DRCN 2011

October 10-12, 2011

Krakow, Poland

Optimization of network protection against virus spread

E. Gourdin (Orange Labs, France)

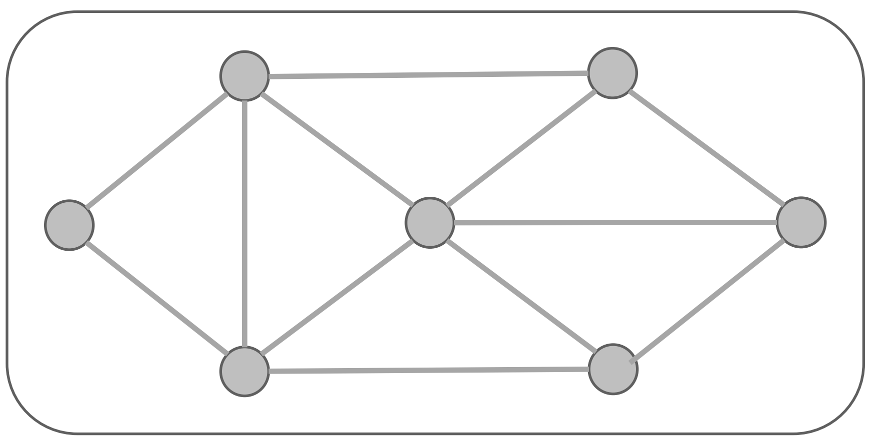
J. Omic, P. Van Mieghem (TU Delft)

Outline

- setting:
  - N-intertwined model
- curing infection:
  - optimization models
- theoretical results
  - characterize optimal « curing vs infection » functions
- algorithms
  - a few words
- examples and numerical results

Setting: notations

- **Undirected graph G=(N,L)**
  - N: set of vertices (nodes)
  - L: set of links (edges)
  - $A = (a_{ij})$: adjacency matrix ($a_{ij}=1$ means $\{i,j\} \in L$)
  - A symmetric, no loops ($a_{ii}=0$)
  - $N(i) = \{j \in N: a_{ij}=1\}$: neighborhood of i



Setting: N-Intertwined Model

- [1] « **Virus spread in networks** », P. Van Mieghem, J. Omic, R. Kooij, IEEE/ATM Trans. Netw., vol. 17, no. 1, pp. 1-14, 2009.
  - X_i: state of node i (X_i = 0 healty, X_i = 1 infected)
  - $v_i(t) = Pr[X_i(t) = 1]$
  - Curing process: Poisson with rate δ_i
  - Infection process: Poisson with rate β_i
  - cases:
    - Homogeneous: ($\delta_i = \delta$, $\beta_i = \beta$) done in [1]
    - Heterogeneous: (δ_i, β_i) done in [2]
    - **Here: (δ_i, $\beta_i = \beta$)**

Diffusion Libre

Setting: N-Intertwined Model

- **Differential equation**
  - Curing process: Poisson with rate δ_i
  - Infection process: Poisson with rate β

$$\frac{dv_i(t)}{dt} = \beta\left(1 - v_i(t)\right)\sum_{j=1}^{N} a_{ij} v_j(t) - \delta_i v_i(t)$$

infection rate

probability of node i not being infected

probability that neighboors of i are infected

curing rate

probability of node i being infected

Setting: N-Intertwined Model

- **Differential equation**
  - Curing process: Poisson with rate δ_i
  - Infection process: Poisson with rate β
  - Steady-state:

$$\frac{dv_i(t)}{dt} = 0$$

  - Solution:

$$v_i = \frac{\beta \sum_{j=1}^{N} a_{ij} v_j}{\beta \sum_{j=1}^{N} a_{ij} v_j + \delta_i} = \frac{\beta\, v(N(i))}{\beta\, v(N(i)) + \delta_i}$$

