Distributed Control of the reactive power in smart microgrids

Sandro Zampieri
Universita' di Padova

In collaboration with Saverio Bolognani

Inspired by Allessandro Costabeber and Paolo Tenti

Outline

- Introduction to distributed (leaderless) decision models
  - Scientific context
  - Multi-agent systems as a distributed (leaderless) estimation and control architecture
  - Example: distributed estimation and consensus algorithm
- Application to the control of the electric power distribution networks (Grid)
  - Modeling of the grid
  - Minimization of the power losses by reactive power compensation
  - A quadratic approximation
  - A distributed algorithm for the reactive power compensation
  - Convergence of the algorithm

Distributed (leaderless) decision models

In the context of optimization, control, estimation, decision making, computation, etc, the word **DISTRIBUTED** is used with different meanings:

- The task is distributed over many agents in order to speed up the task completion (i.e. parrallel computers).
- The system itself is constituted by several interacting parts which need to be coordinated (i.e. wireless sensor networks).

In the context of the distributed decision models we can distinguish:

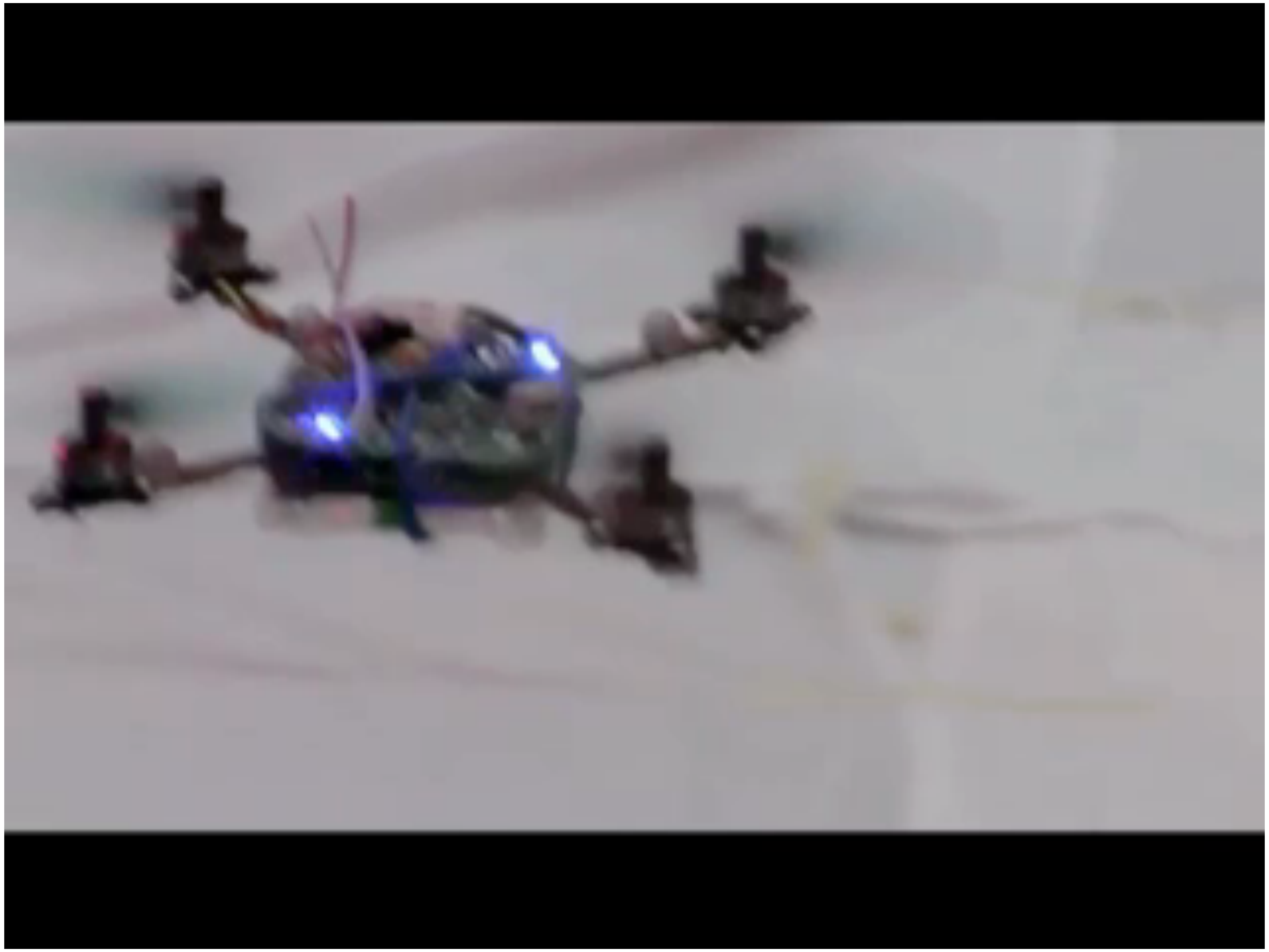
- Distributed decision models with leaders or with a hierarchy (based on spanning trees construction).
- Leaderless distributed decision models in which the agents are peers in the network. In this case the goal is not perfomance, but the robustness and the of self-organization.

Example: Communication networks

Example: robotic networks

Kiva systems

Example: robotic networks



GRASP Lab at the University of Pennsylvania

Example: wireless sensor networks

Distributed decision models

Water distribution

Traffic

Leaderless distributed decision models

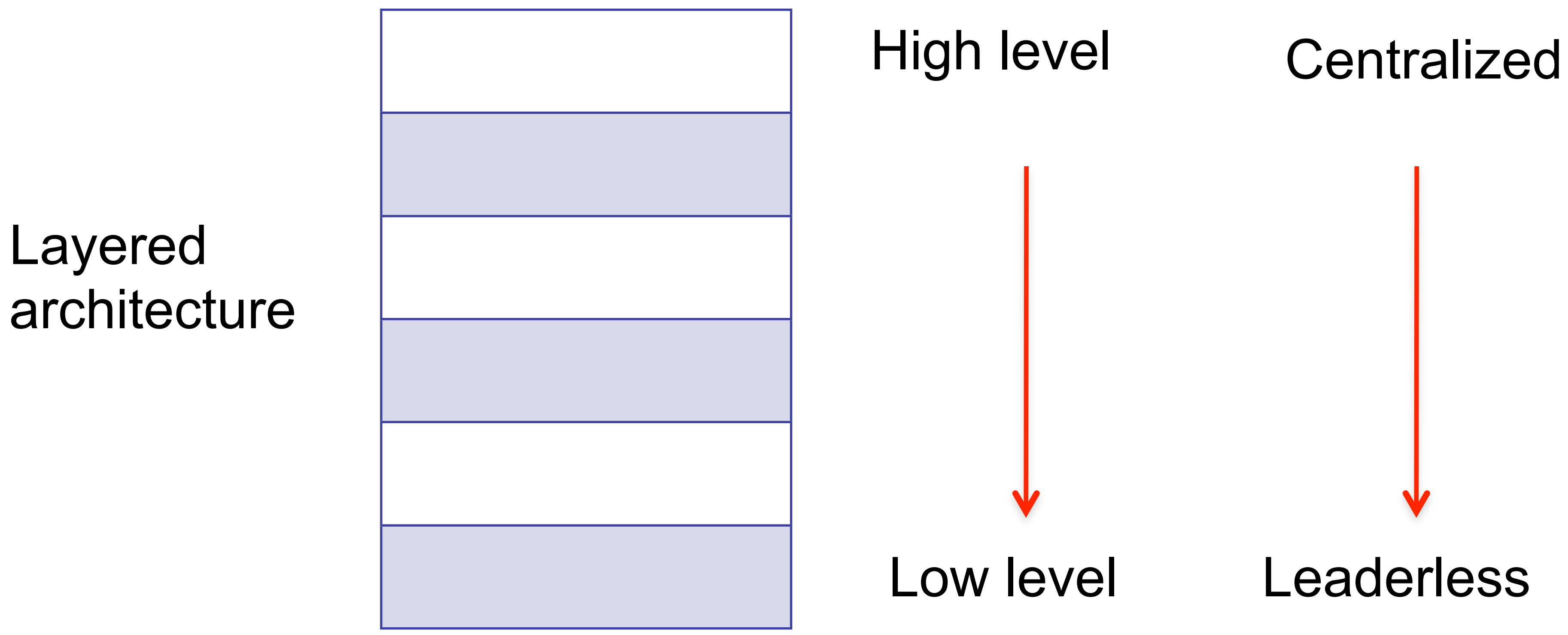
Centralized vs. Leaderless

	Required unit reliability	performance	Cost	Effort in configuration
Centralized/hierarchical	high	high	high	high
Distributed/Leaderless	low	low	low	low

Complex systems → High cost in the initial configuration

Leaderless distributed decision models

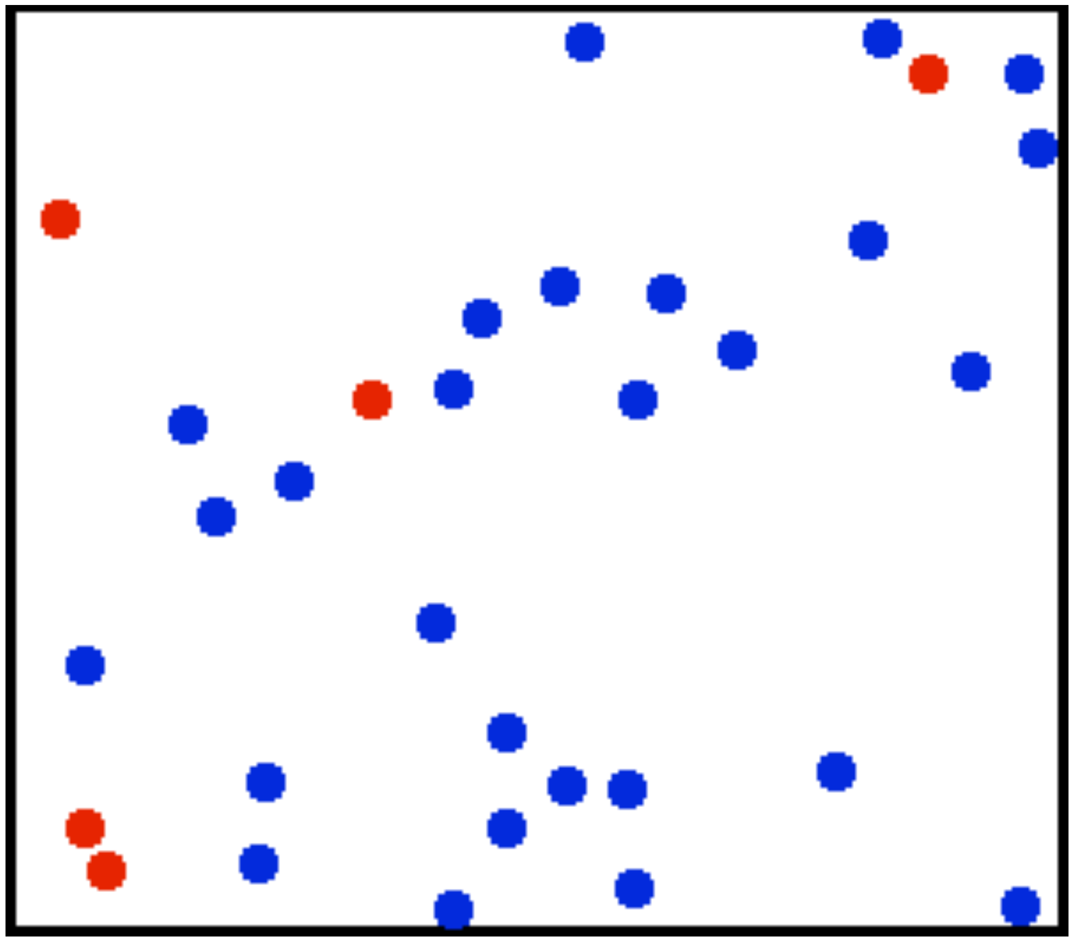
Leaderless decision models = extreme design paradigm



Scientific context

Statistical mechanics

how the local interactions of particles may yield simple thermodynamics laws describing the global **emerging** behavior.



