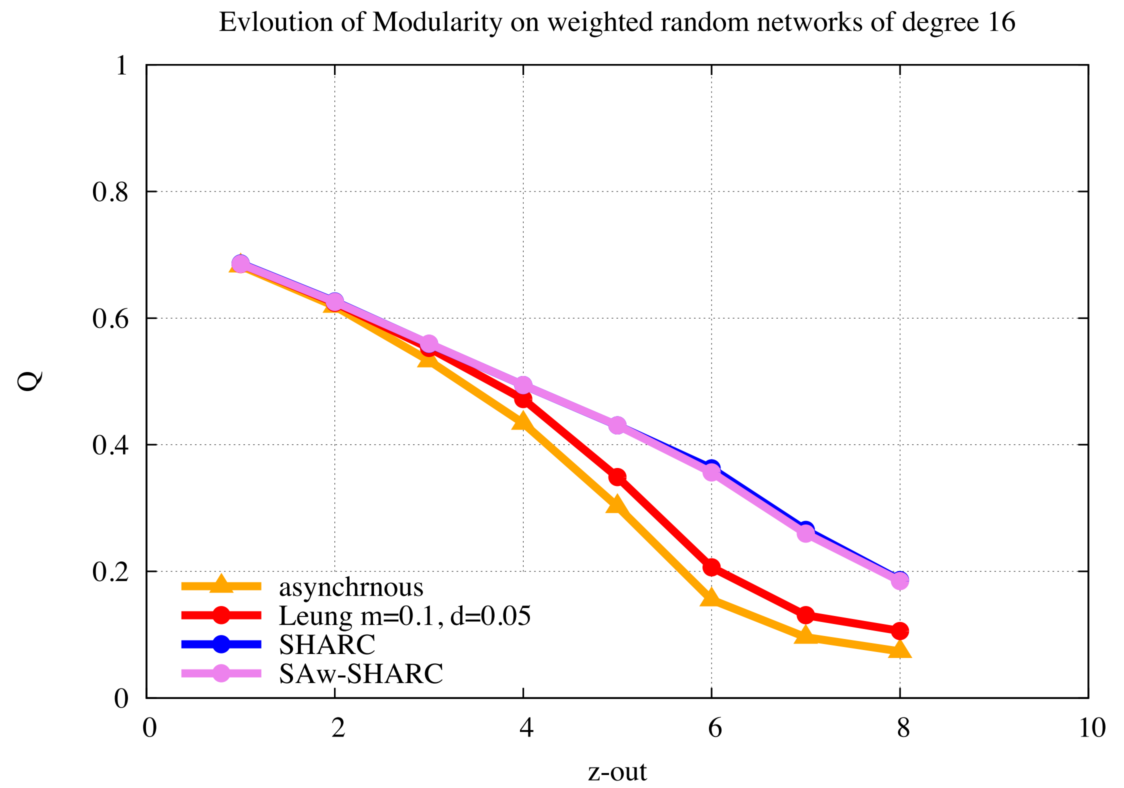
- Construct a CDF of link metrics amongst edges and use it as stability estimator
  - Gives the ratio of edges whose stability is equal or below the current edge value