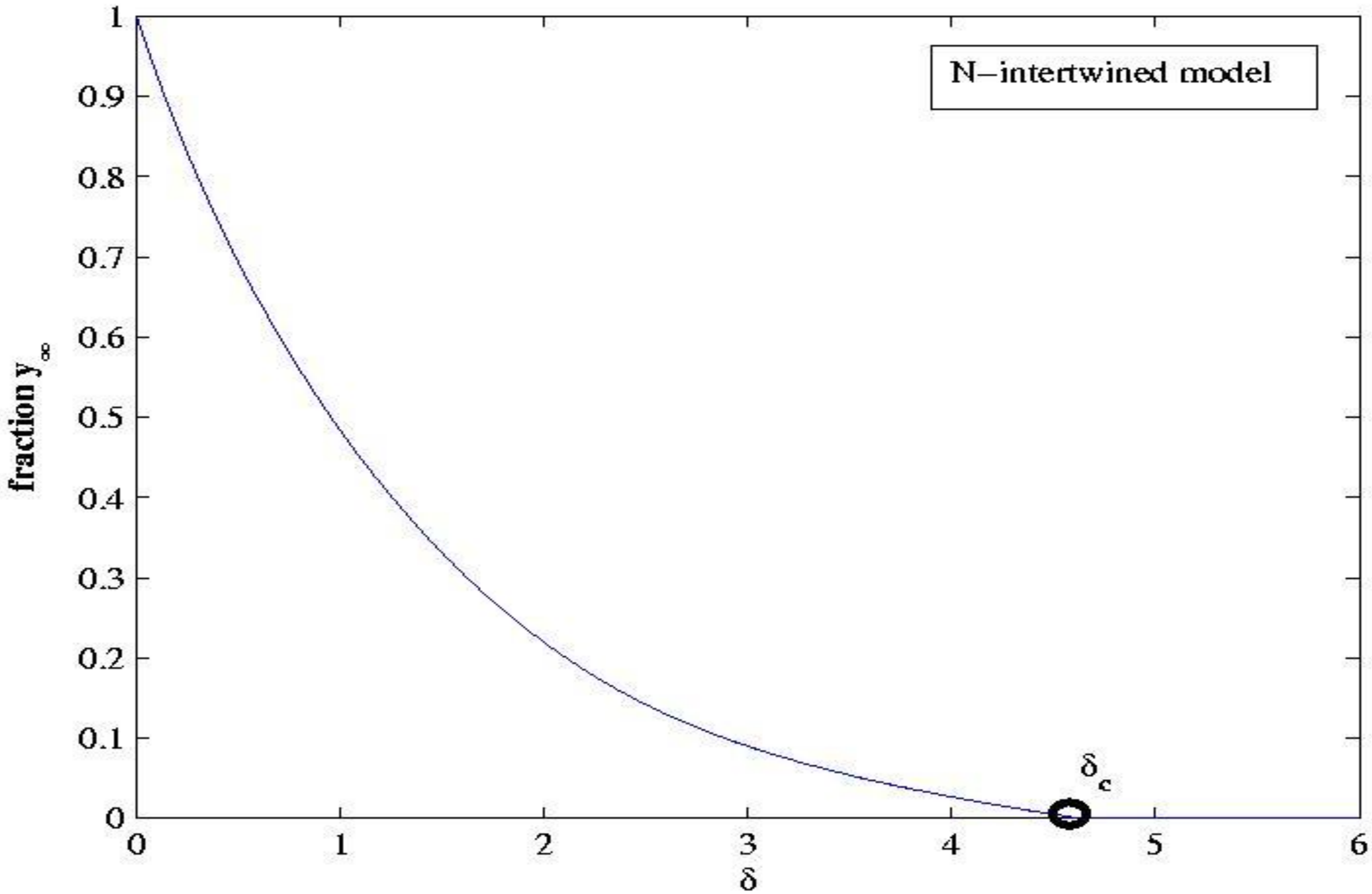
notation

v is an implicit function of δ !!!

Setting: N-Intertwined Model

- **In the homogeneous case ($\delta_i = \delta$):**

fraction of infected nodes



Optimizations models

➔ (P1): Minimize total infection, given a curing « budget »

$$\min \quad \sum_{j \in N} v_j$$
$$\text{s.t.} \quad \sum_{j \in N} \delta_j = 2L\alpha\beta$$
$$0 \le \delta_j \le \delta_c, \qquad j \in N$$

implicit function of δ

number of links

parameter

infection rate

difficult problem !

numerical approaches can be considered…

Optimizations models

- ➔ v is an implicit function of δ:

$$v_i = \frac{\beta\, v(N(i))}{\beta\, v(N(i)) + \delta_i}$$

- ➔ but δ is an **explicit** function of v:

$$\delta_i = \beta \left(\frac{1 - v_i}{v_i} \right) v(N(i))$$

Optimizations models

- (P1): Minimize total infection, given a curing « budget »

$$\begin{aligned} \min \quad & \sum_{j \in N} v_j(\delta) \\ \text{s.t.} \quad & \sum_{j \in N} \delta_j = 2L\alpha\beta \\ & 0 \leq \delta_j \leq \delta_c, \qquad j \in N \end{aligned}$$

- (P2): Minimize curing « budget », given a level of infection

$$\begin{aligned} \min \quad & \sum_{j \in N} \delta_j(v) \\ \text{s.t.} \quad & \sum_{j \in N} v_j = N\alpha \\ & 0 \leq \delta_j(v) \leq \delta_c, \qquad j \in N \end{aligned}$$