pV = nRT

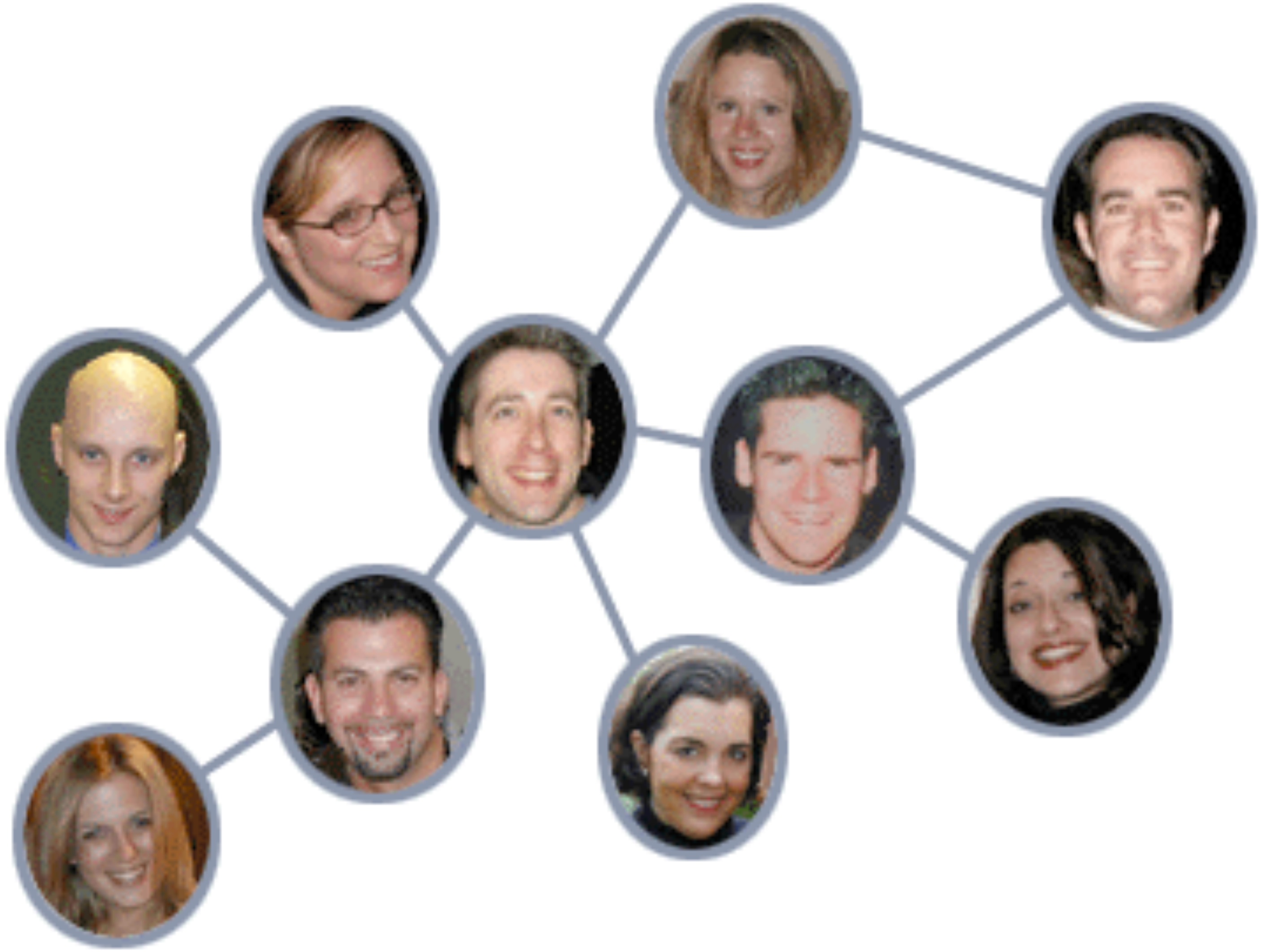
Scientific context

Cooperation: simple global behavior from local interactions

Flocking: collective animal behavior given by the motion of a large number of coordinated individuals

Scientific context

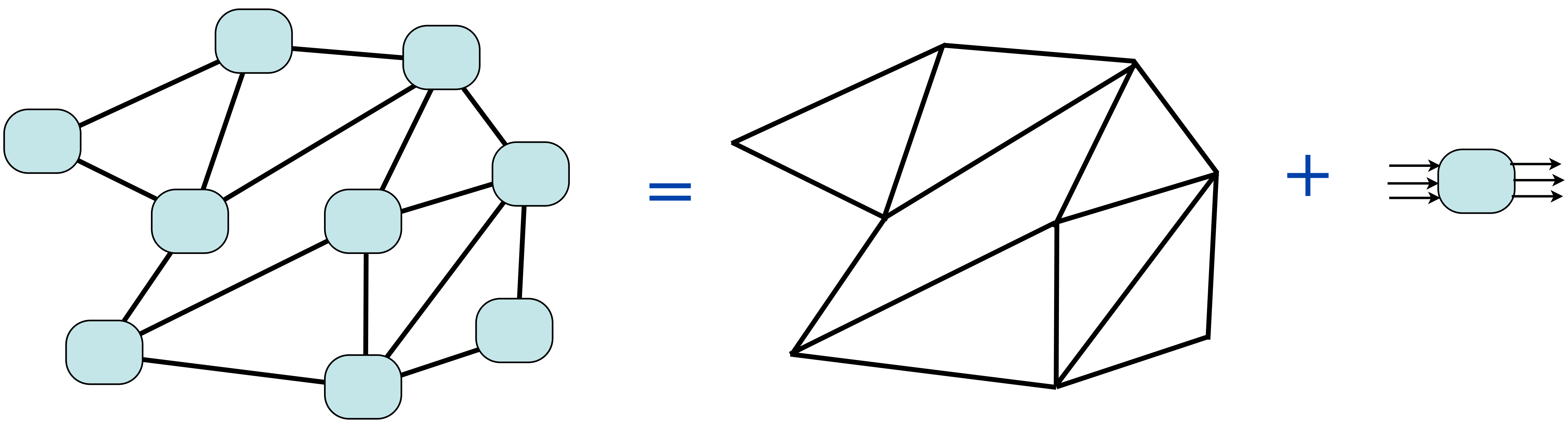
Social and economic networks: individual social and economic interactions produce global phenomena



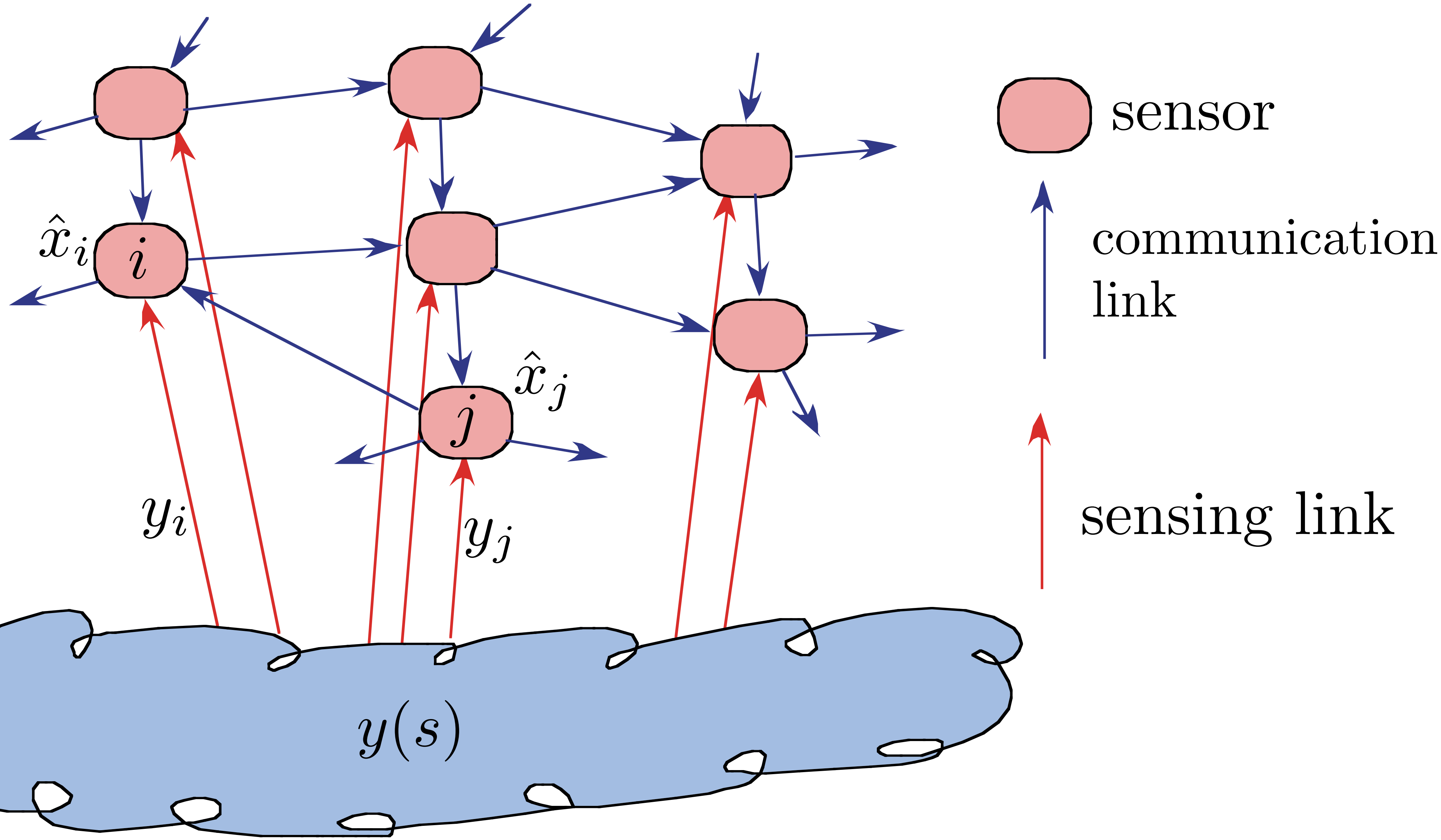
Problem description

The object of our investigations is to study the behavior of “complex” systems constituted by the interconnection of many units which are themselves dynamical systems.

The behavior of these systems will depend on the dynamics of the units and on the interconnection topology. We want to understand how these two features produce the global dynamics.



Multi-agent systems architecture for distributed estimation



$\hat{x}_i$ is opinion of the node i has of $\hat{x}$

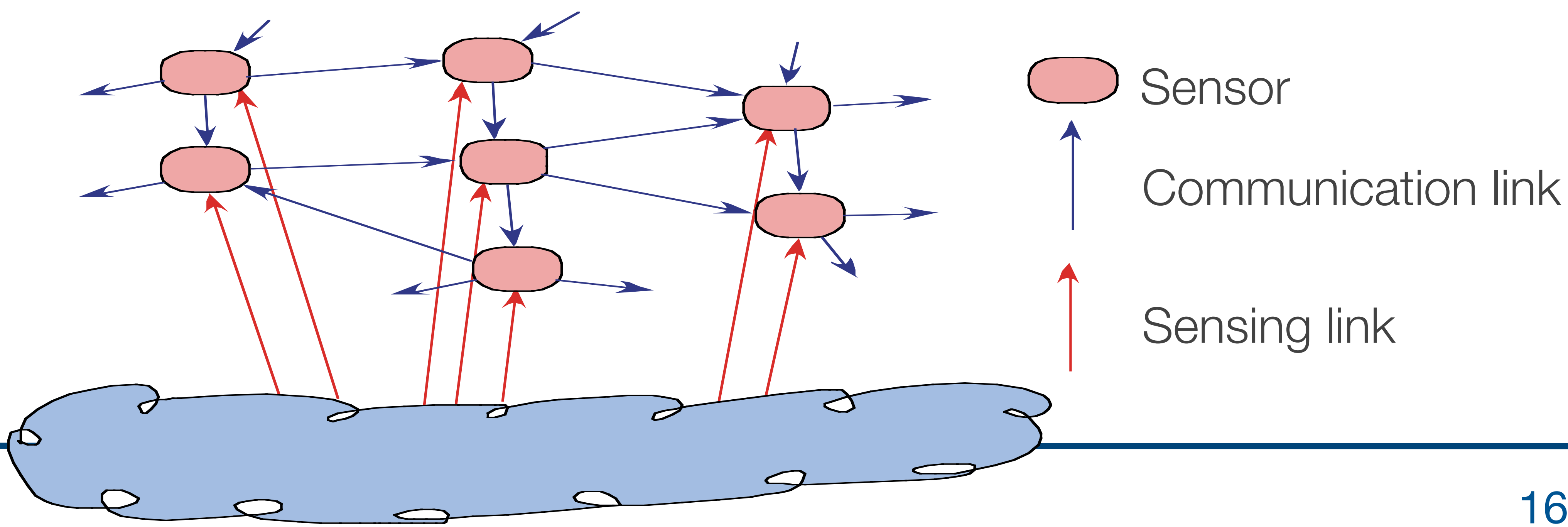
Example: distributed estimation

Assume that N sensors have to estimate a quantity $x \in \mathbb{R}$ from their noisy measurements. The result of the measurement of the sensor i is

$$\boxed{y_i = x + n_i}$$

where n_i are independent noises of zero mean and the same variance. The best estimate of x from the measurements is

$$\boxed{\hat{x} := \frac{1}{N} \sum_i y_i}$$



The consensus algorithm

GOAL: each node has to obtain the average of $y_1, \ldots, y_N$, where y_i is known only by the node i.