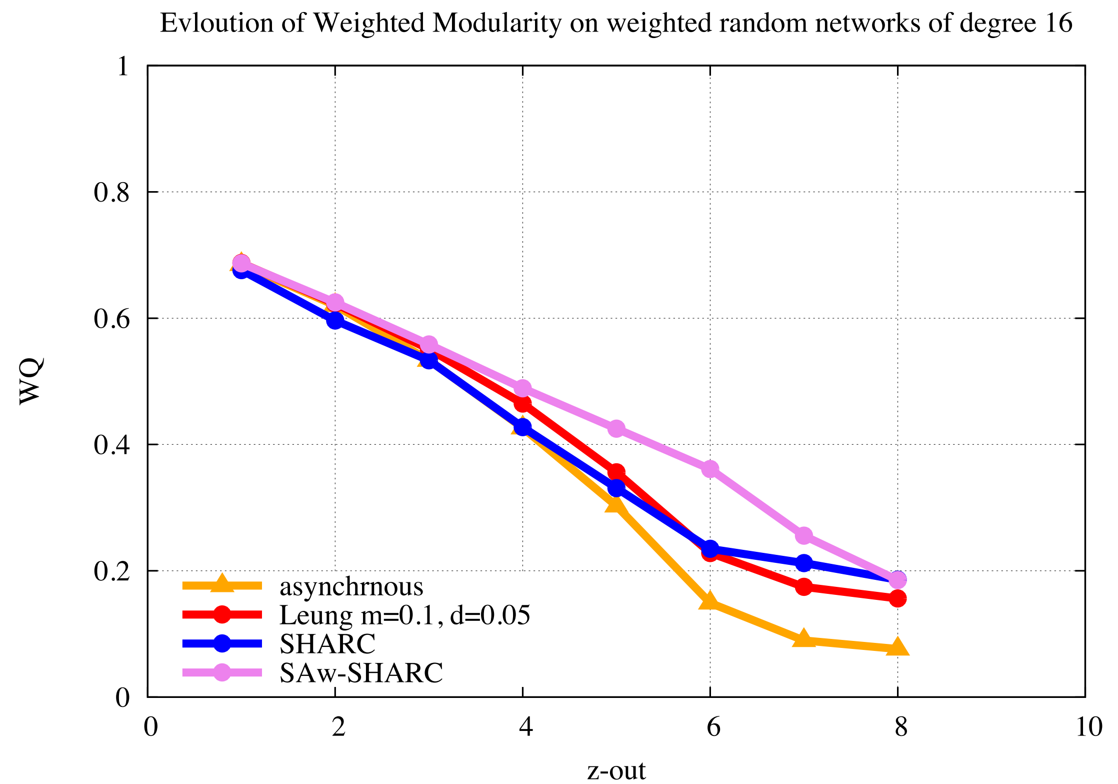


- CDF is used as modulation exponent of the neighborhood similarity measure
  - Neighbors with stable links contribute more to their community label score