Optimizations models

➔ **(P2): Minimize curing « budget », given a level of infection**

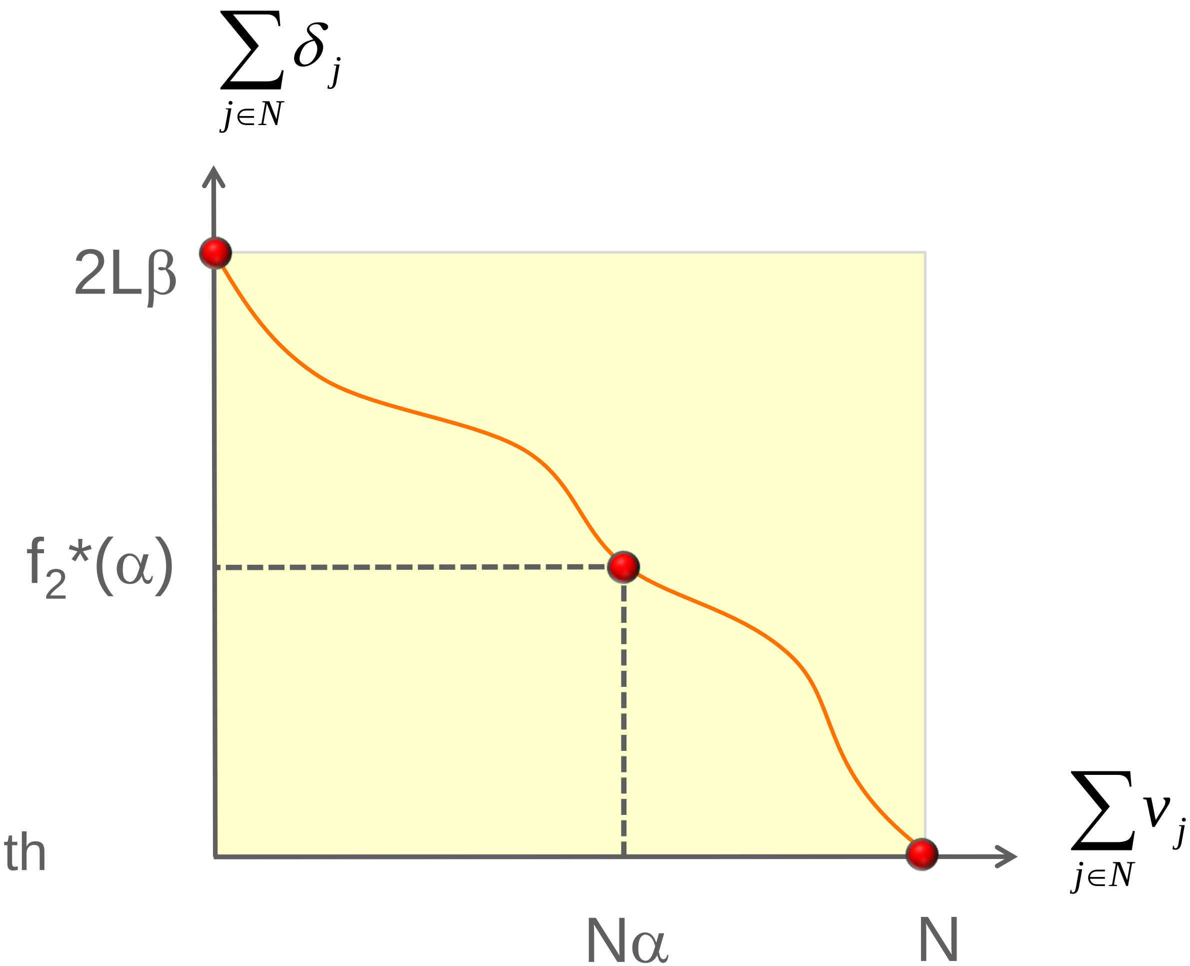
$$\begin{aligned} \min \quad & \sum_{j \in N} \beta \left(\frac{1 - v_j}{v_j} \right) v(N(j)) & \\ \text{s.t.} \quad & \sum_{j \in N} v_j = N\alpha & \\ & 0 \leq \delta_j \leq \delta_c, & j \in N \end{aligned}$$

Some theoretical results

- **denote**
  - $f_1^*(\alpha)$ the optimal value of (P_1)
  - $f_2^*(\alpha)$ the optimal value of (P_2)
- **then:**
  - f_2^* is strictly decreasing w.r.t. α
  - $f_2^*(\alpha) \leq 2L(1-\alpha)\beta$
  - optimal solutions of (P_1) and (P_2) are linked

Some theoretical results

- **Problem P_2:**
  - choose α
  - lhs = $N\alpha$
  - Solve $P_2(\alpha)$



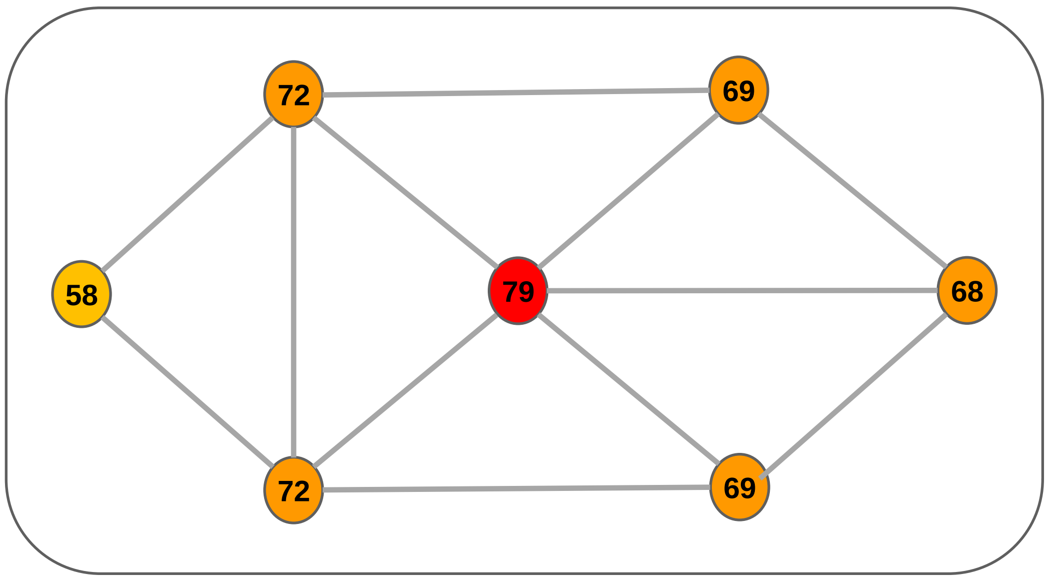
- **Problem P_1:**
  - $N\alpha$ = optimal value of P1 with param. $f_2^*(\alpha)$