ALGORITHM: Each sensor produces at time t an estimate $x_i(t)$ of the average as follows

$$\boxed{\begin{array}{c} x_i(0) = y_i \qquad x_i(t+1) = \sum_{j=1}^{N} P_{ij} x_j(t) \\ \\ P_{ij} \geq 0 \qquad \sum_{j=1}^{N} P_{ij} = 1 \end{array}}$$

COMMUNICATION: $x_j(t)$ needs to be transmitted from the node i to the node j iff

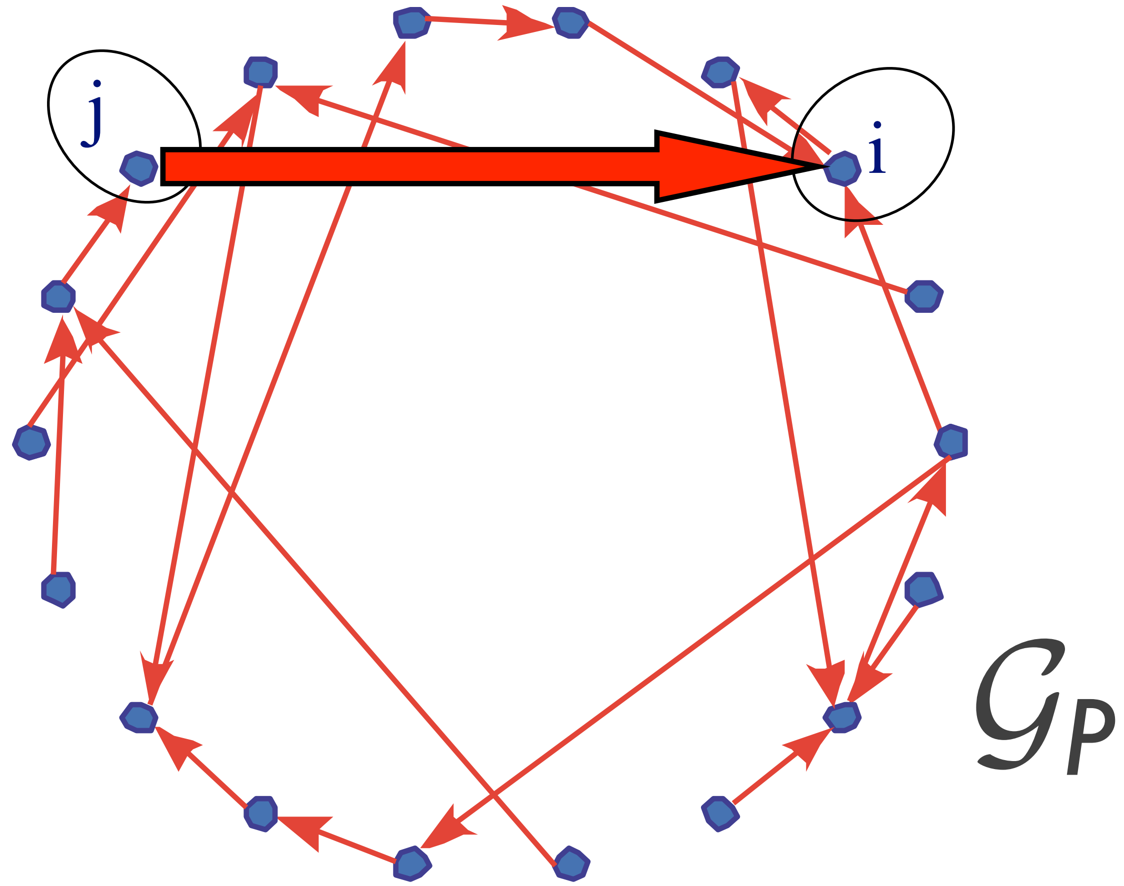
$$P_{ij} \neq 0$$

i.e. $P_{ij} = \frac{1}{N_i}$ where N_i is the numebr of neighbors of the node i.

The consensus algorithm

$$\boxed{\begin{aligned} x(t+1) &= Px(t) \\ x(0) &= y \end{aligned}}$$

where P stochastic.



If $P_{ii} > 0$ for all i and the graph $\mathcal{G}_P$ associated with P is strongly connected, then all estimates converge to the same value (consensus)

$$\boxed{x_i(t) \longrightarrow \sum_{j=1}^{N} \mu_j x_j(0)}$$

where the weights μ_j are nonnegative and sum to one.

The consensus algorithm

In case P^T is stochastic as well (for instance if P is symmetric), then $\mu_j = 1/N$ and so

$$\boxed{x_i(t) \longrightarrow \frac{1}{N}\sum_{j=1}^{N} x_j(0)}$$

Pros and cons

Advantages

1. Very robust to node and link failures and to time asynchronocity.
2. Very simple implementation (may be implemented asynchrously)
3. No need of a centralized design.
4. Incremental i.e. anytime algorithm.

Disadvantages

1. Slow convergence

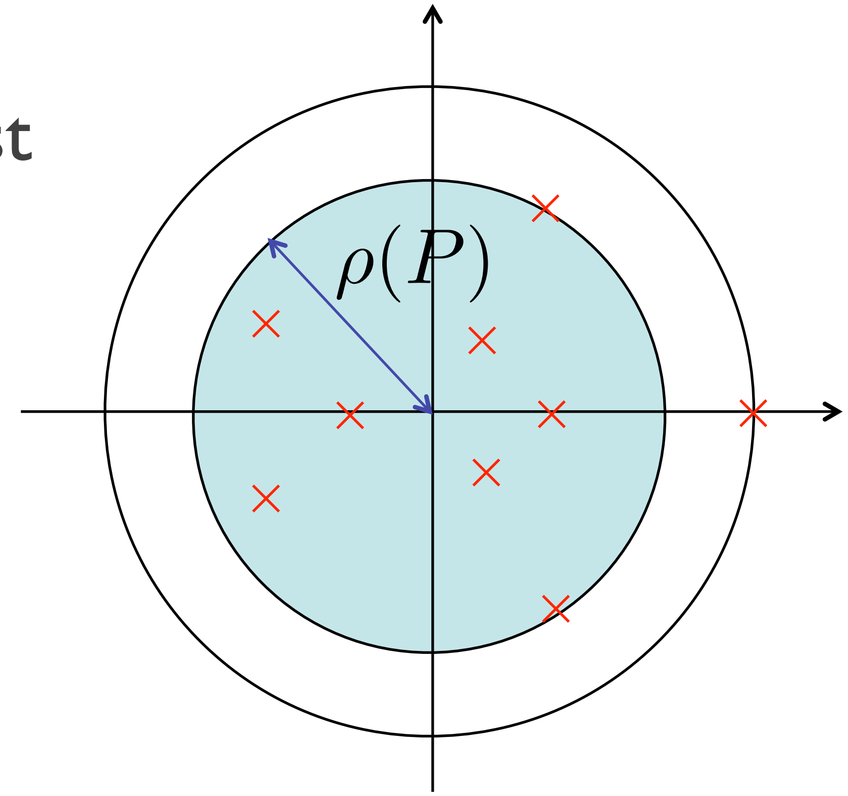
Convergence rate

$$x_i(t) \longrightarrow x_i(\infty)$$

exponentially fast with rate given by the second largest eigenvalue of P

$$\rho(P) = \max\{|\lambda| \ : \lambda \text{ are the eigenvalues of } P\}$$

$$\boxed{|x_i(t) - x_i(\infty)| \leq \mathrm{cost}\,\rho(P)^t}$$



Convergence rate

There are two types of problems:

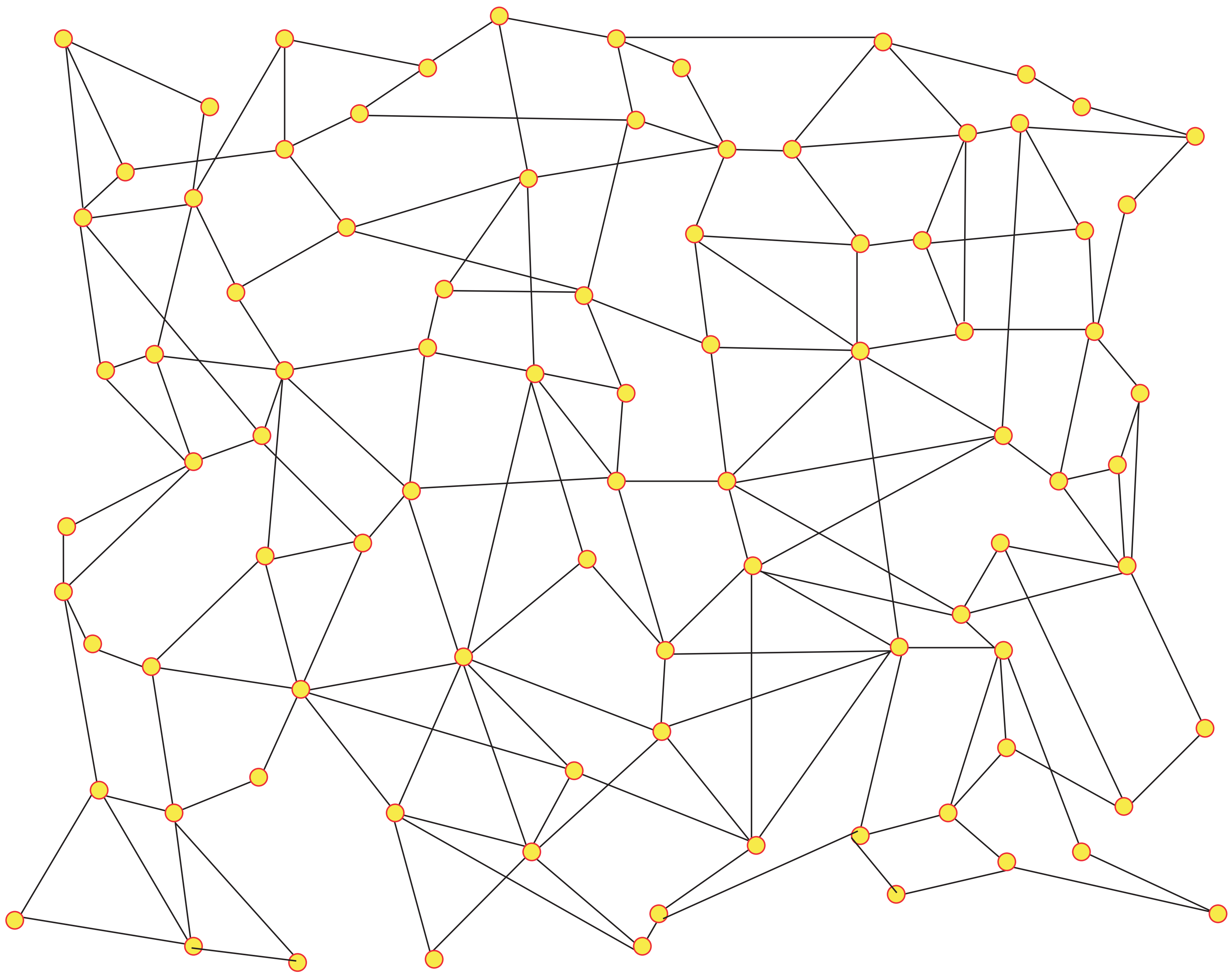
1. Optimization problems: find the matrix P in a class which optimize the performance index.

2. Influence of the network topology: find how the network topology influences the the performance index.

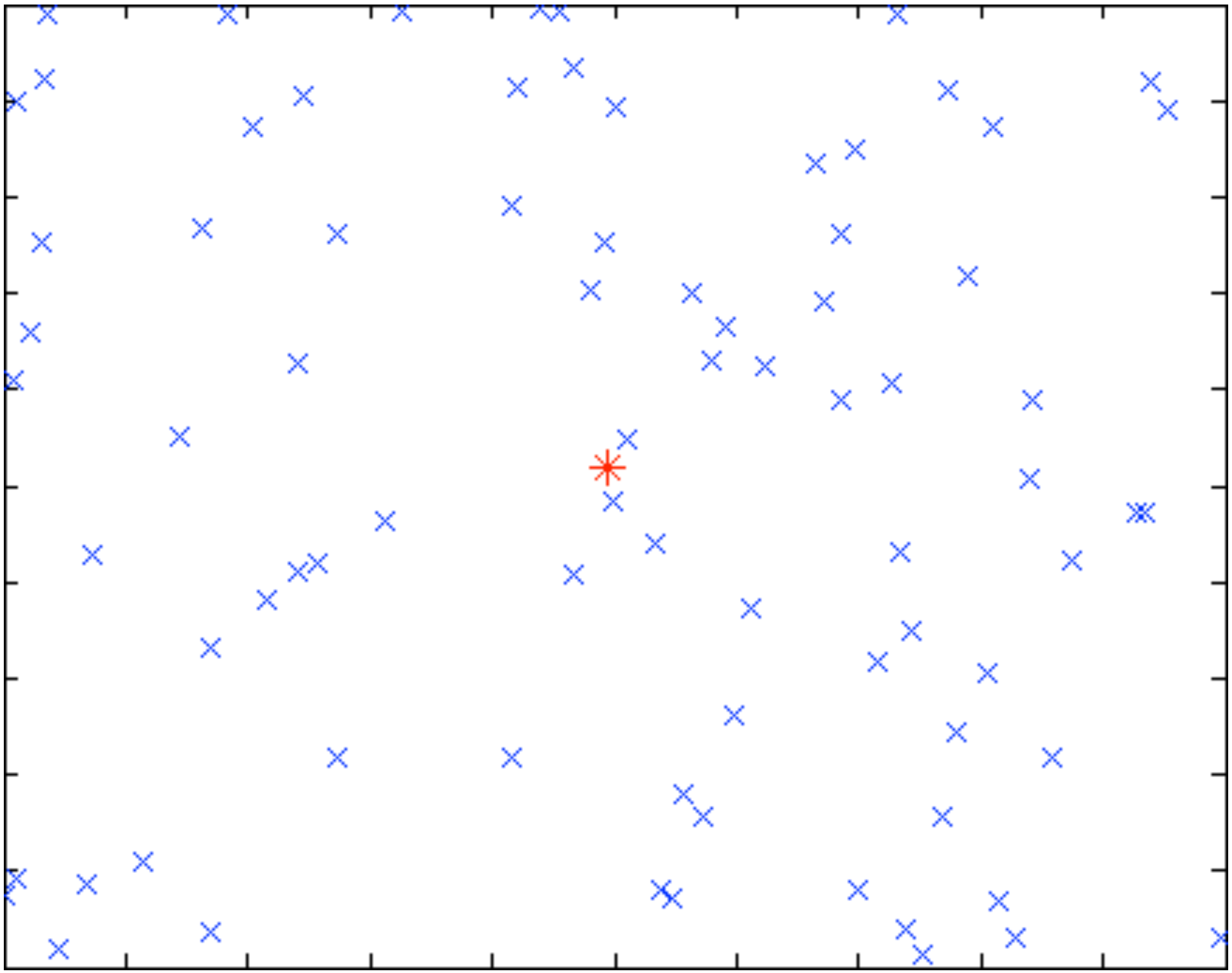
We consider the second type of problems and more specifically we are interested in the influence of the number of nodes on the performance for the various types of network topologies.