  - Similar notions as in the α model of Watts and Strogatz for small-world



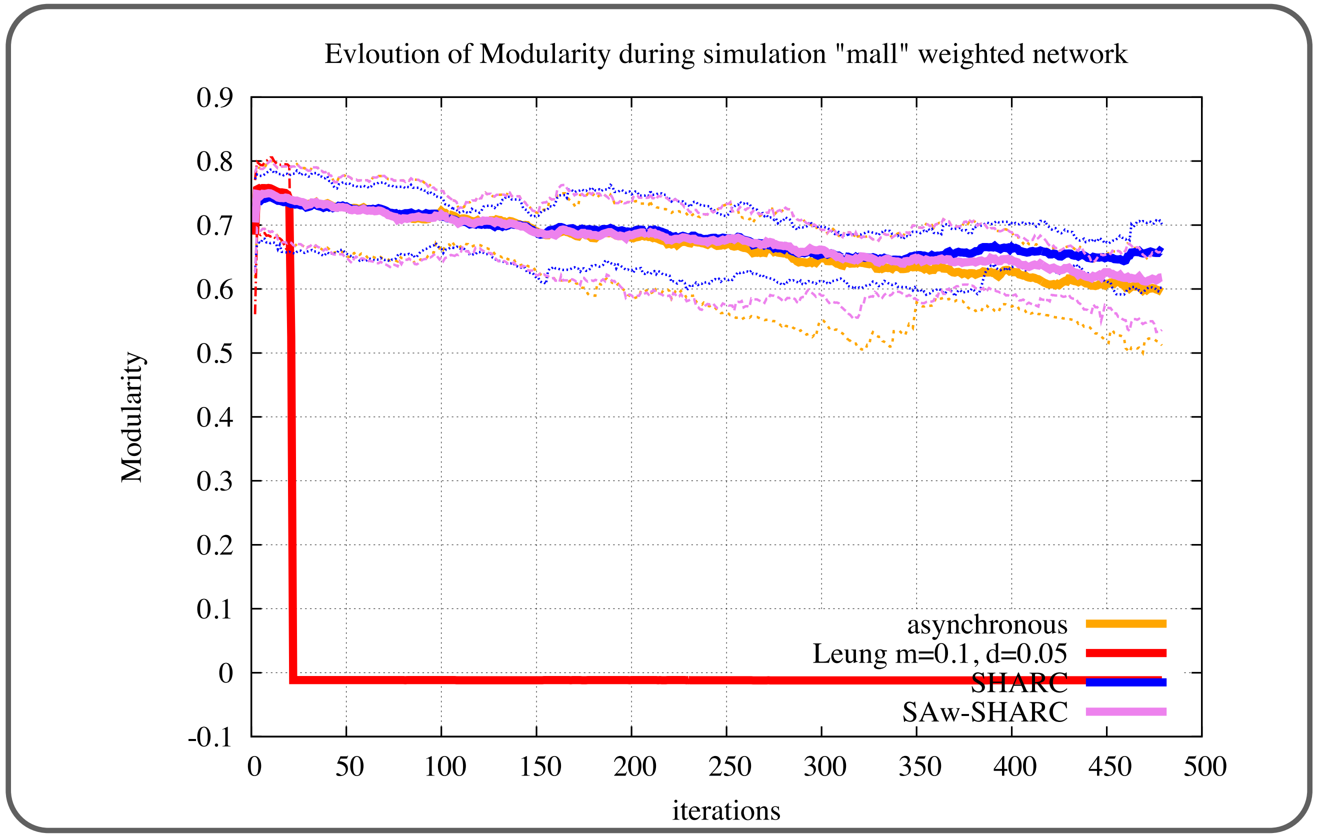
SAw-SHARC performances (static networks)