Some algorithms

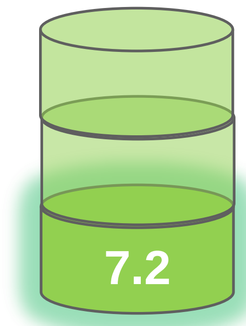
- **Random search**
- **Local search**
  - descent along projected gradient
- **Branch-and-bound (exact)**
  - divide the domain iteratively
  - evaluate function at center
  - use gradient to obtain upper-bound

Some examples

➔ **example: 7 nodes**



level
of
protection

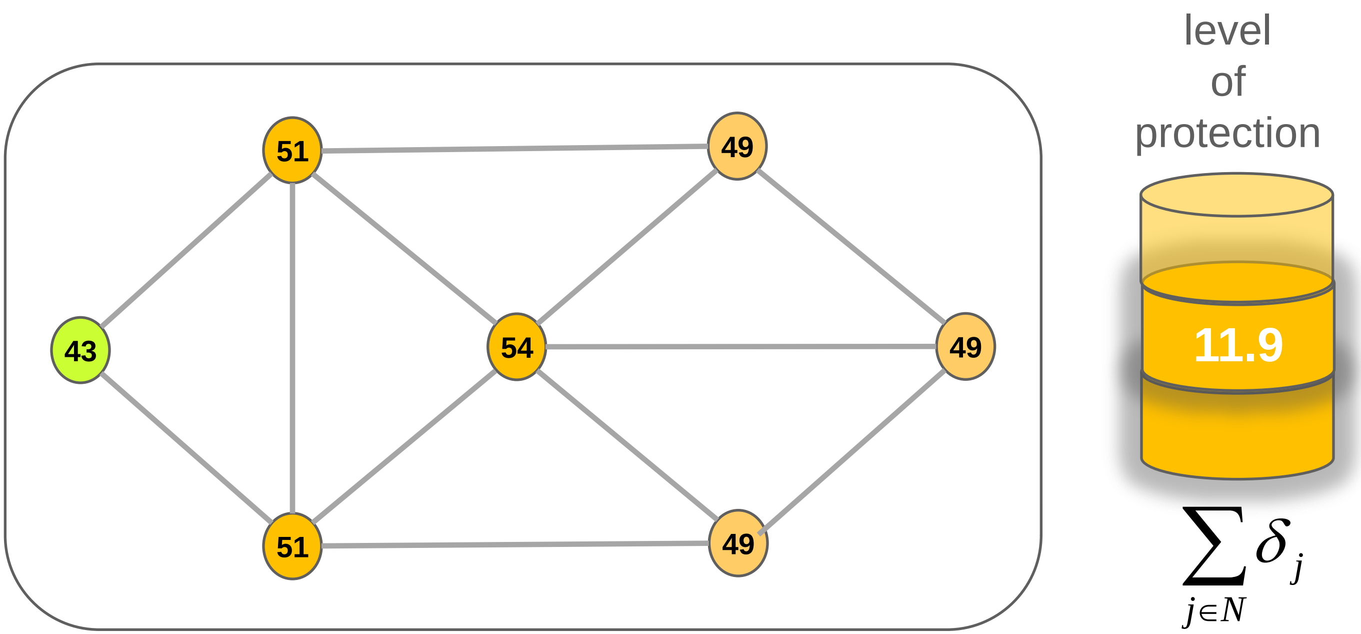


$$\sum_{j \in N} \delta_j$$

$$\alpha = 0.7, \sum_{j \in N} v_j = 4.9$$

Some examples

→ **example: 7 nodes**

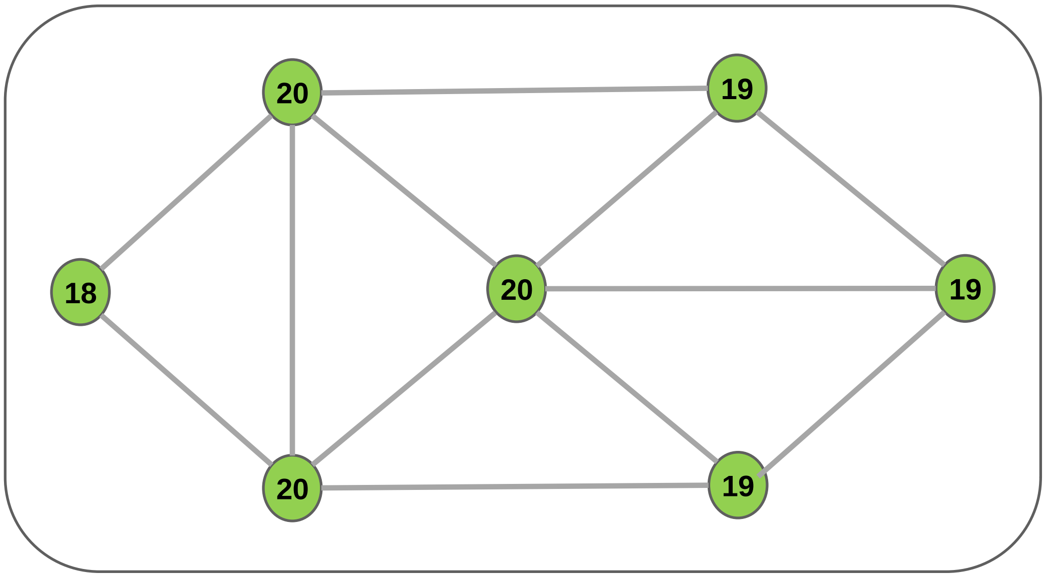


$$\sum_{j \in N} \delta_j$$

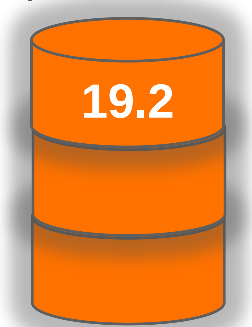
$$\alpha = 0.5, \sum_{j \in N} v_j = 3.5$$