Network topologies

We consider here network topologies coming from wireless sensor networks applications, namely the geometric graphs

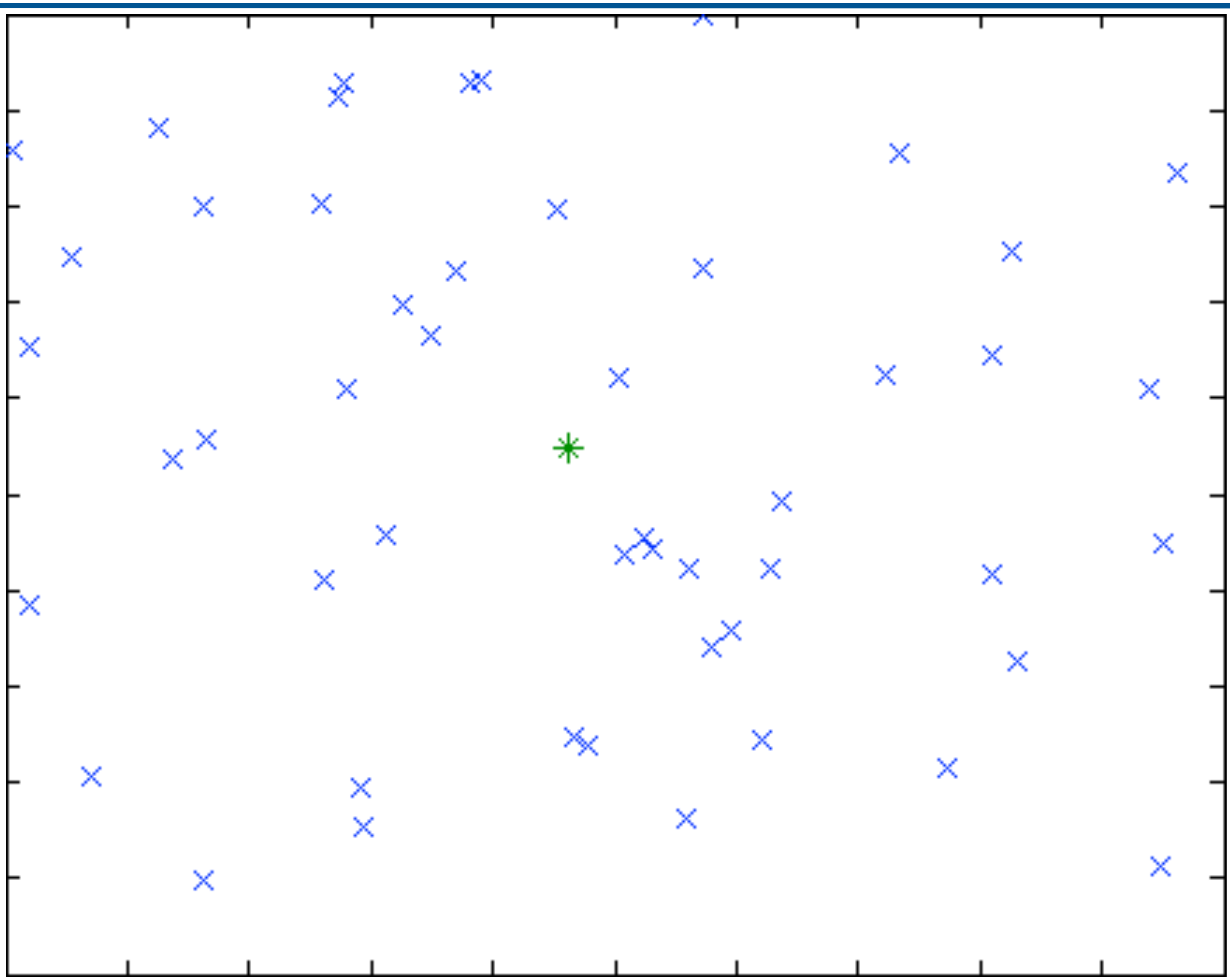


Rate of convergence



Geometric graph

Rate of convergence



Random graph

Performance of randomized consensus algorithms

We now consider randomly time varying stochastic matrices $P(t)$. We obtain the system

$$x(t+1) = P(t)x(t)$$

PROBABILISTIC CONSENSUS

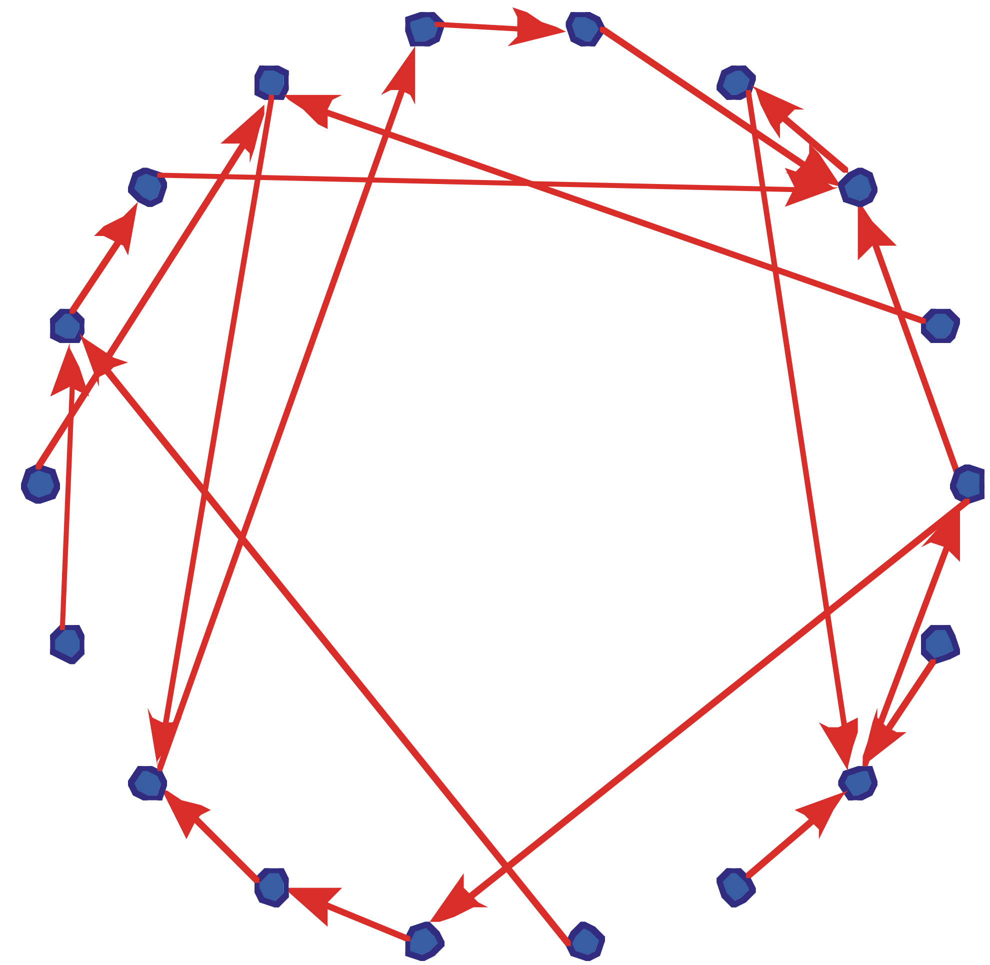
$$x_i(t) \to c \qquad \text{almost surely}$$

where $c = \sum \mu_i x_i(0)$ and where μ_i are random variables.

Performance of randomized consensus algorithms

We now consider randomly time varying stochastic matrices $P(t)$. We obtain the system

$$x(t+1) = P(t)x(t)$$



PROBABILISTIC CONSENSUS

$$x_i(t) \to c \qquad \text{almost surely}$$

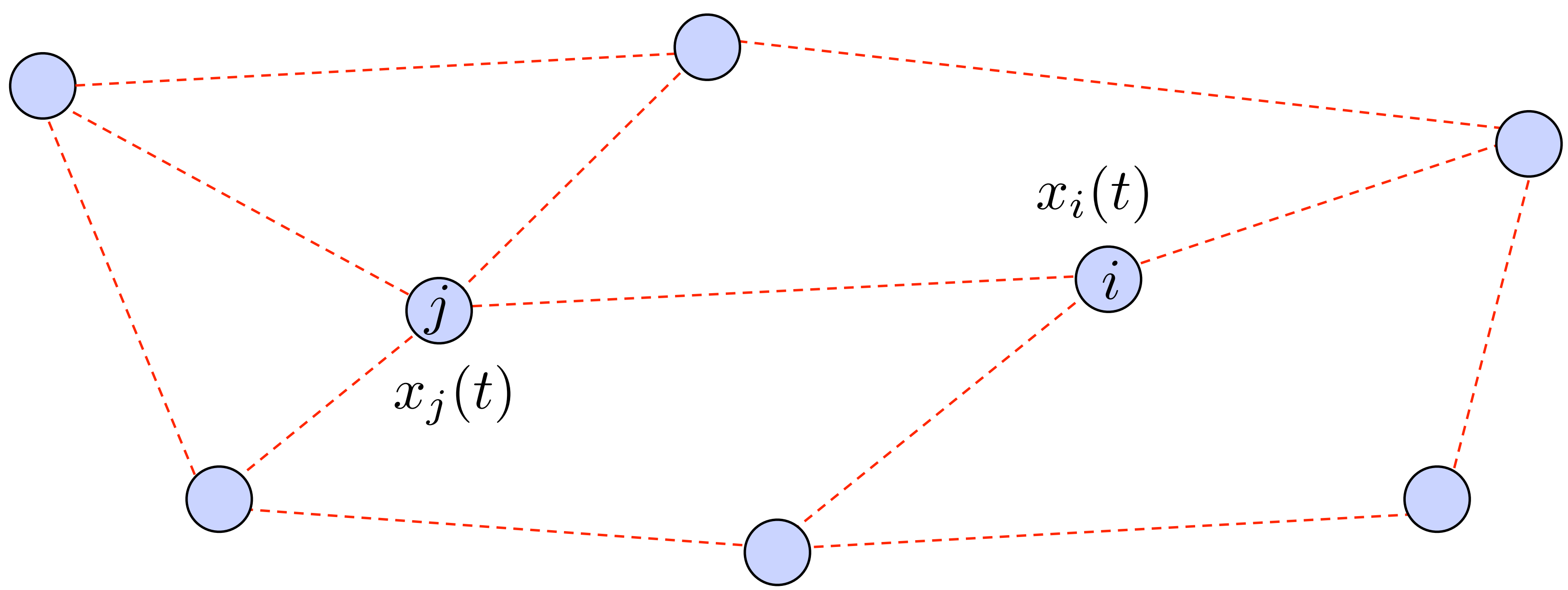
where $c = \sum \mu_i x_i(0)$ and where μ_i are random variables.

Example: gossip algorithm

Consider an indirected strongly connected graph $\mathcal{G}$.

At each time step, an edge (j, i) is chosen randomly among the edges of $\mathcal{G}$ and the following iteration is done

$$\begin{aligned} x_i(t+1) &= 1/2x_i(t) + 1/2x_j(t) \\ x_j(t+1) &= 1/2x_i(t) + 1/2x_j(t) \\ x_h(t+1) &= x_h(t) \qquad h \neq i, j \end{aligned}$$

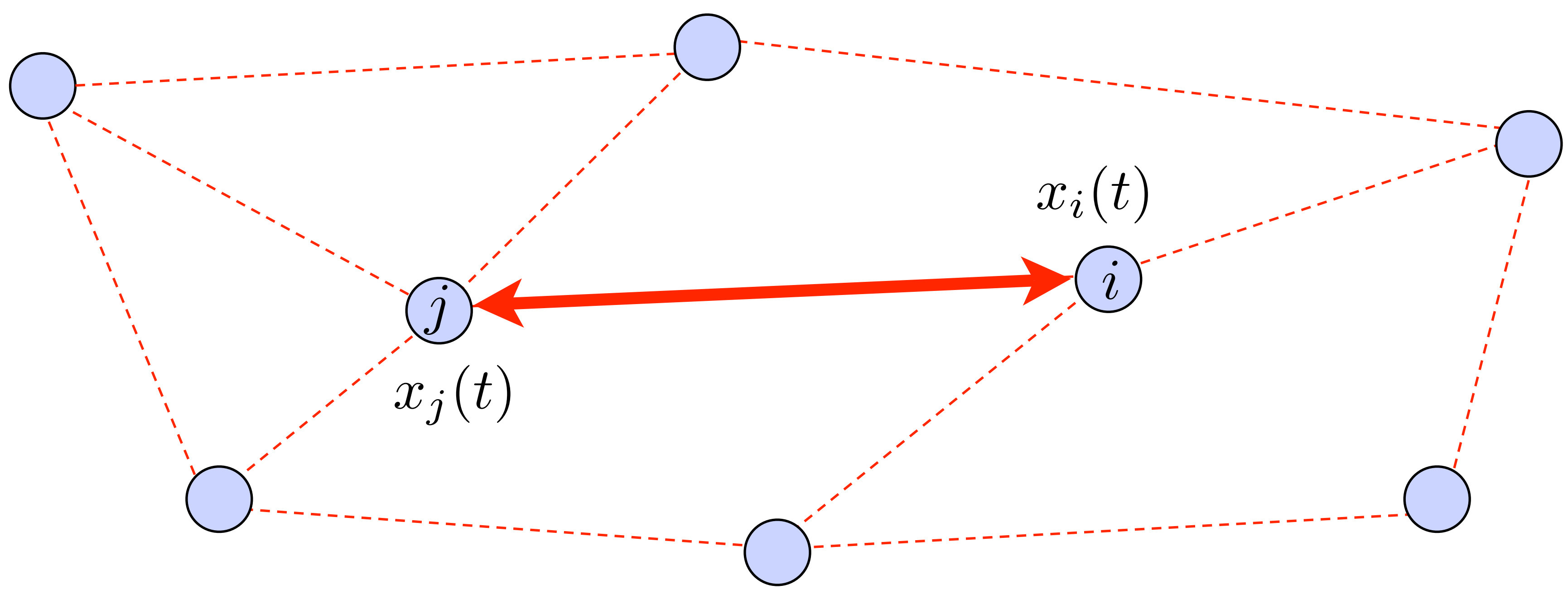


Example: gossip algorithm

Consider an indirected strongly connected graph $\mathcal{G}$.