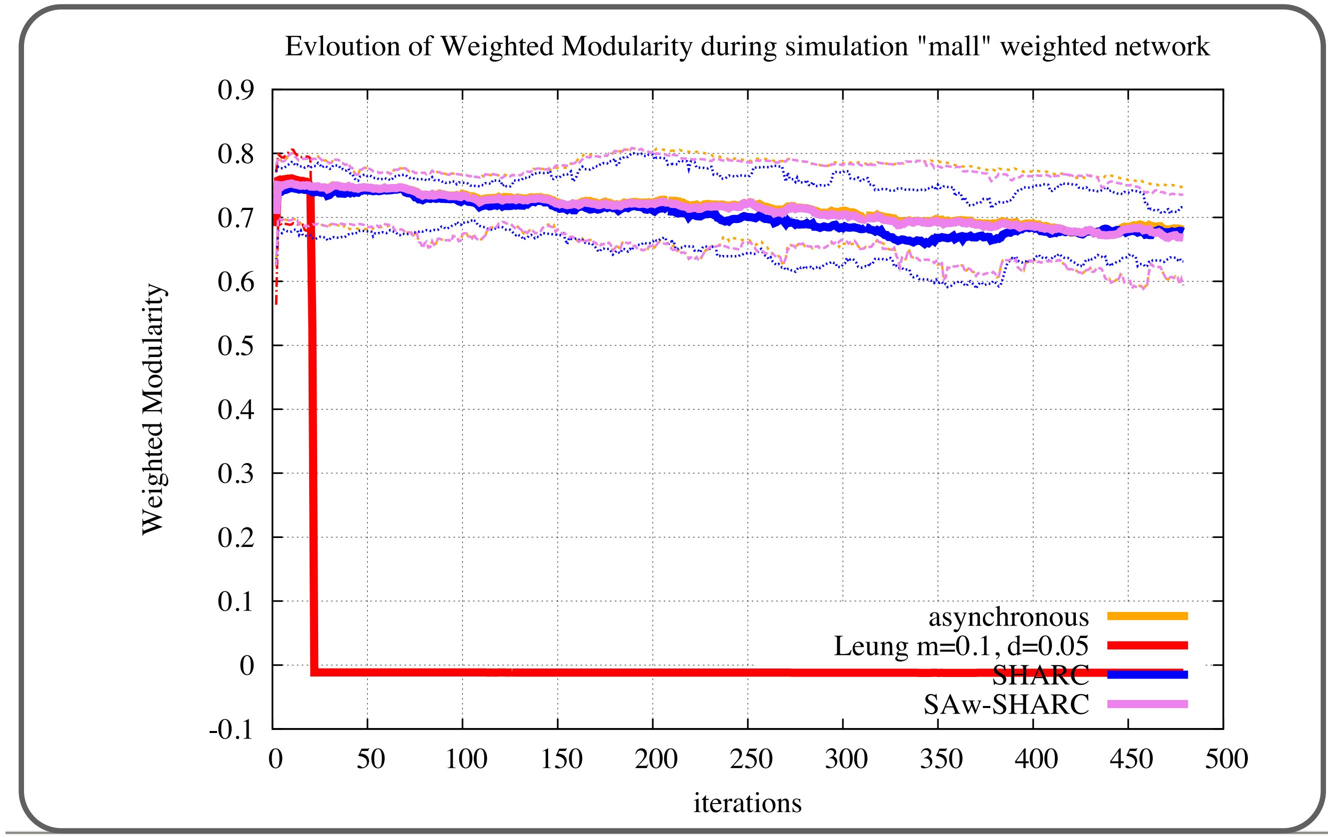




- Random networks with varying *z-out* and uniformly distributed edge weight
- SHARC and SAw-SHARC modularity Q are equivalent
- SAw-SHARC yields better results in terms of *weighted modularity WQ*
  - especially when compared with other *weight-aware* algorithms

SAw-SHARC performances (dynamic networks)





- "Mall" scenario with edge lifetime as quality metric
- SAw-SHARC performs good in terms of modularity *Q* and weighted modularity *WQ*
- Different trade-off between unweighted and weighted metrics
- Strange behavior for one of the algorithms on dynamic networks
- Need to investigate under more aggressive dynamic networks

Perspectives / Future work

- Use edge weight to represent **shared content/interest** between nodes
  - Application to MoSoNets (Mobile Social Networks)
  - Need a function to express the interest as a scalar (Locality Sensitive Hashing ?)
  - Maybe a mixed approach **stability+interest** in a **multi-objective** problem

- Introduce **long-term interaction** metrics to the assignment process
  - Favor people with regular interaction pattern to be grouped in the same community

- Use of community structures to **help algorithms with scalability**
  - Experiments on the tree/community structures matching
  - Scalability in k(connected)-m(dominant)-CDS creation

- Use of community structures for **reputation and trust**
  - Can we derive (partially ?) node reputation from its community reputation ?
  - Rough idea, further investigation needed

Contact information

- Guillaume-Jean Herbiet
  <guillaume.herbiet@uni.lu>
  Office E008
  Campus Kirchberg
  6, rue Coudenhove-Kalergi
  L-1359 Luxembourg

- http://herbiet.gforge.uni.lu

Thanks for your attention

SHARC Mathematical Model (1/2)

SHARC Mathematical Model (1/2)

- Considering a network $\mathcal{N} = (E, V)$

SHARC Mathematical Model (1/2)

- Considering a network $\mathcal{N} = (E, V)$
- We define the **adjacency matrix** $\mathbf{A}$

$$a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are adjacent} \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

SHARC Mathematical Model (1/2)

- Considering a network $\mathcal{N} = (E, V)$
- We define the **adjacency matrix** $\mathbf{A}$
- We define the set of communities $\mathcal{C} \subset V$

$$a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are adjacent} \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

SHARC Mathematical Model (1/2)

- Considering a network $\mathcal{N} = (E, V)$
- We define the **adjacency matrix** $\mathbf{A}$
- We define the set of communities $\mathcal{C} \subset V$

$$a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are adjacent} \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

- We define the **strength matrix** $\mathbf{S}$ where s_{ik} is the strength with which node belongs to community

SHARC Mathematical Model (1/2)