Some examples

- example: 7 nodes



level of protection

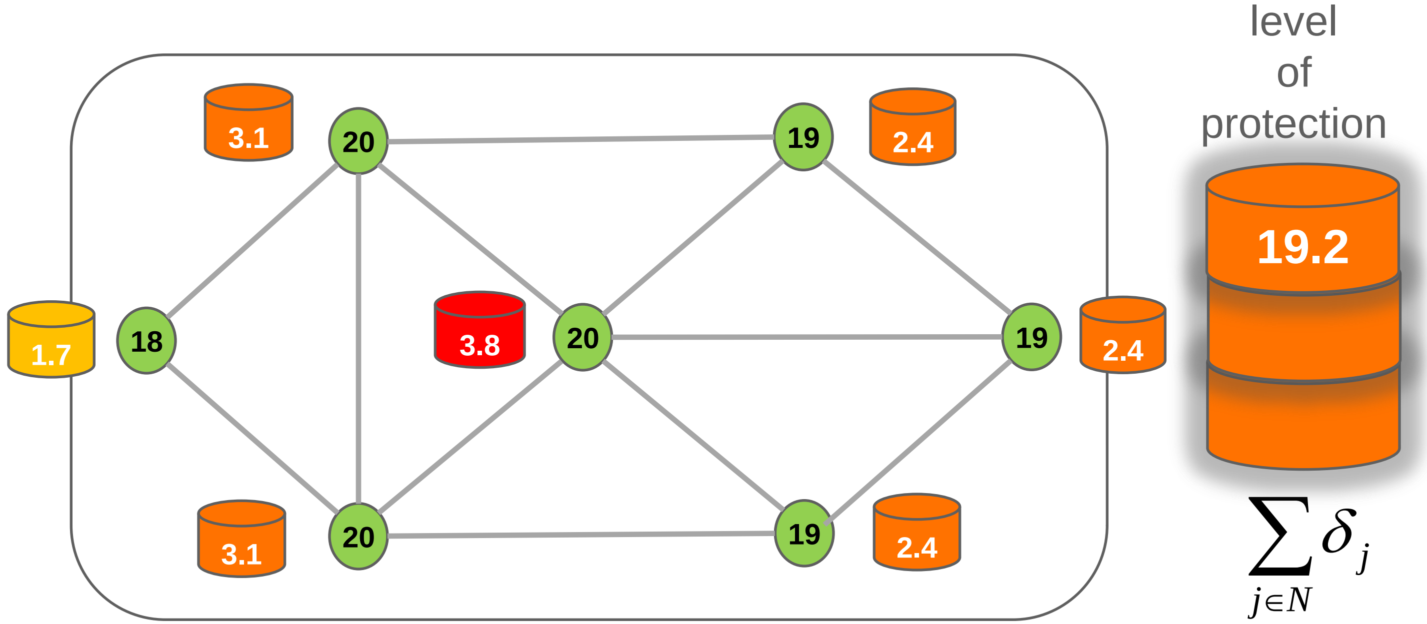


$$\sum_{j \in N} \delta_j$$

$$\alpha = 0.2, \sum_{j \in N} v_j = 1.4$$

Some examples

- ➔ **example: 7 nodes**