At each time step, an edge (j, i) is chosen randomly among the edges of $\mathcal{G}$ and the following iteration is done

$$\begin{aligned} x_i(t+1) &= 1/2x_i(t) + 1/2x_j(t) \\ x_j(t+1) &= 1/2x_i(t) + 1/2x_j(t) \\ x_h(t+1) &= x_h(t) \qquad h \neq i, j \end{aligned}$$

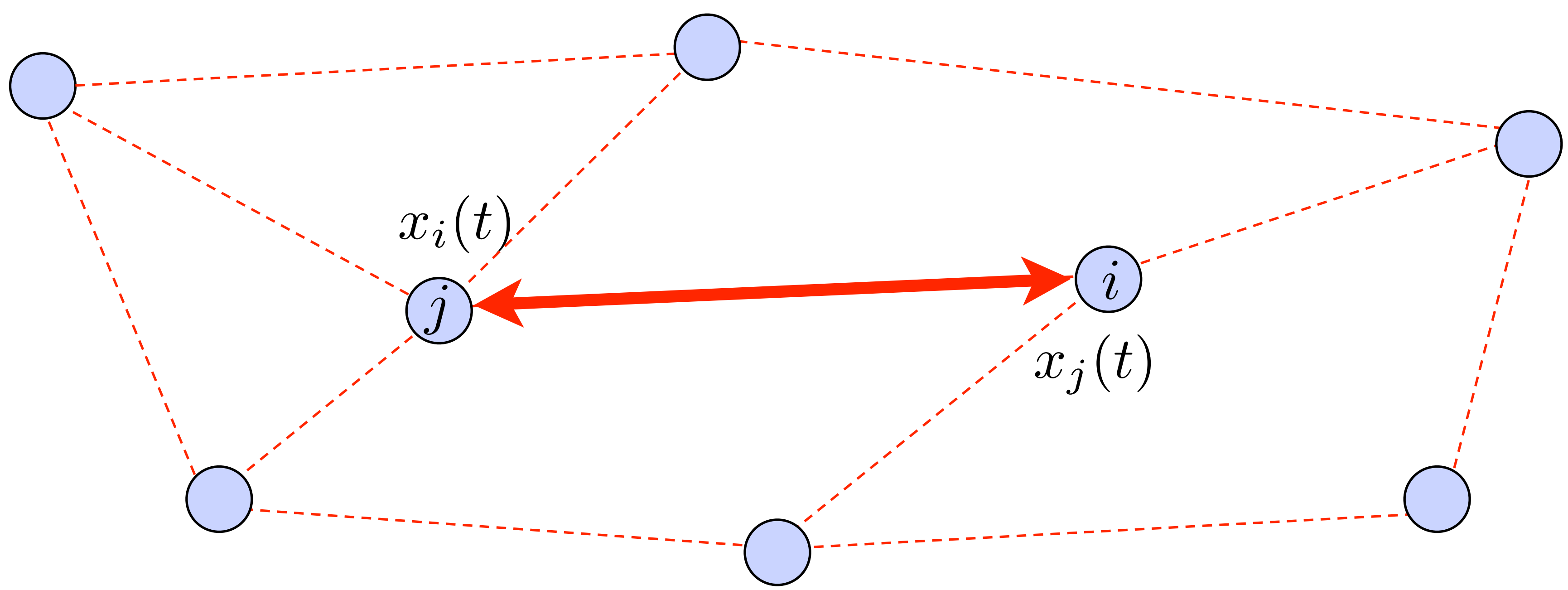


Example: gossip algorithm

Consider an indirected strongly connected graph $\mathcal{G}$.

At each time step, an edge (j, i) is chosen randomly among the edges of $\mathcal{G}$ and the following iteration is done

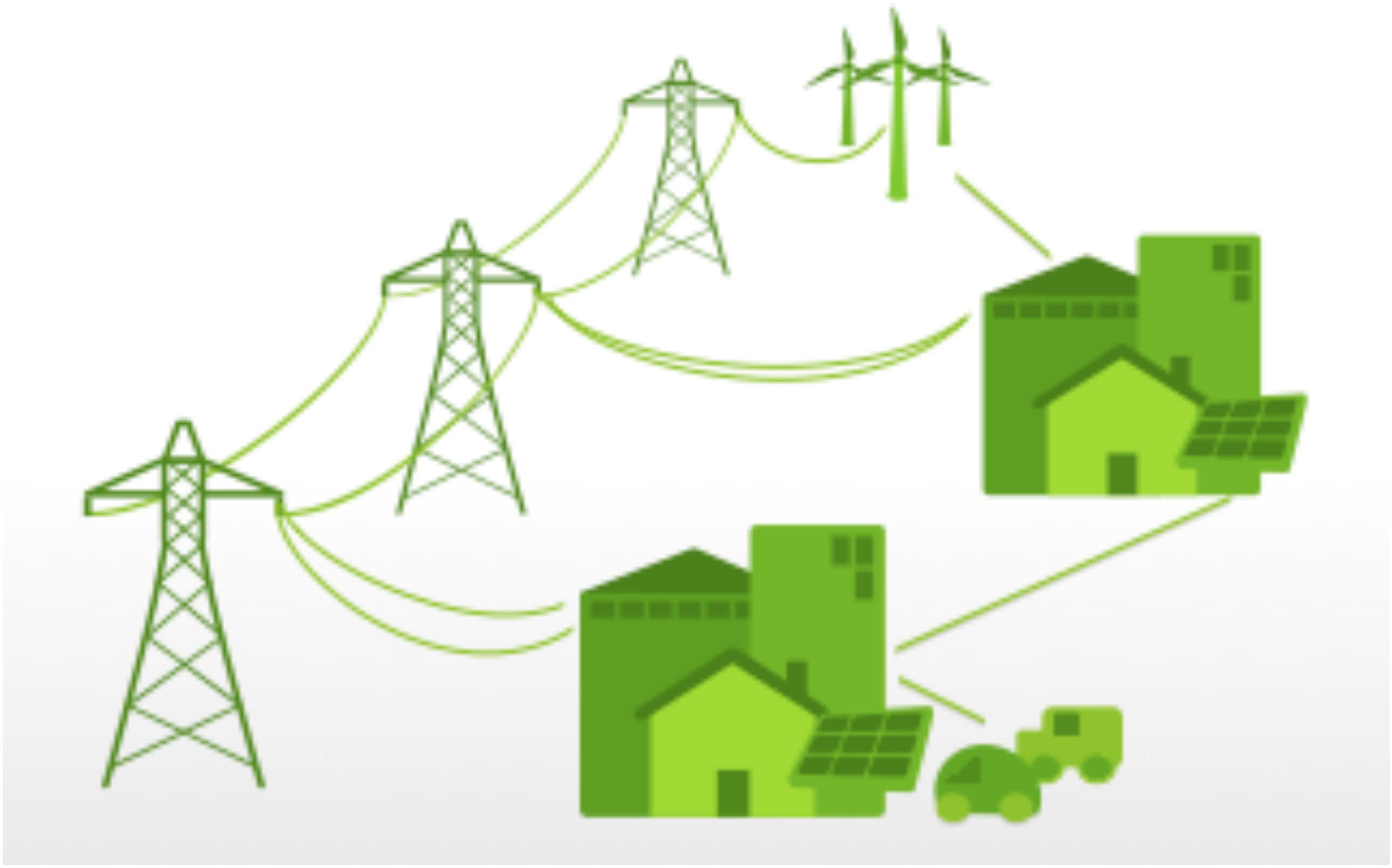
$$
\begin{aligned}
x_i(t+1) &= 1/2x_i(t) + 1/2x_j(t) \\
x_j(t+1) &= 1/2x_i(t) + 1/2x_j(t) \\
x_h(t+1) &= x_h(t) \qquad h \neq i, j
\end{aligned}
$$



Gossip algorithm (Boyd et al. 2006)

$$
P(t) = \begin{array}{cc}
\begin{array}{c} \\ \\ \\ i \\ \\ \\ \\ j \\ \\ \\ \\ \end{array} &
\begin{bmatrix}
1 & & & & & & & & & & \\
 & \ddots & & & & & & & & & \\
 & & 1 & & & & & & & & \\
 & & & \boxed{1/2} & & & & \boxed{1/2} & & & \\
 & & & & 1 & & & & & & \\
 & & & & & \ddots & & & & & \\
 & & & & & & 1 & & & & \\
 & & & \boxed{1/2} & & & & \boxed{1/2} & & & \\
 & & & & & & & & 1 & & \\
 & & & & & & & & & \ddots & \\
 & & & & & & & & & & 1
\end{bmatrix} \\
 & \begin{array}{ccccccccccc} & & & i & & & & j & & & \end{array}
\end{array}
$$

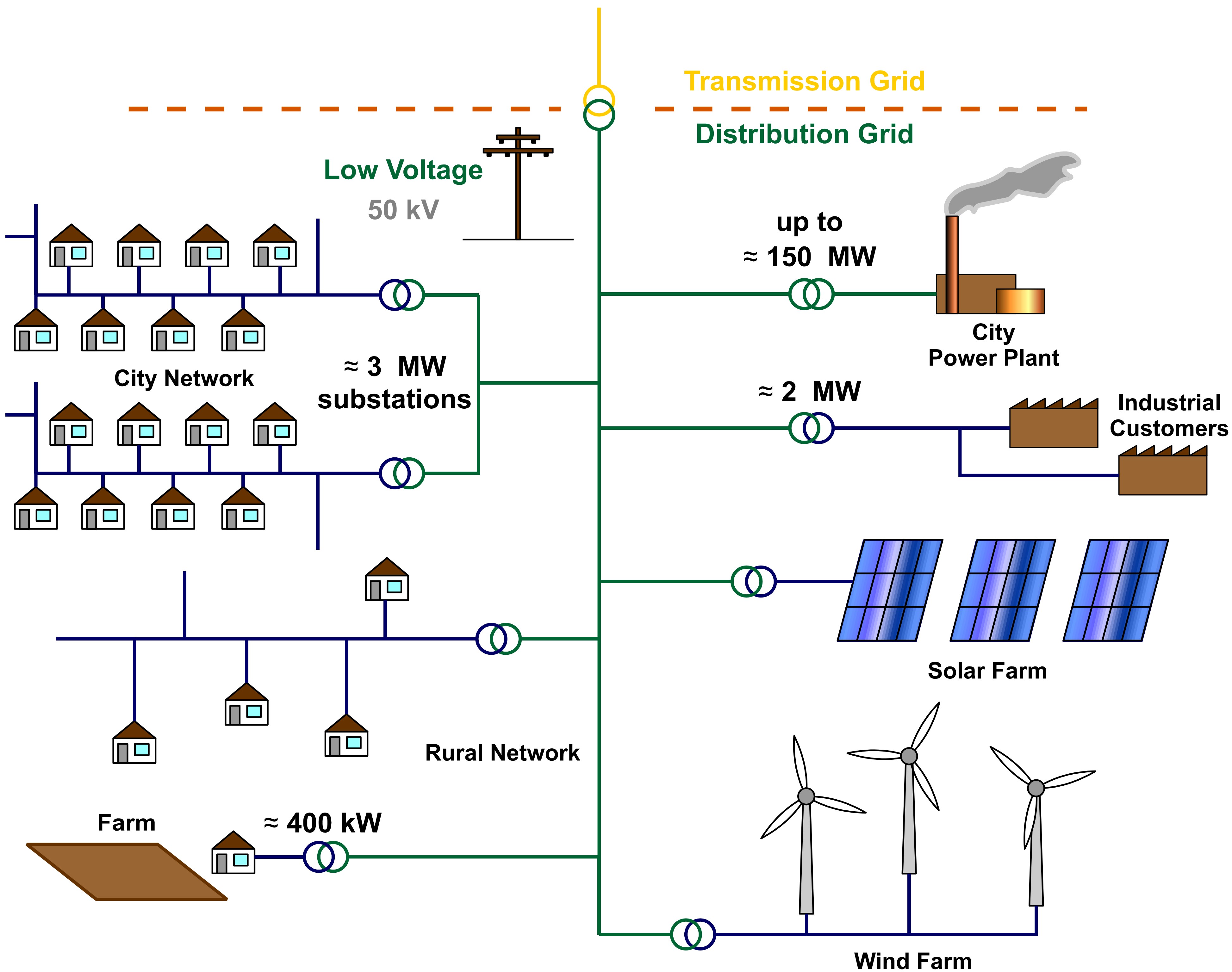
Smart grids



Smart grids

Examples of a leaderless management of an electric grid

Power distribution network



Microgrid

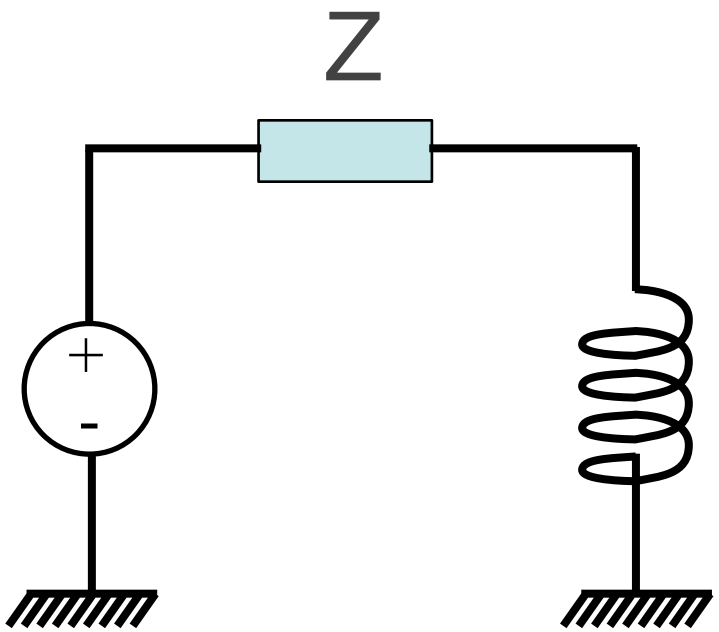
Definition of a microgrid

We define a smart microgrid as a portion of the electrical power distribution network that connects to the transmission grid (utility) in one point and that is managed autonomously from the rest of the network.