- Considering a network $\mathcal{N} = (E, V)$
- We define the **adjacency matrix** $\mathbf{A}$

$$a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are adjacent} \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

- We define the set of communities $\mathcal{C} \subset V$
- We define the **strength matrix** $\mathbf{S}$ where s_{ik} is the strength with which node belongs to community
- We define the **assignment matrix** $\mathbf{C}$

$$c_{ik} = \begin{cases} 1 & \text{if } i \text{ is in community } k \\ 0 & \text{otherwise} \end{cases}$$

SHARC Mathematical Model (1/2)

- Considering a network $\mathcal{N} = (E, V)$
- We define the **adjacency matrix** $\mathbf{A}$ $\quad a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are adjacent} \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$
- We define the set of communities $\mathcal{C} \subset V$
- We define the **strength matrix** $\mathbf{S}$ where s_{ik} is the strength with which node belongs to community
- We define the **assignment matrix** $\mathbf{C}$ $\quad c_{ik} = \begin{cases} 1 & \text{if } i \text{ is in community } k \\ 0 & \text{otherwise} \end{cases}$
- Then $\forall i \in V, \forall k \in \mathcal{C}$

$$s_{ik} = \sum_{j \in V} a_{ij} c_{jk} \left(1 - \frac{\sum_{p \in V} a_{ip}(1 - a_{jp}) + \sum_{q \in V} a_{jq}(1 - a_{iq})}{\sum_{n \in V} a_{in} + \sum_{m \in V} a_{jm}} \right)$$

SHARC Mathematical Model (1/2)

- Considering a network $\mathcal{N} = (E, V)$
- We define the **adjacency matrix** $\mathbf{A}$ $\quad a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are adjacent} \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$
- We define the set of communities $\mathcal{C} \subset V$
- We define the **strength matrix** $\mathbf{S}$ where s_{ik} is the strength with which node belongs to community
- We define the **assignment matrix** $\mathbf{C}$ $\quad c_{ik} = \begin{cases} 1 & \text{if } i \text{ is in community } k \\ 0 & \text{otherwise} \end{cases}$
- Then $\forall i \in V, \forall k \in \mathcal{C}$

$$s_{ik} = \sum_{j \in V} a_{ij} c_{jk} \left(\underbrace{\boxed{1 - \frac{\sum_{p \in V} a_{ip}(1 - a_{jp}) + \sum_{q \in V} a_{jq}(1 - a_{iq})}{\sum_{n \in V} a_{in} + \sum_{m \in V} a_{jm}}}}_{n_{sim}(i,j)} \right)$$

SHARC Mathematical Model (1/2)

- Considering a network $\mathcal{N} = (E, V)$
- We define the **adjacency matrix** $\mathbf{A}$ $\quad a_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are adjacent} \\ 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$
- We define the set of communities $\mathcal{C} \subset V$
- We define the **strength matrix** $\mathbf{S}$ where s_{ik} is the strength with which node belongs to community
- We define the **assignment matrix** $\mathbf{C}$ $\quad c_{ik} = \begin{cases} 1 & \text{if } i \text{ is in community } k \\ 0 & \text{otherwise} \end{cases}$
- Then $\forall i \in V, \forall k \in \mathcal{C}$

$$s_{ik} = \sum_{j \in V} a_{ij} c_{jk} \left(\underbrace{\boxed{1 - \frac{\sum_{p \in V} a_{ip}(1 - a_{jp}) + \sum_{q \in V} a_{jq}(1 - a_{iq})}{\sum_{n \in V} a_{in} + \sum_{m \in V} a_{jm}}}}_{n_{sim}(i,j)} \right)$$

- And

$$c_{ik} = \begin{cases} 1 & \text{if } s_{ik} = max_{c \in \mathcal{C}}(s_{ic}) \\ 0 & \text{otherwise} \end{cases}$$

SHARC Mathematical Model (2/2)

- Mathematically, the community assignment problem sums up to computing the **assignment matrix** $\mathbf{C}$ at each iteration step
- Problem is neither quadratic nor linear
- Proof of convergence is hard
- Need to investigate for mathematical programming model