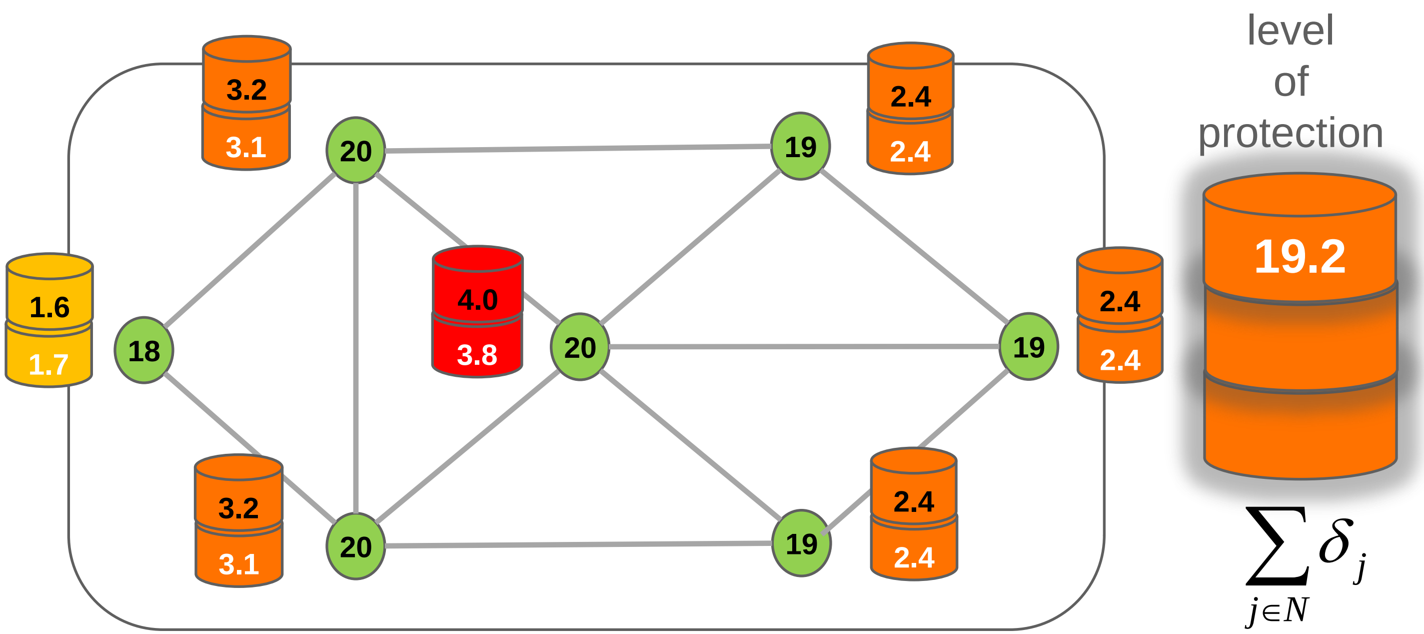


$$\sum_{j \in N} \delta_j$$

$$\alpha = 0.2, \sum_{j \in N} v_j = 1.4$$

Some examples

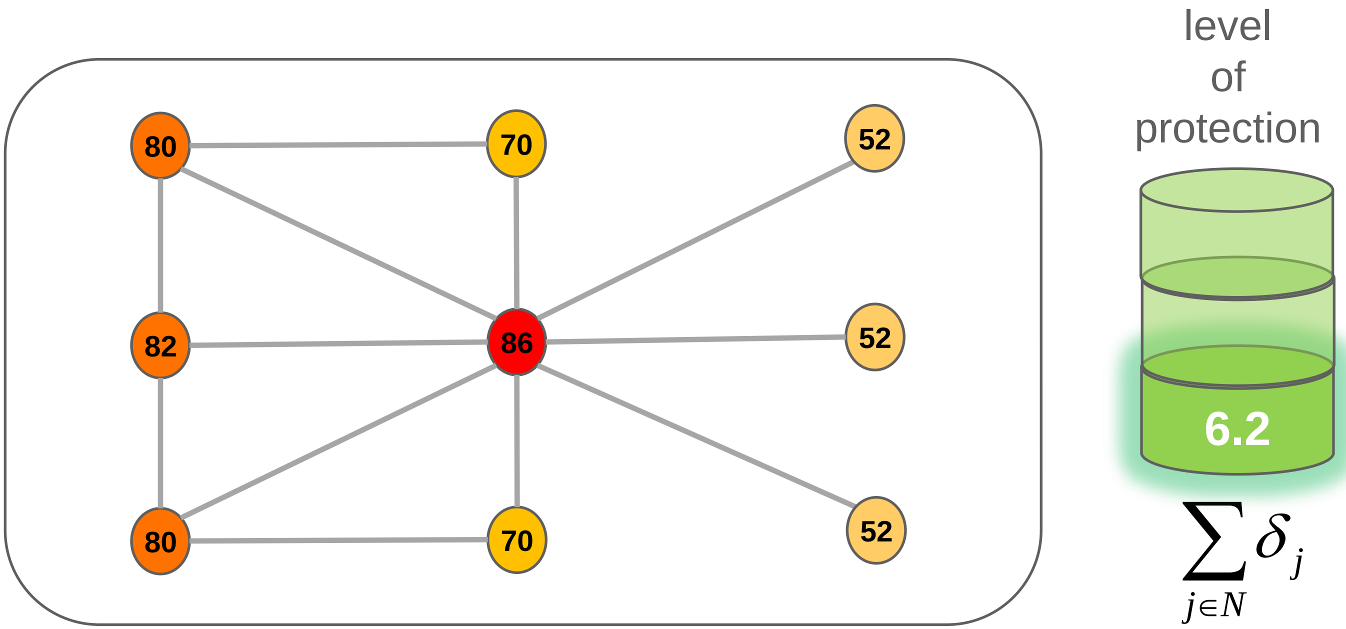
- example: 7 nodes
- protection proportional to node degree:



$$\alpha = 0.2, \sum_{j \in N} v_j = 1.4$$

Some examples

➔ **example: 9 nodes**

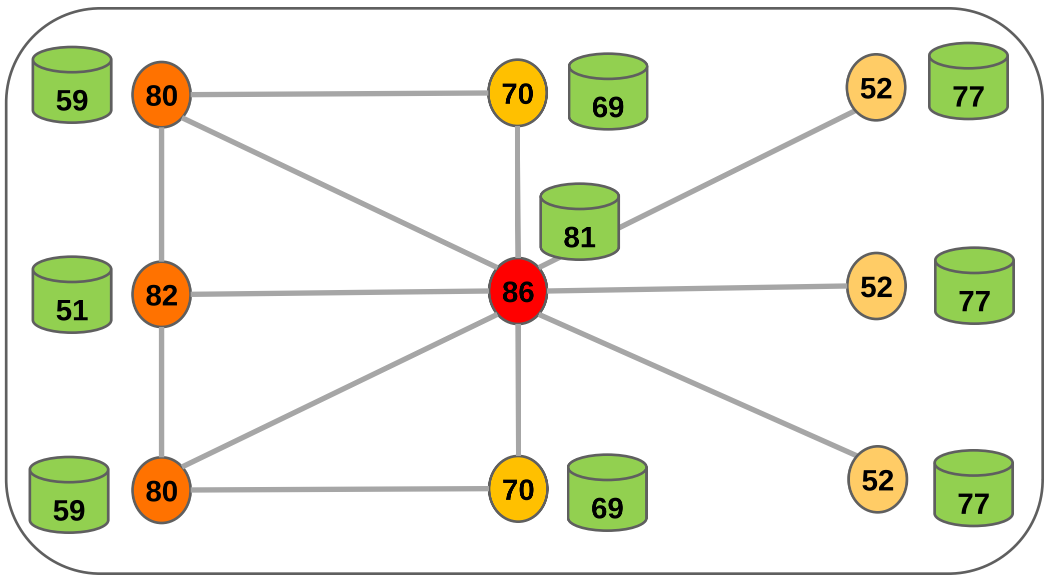


$$\sum_{j \in N} \delta_j$$

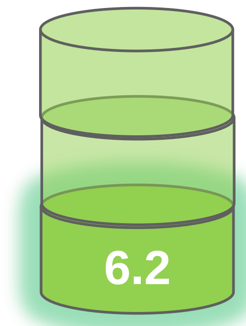
$$\alpha = 0.7, \sum_{j \in N} v_j = 6.3$$