In particular, ancillary services are taken care by some microgrid controllers, whose objective is to operate the microgrid in an optimal way while satisfying some constraint on how the microgrid interfaces with the rest of the network.

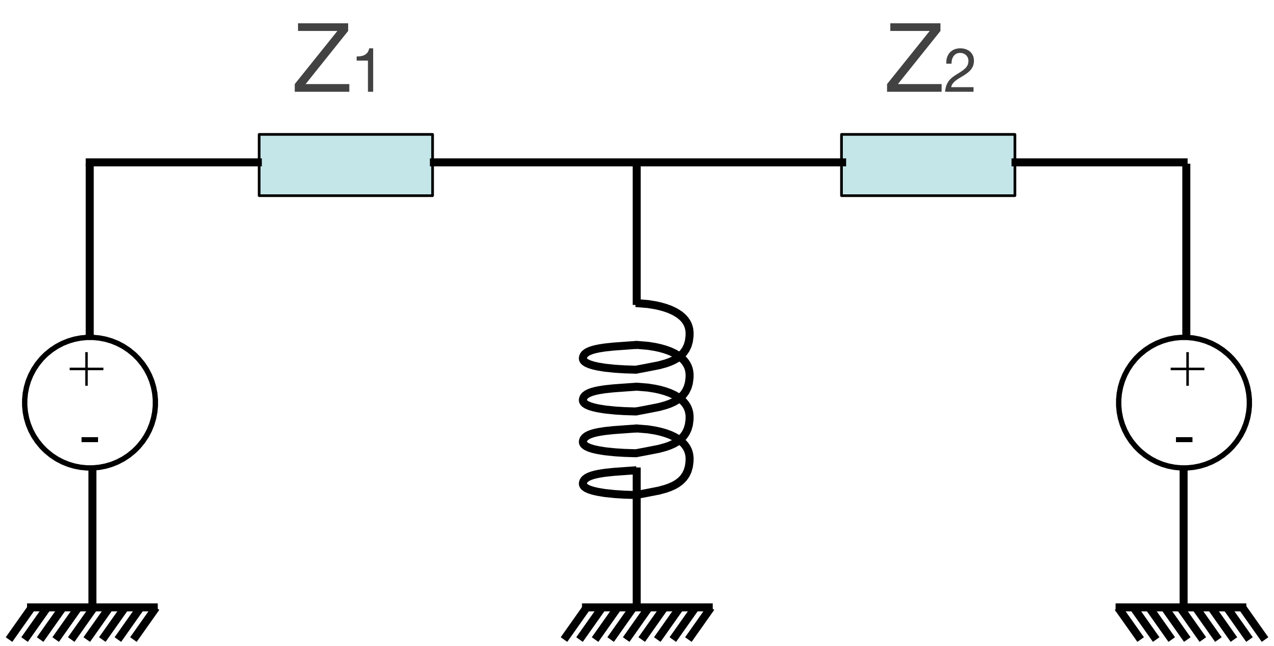
Among them, we focus on the problem of optimal reactive power compensation.

Reactive power



Lossless electric components need current but they do not need electric power.

However, in order to bring this current to these components some electric power is dissipated along the transmission lines.



It is convenient that the current is provided by the generator which is closed, namely by the generator which is connected to the lossless component by a line with smaller resistance.

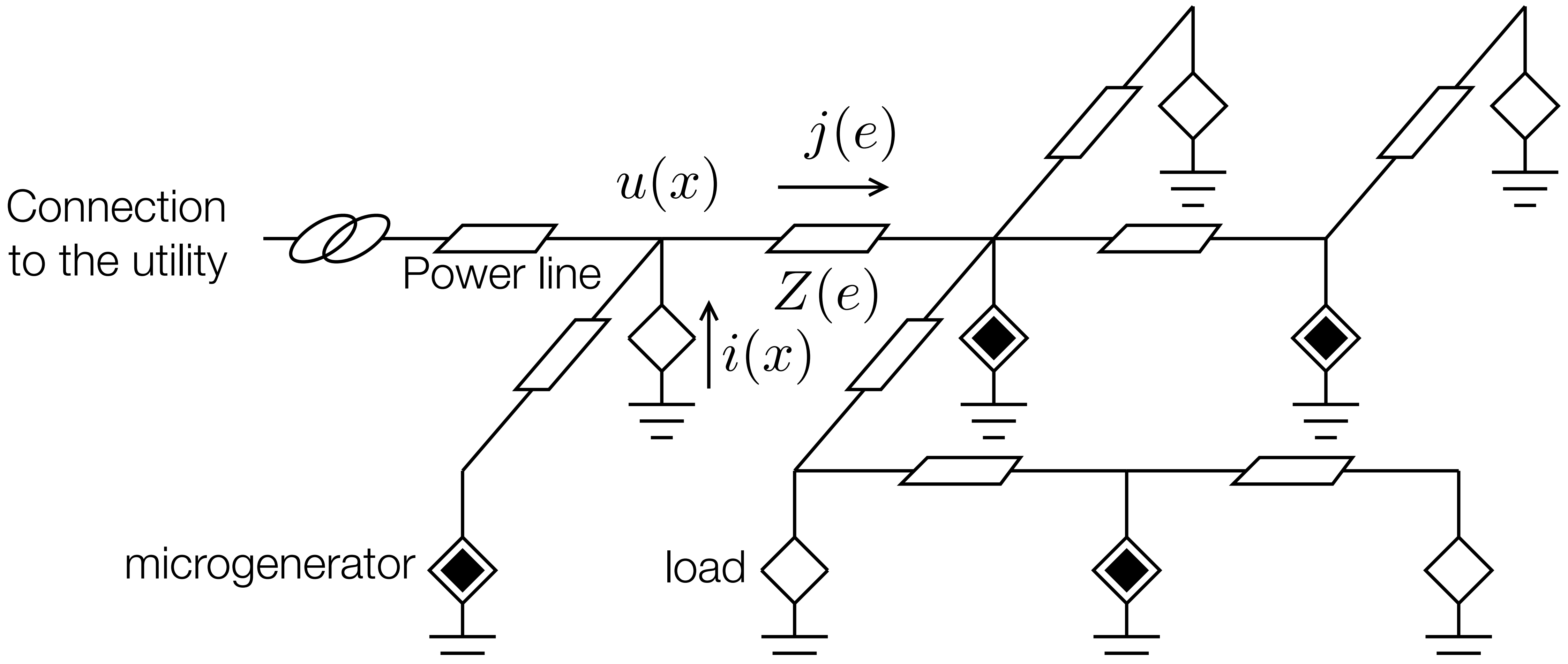
Reactive power

Definition of the reactive power

We have reactive power whenever voltage and current are out of phase, i.e phase angle is not zero.

- Users in the microgrid may require reactive power
- it can be obtained from the utility or produced by the electronic interfaces of microgenerators in the grid
- the utility charges the microgrid for reactive power
- producing reactive power has no fuel cost
- larger flows of reactive power correspond to quadratically larger power losses on the cables.

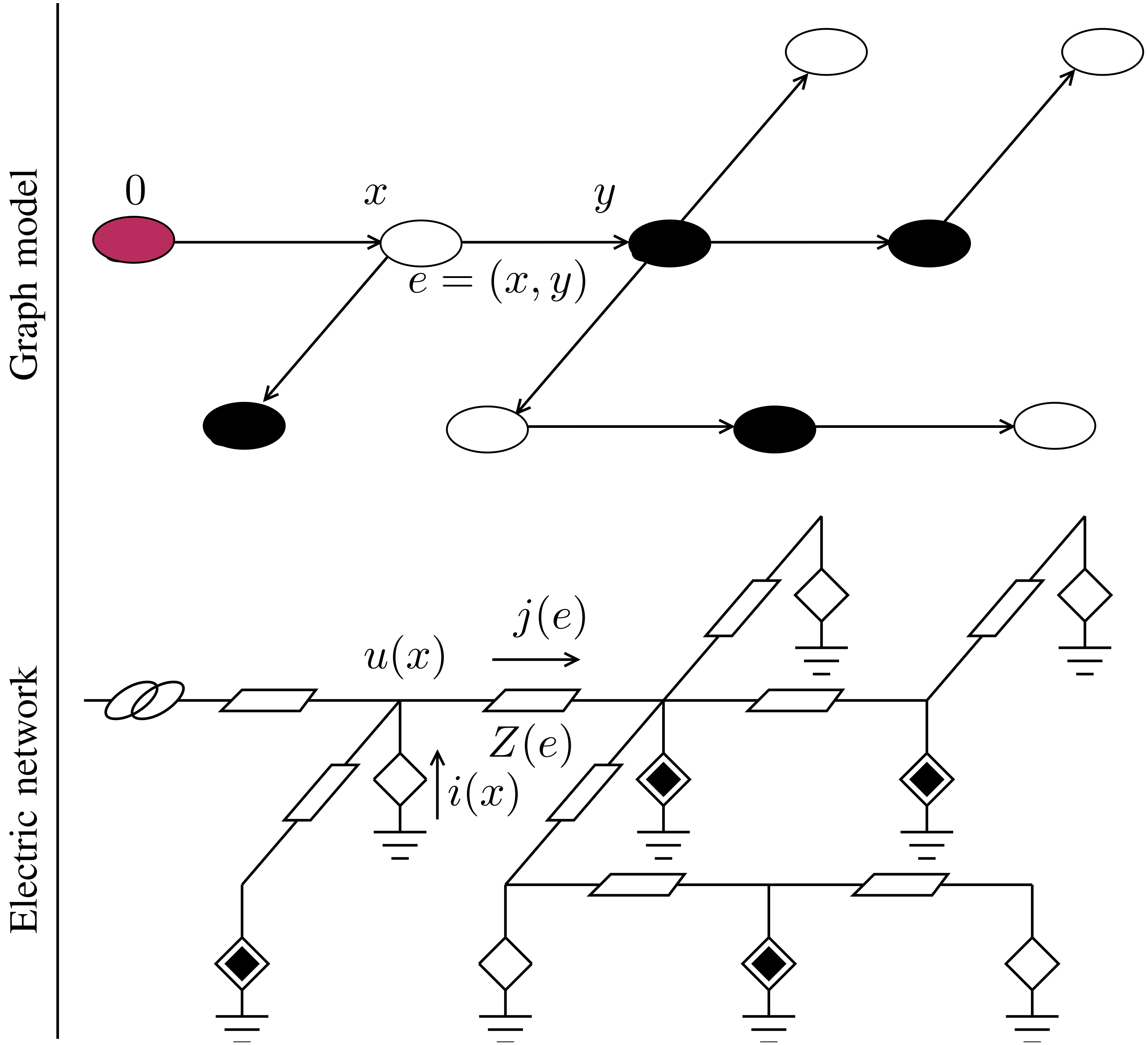
A model of a microgrid



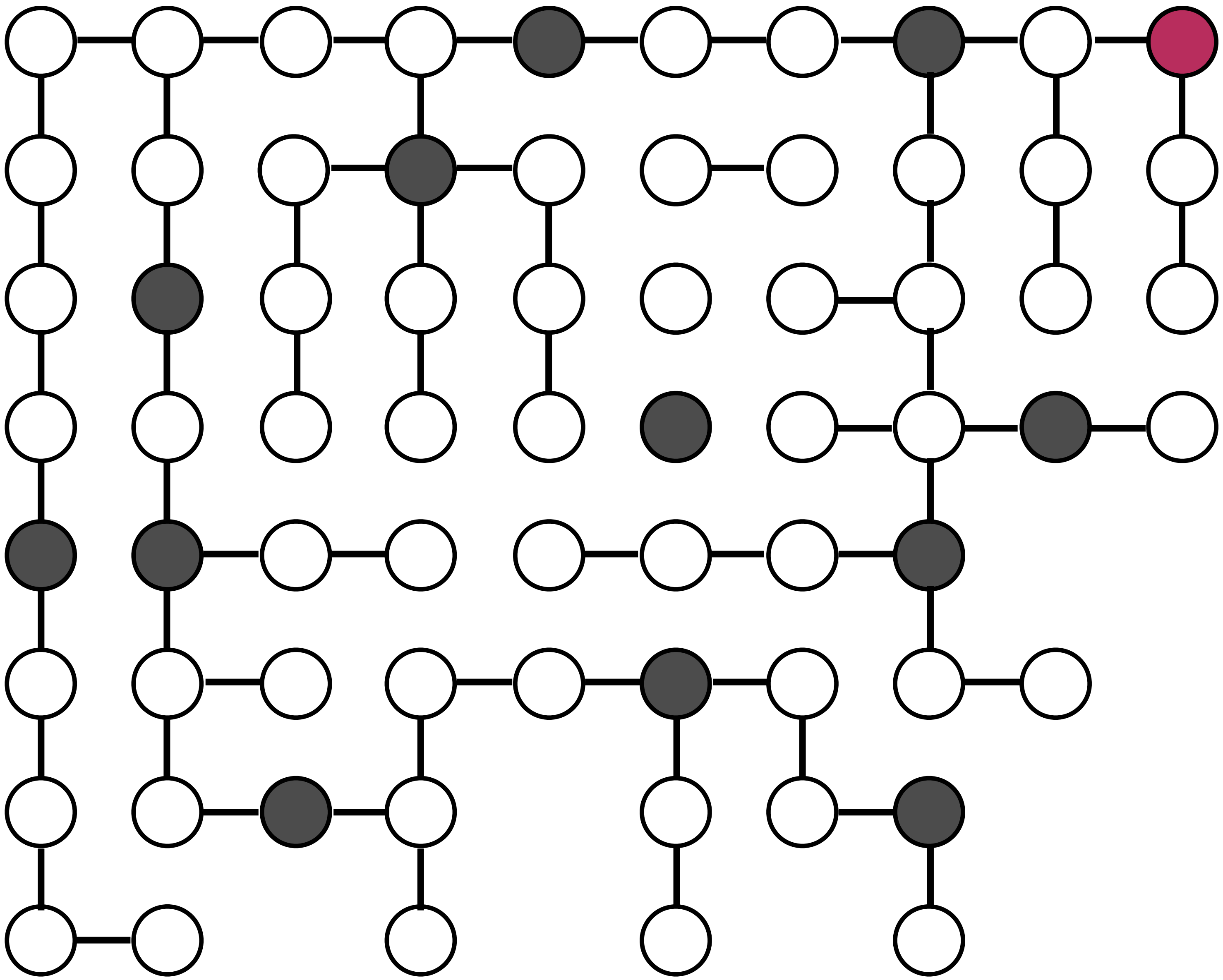
Microgrid: a portion of the power distribution network

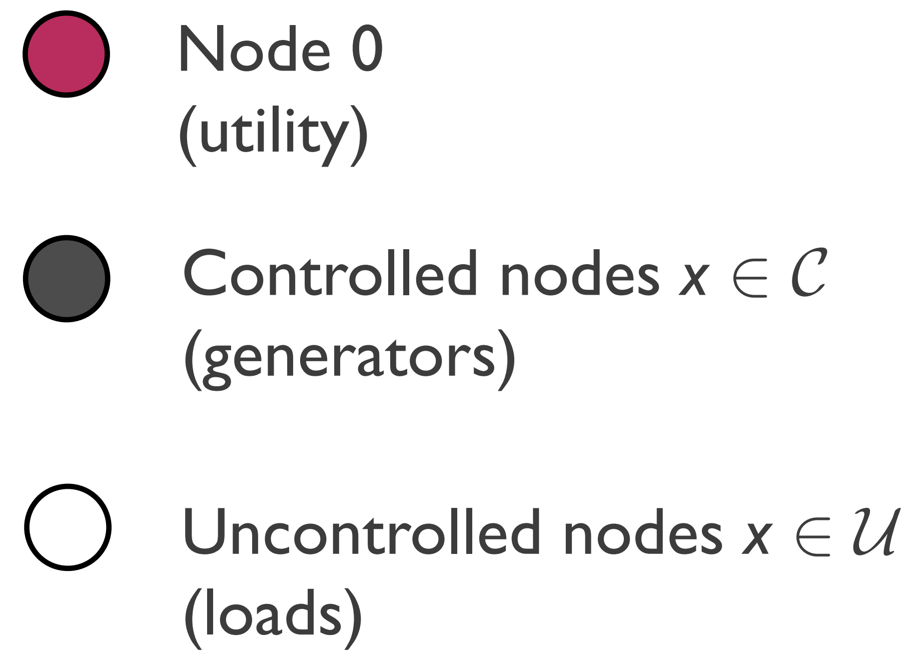
- connected to the utility
- populated by loads ◊ and microgenerators ⧫

A model of a microgrid

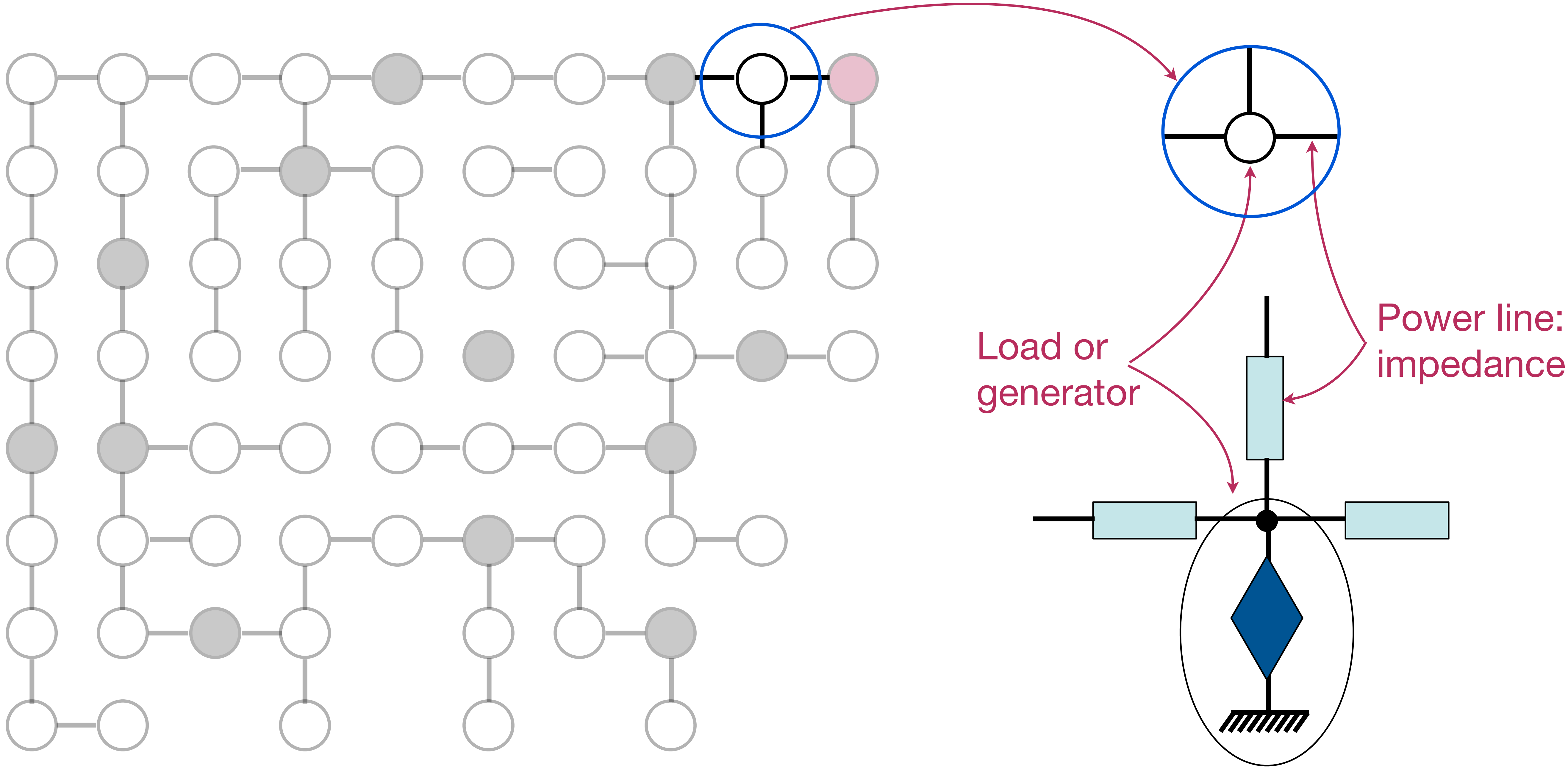


A model of a microgrid





A model of a microgrid



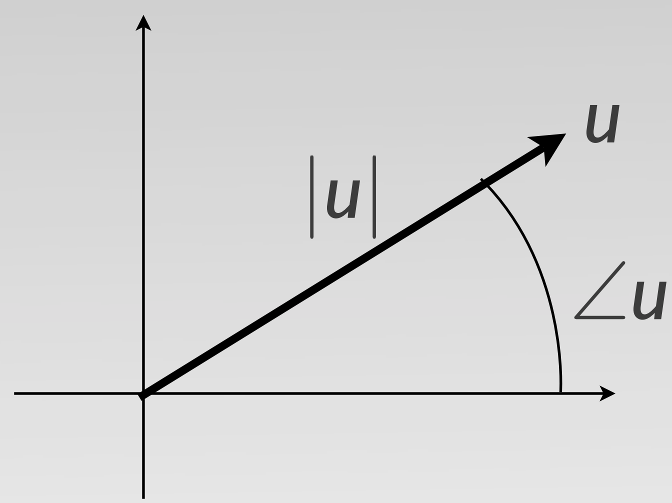
A model of a microgrid

Sinusoidal regime

We assume that the circuit is at the sinusoidal regime at a certain fixed frequency. In this way every signal $u(t)$ are described by a complex number $u \in \mathbb{C}$ describing amplitudes and phases

$$u(t) = |u| \cos(\omega t + \angle u)$$

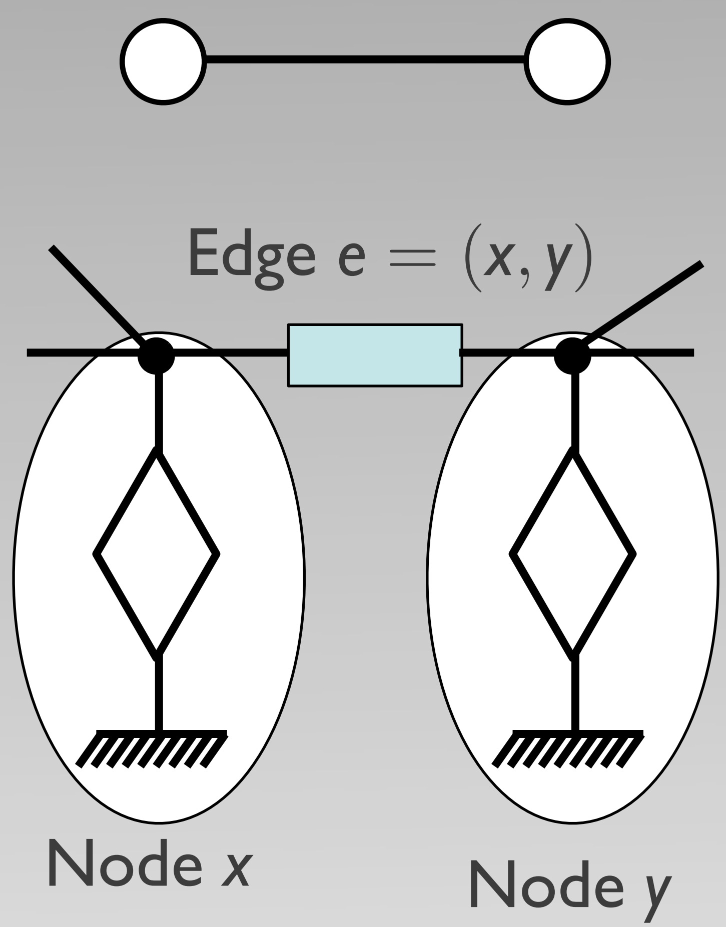
where $|u|$ is the absolute value of u and $\angle u$ is the phase of u.

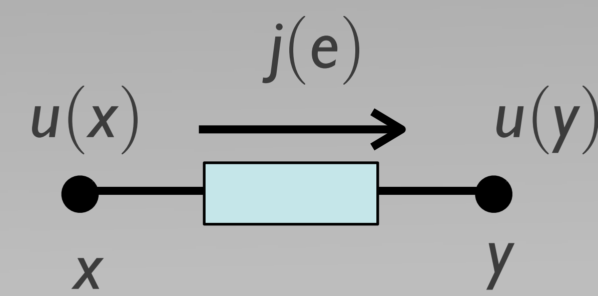


$$u = |u| e^{j\angle u}$$

A model of a microgrid

Model of power lines





$$\boxed{u(x) - u(y) = Z(e)j(e)}$$

$u(x)$ potential at node x

$u(y)$ potential at node y

$j(e)$ current at edge $e = (x, y)$

$Z(e)$ impendence at edge $e = (x, y)$

A model of a microgrid

Let u be the vectors with components $u(x)$ and j be the vector with components $j(e)$.

Let A be the incidence matrix of the graph with a row per edge, a column per node and entries $0, \pm 1$.