Some examples

➔ **example: 9 nodes**



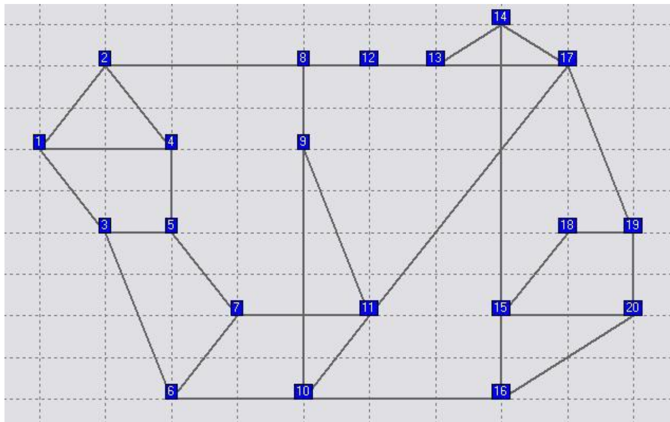
level of protection

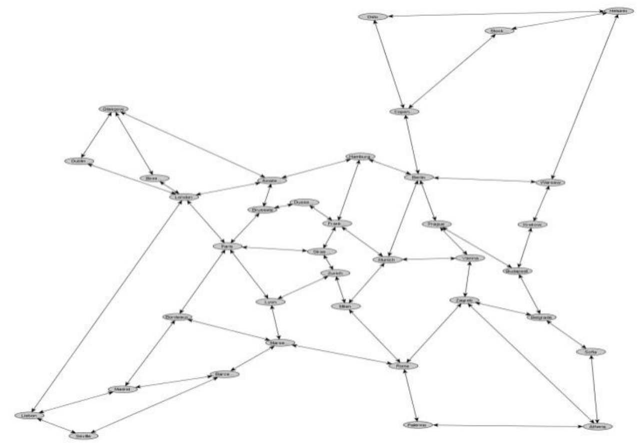


$$\sum_{j \in N} \delta_j$$

$$\alpha = 0.7, \sum_{j \in N} v_j = 6.3$$

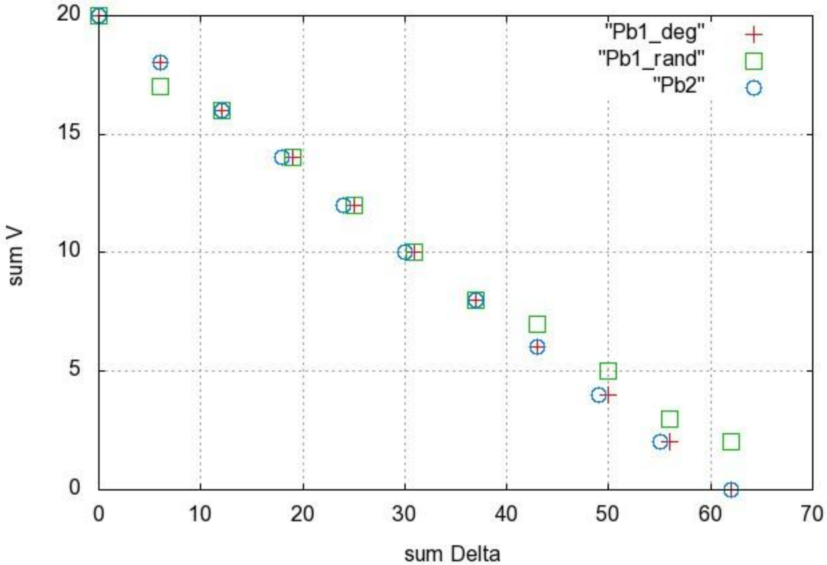
Some numerical results

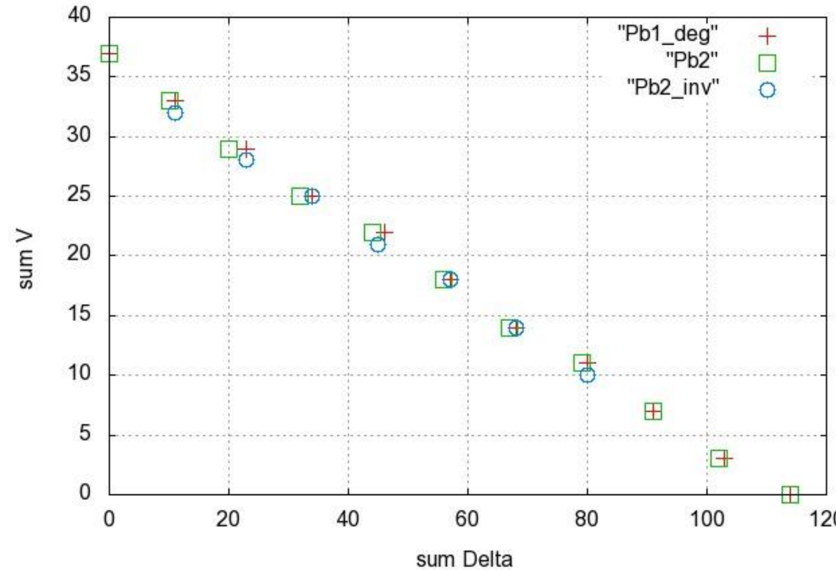




ARPAnet

Cost266

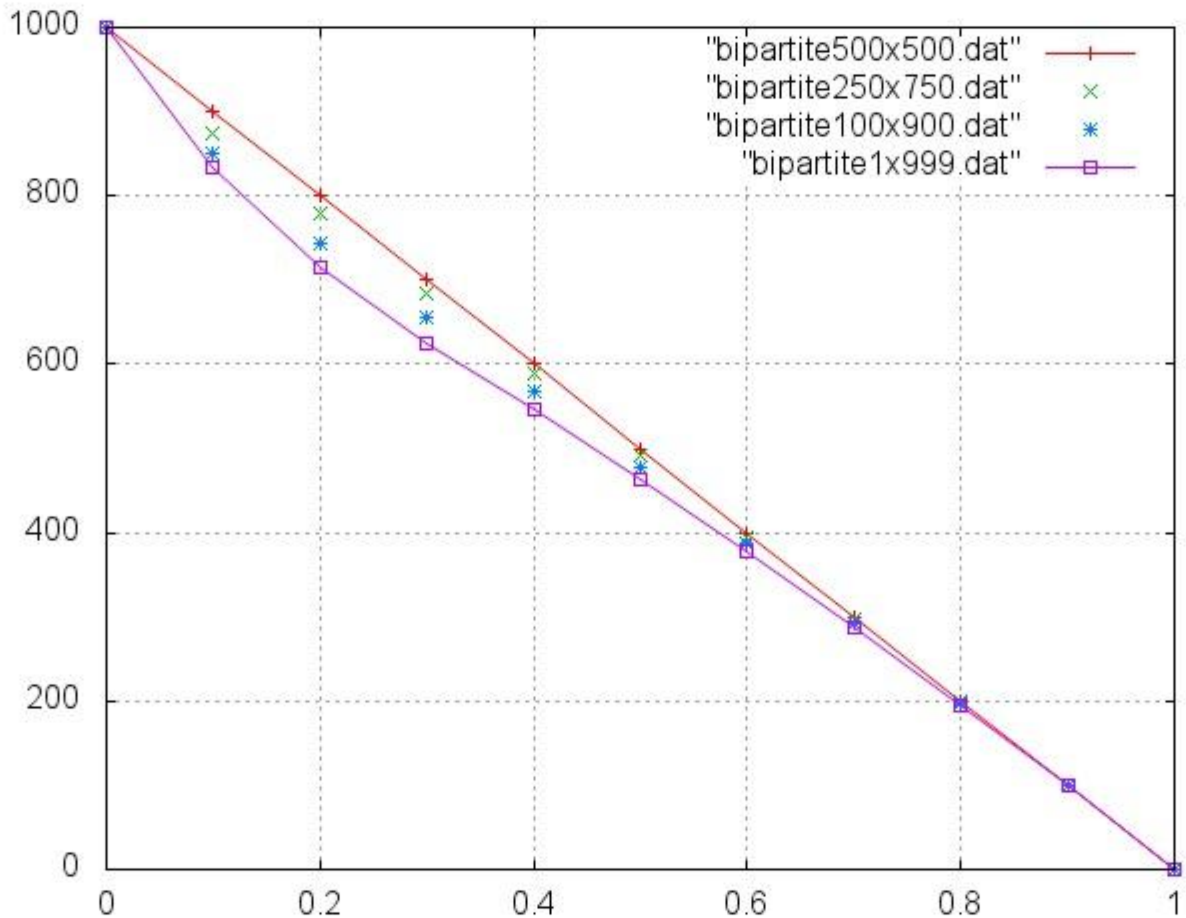




nice alignement of points !

Some numerical results

➔ **bipartite graphs: analytical solution available**




merci !