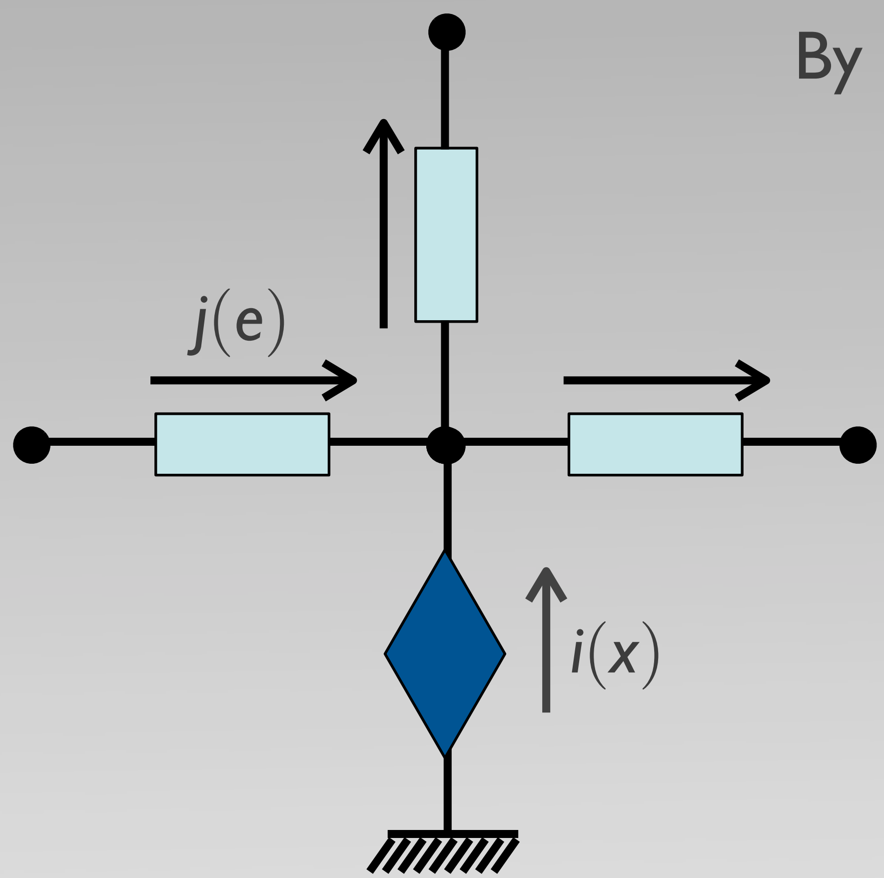
$$A = \begin{array}{cc} \begin{bmatrix} & & & & \\ & -1 & & 1 & \\ & & & & \end{bmatrix} & \text{edge } e \\ \begin{matrix} \quad & \text{node } x & \quad & \text{node } y & \quad \end{matrix} & \end{array} \qquad (Au)_e = u(x) - u(y)$$

Let Z be the diagonal matrix with diagonal entries $Z(e)$.

$$Z = \begin{bmatrix} \ddots & & \\ & Z(e) & \\ & & \ddots \end{bmatrix} \qquad \boxed{Au + Zj = 0}$$

A model of a microgrid

Kirchhoff's current law

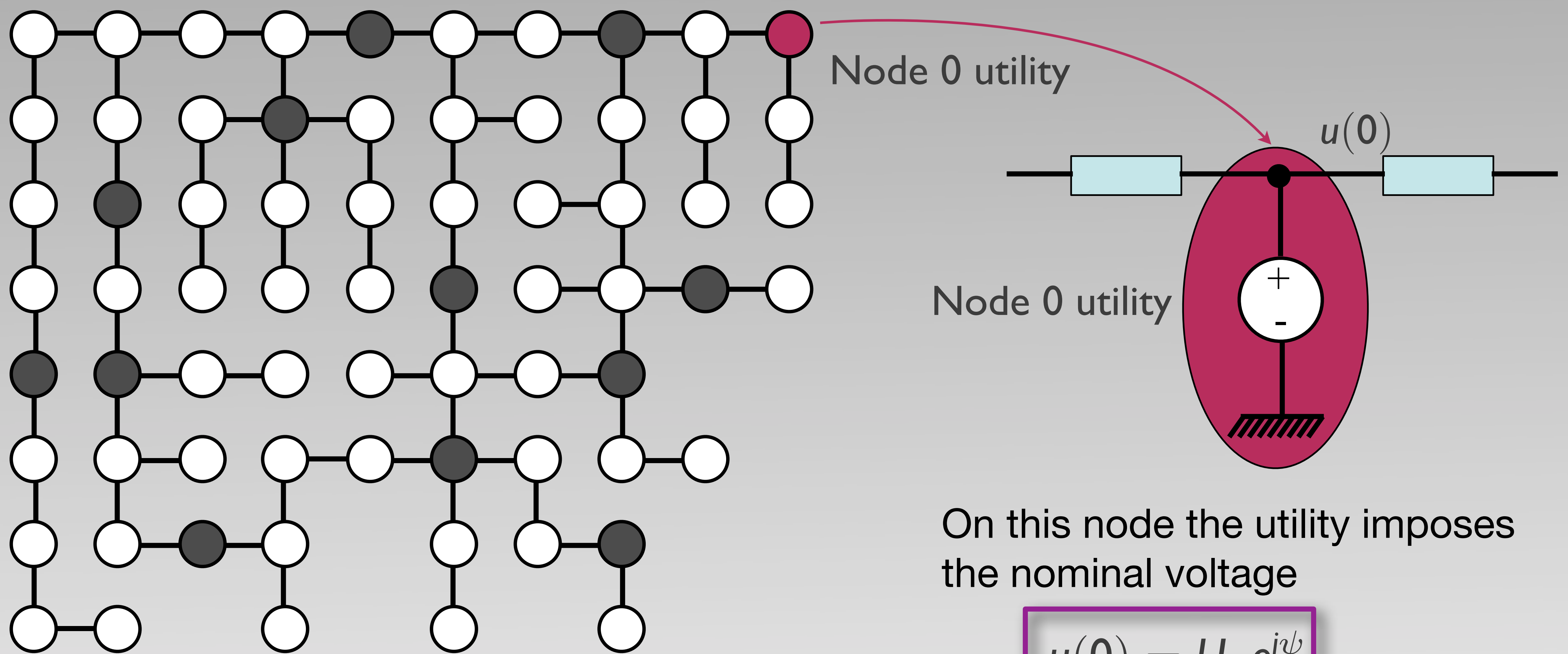


For the Kirchhoff's current law the sum of the currents is zero.
By using the incidence matrix A we get

$$\boxed{A^T j + i = 0}$$

A model of a microgrid

Model of the nodes



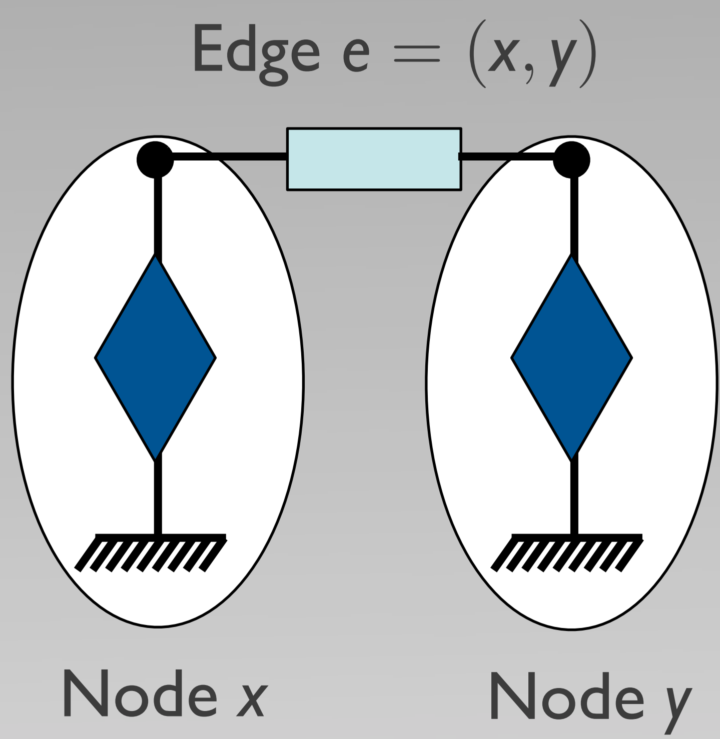
On this node the utility imposes the nominal voltage

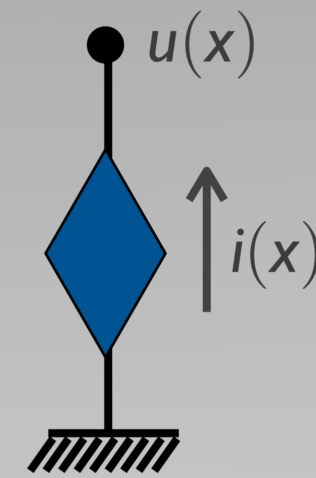
$$\boxed{u(0) = U_N e^{j\psi}}$$

U_N is the nominal voltage

A model of a microgrid

Model of the load/generator nodes





Fixed power law $\boxed{u(x)i(x)^* = s(x)}$

Loads/generators impose a given amount of active power and reactive power

$u(x)$ voltage at the node x

$i(x)$ current at the load/generator x

$s(x)$ complex power imposed by the load/generator x

Non-linear constraint

A model of a microgrid

Controlled and uncontrolled (disturbance) variables

In the uncontrolled nodes (loads) $x \in \mathcal{U}$ the complex power is not controllable.

In the controlled nodes (generators) $x \in \mathcal{C}$ the active power

$$p(x) := \mathrm{Re}[s(x)]$$

is typically the maximum that can be generated, while

$$q(x) := \mathrm{Im}[s(x)]$$

can be decided. Therefore $q(x)$, $x \in \mathcal{C}$ are the control variables, while $p(x)$, $x \in \mathcal{C}$ and $s(x)$, $x \in \mathcal{U}$ can be considered as the disturbance variables.

A model of a microgrid

Equations describing the model

- Let u, i, s, j be the vectors of components $u(x), i(x), s(x), j(e)$ respectively.
- Let A be the incidence matrix of the graph.
- Let Z be the diagonal matrix with diagonal entries $Z(e)$.

$$\begin{cases} \left.\begin{array}{l} A^T j + i = 0 \\ Au + Zj = 0 \\ u(0) = U_N e^{j\psi} \end{array}\right\} \text{Linear constraints} \\ u(x) i(x)^* = s(x) \text{ for all } x \neq 0 \Big\} \text{Nonlinear constraints} \end{cases}$$

$$F(u, i, j, s, U_N) = 0$$

U_N (nominal voltage)
$s(x)$, $x \neq 0$ (complex powers)

Determine $\Rightarrow$ u, i, j

A model of a microgrid

Equations describing the model

- Let u, i, s, j be the vectors of components $u(x), i(x), s(x), j(e)$ respectively.
- Let A be the incidence matrix of the graph.
- Let Z be the diagonal matrix with diagonal entries $Z(e)$.

$$\begin{cases} \left.\begin{array}{l} A^T j + i = 0 \\ Au + Zj = 0 \\ u(0) = U_N e^{j\psi} \end{array}\right\} \text{Linear constraints} \\ u(x)i(x)^* = s(x) \text{ for all } x \neq 0 \Big\} \text{Nonlinear constraints} \end{cases}$$

$$F(u, i, j, s, U_N) = 0$$

inputs $\begin{cases} U_N \text{ (nominal voltage)} \\ s(x),\ x \neq 0 \text{ (complex powers)} \end{cases}$ Determine u, i, j

A model of a microgrid

Equations describing the model

- Let u, i, s, j be the vectors of components $u(x), i(x), s(x), j(e)$ respectively.
- Let A be the incidence matrix of the graph.
- Let Z be the diagonal matrix with diagonal entries $Z(e)$.

$$\begin{cases} \left.\begin{array}{l} A^T j + i = 0 \\ Au + Zj = 0 \\ u(0) = U_N e^{j\psi} \end{array}\right\} \text{Linear constraints} \\ u(x) i(x)^* = s(x) \text{ for all } x \neq 0 \Big\} \text{Nonlinear constraints} \end{cases}$$

Implicit function

$$F(u, i, j, s, U_N) = 0$$

states inputs

$$\text{inputs} \begin{cases} U_N \text{ (nominal voltage)} \\ s(x),\ x \neq 0 \text{ (complex powers)} \end{cases}$$

Determine

$$\left. u, i, j \ \right\} \text{states}$$

A model of a microgrid

Equations describing the model

- Let u, i, s, j be the vectors of components $u(x), i(x), s(x), j(e)$ respectively.
- Let A be the incidence matrix of the graph.
- Let Z be the diagonal matrix with diagonal entries $Z(e)$.

$$\begin{cases} \left.\begin{aligned} & A^T j + i = 0 \\ & Au + Zj = 0 \\ & u(0) = U_N e^{j\psi} \end{aligned}\right\} \text{Linear constraints} \\ u(x)i(x)^* = s(x) \text{ for all } x \neq 0 \,\} \text{ Nonlinear constraints} \end{cases}$$

Implicit function

$$F(\underbrace{u, i, j}_{\text{states}}, s, U_N) = 0$$

inputs

$$\text{inputs}\begin{cases} U_N \text{ (nominal voltage)} \\ s(x),\ x \neq 0 \text{ (complex powers)} \end{cases} \xrightarrow{\text{Determine}} \left. u, i, j \right\} \text{states}$$

A model of a microgrid

Equations describing the model

- Let u, i, s, j be the vectors of components $u(x), i(x), s(x), j(e)$ respectively.
- Let A be the incidence matrix of the graph.
- Let Z be the diagonal matrix with diagonal entries $Z(e)$.

$$
\begin{cases}
\left.\begin{array}{l} A^T j + i = 0 \\ Au + Zj = 0 \\ u(0) = U_N e^{j\psi} \end{array}\right\} \text{Linear constraints} \\
u(x) i(x)^* = s(x) \text{ for all } x \neq 0 \Big\} \text{Nonlinear constraints}
\end{cases}
$$

Implicit function

$$F(\underbrace{u, i, j}_{\text{states}}, s, U_N) = 0$$

inputs

$$\text{inputs} \begin{cases} U_N \text{ (nominal voltage)} \\ s(x),\ x \neq 0 \text{ (complex powers)} \end{cases}$$

Determine $\Rightarrow$

$$u, i, j \Big\} \text{ states}$$

A model of a microgrid

Equations describing the model

- Let u, i, s, j be the vectors of components $u(x), i(x), s(x), j(e)$ respectively.
- Let A be the incidence matrix of the graph.
- Let Z be the diagonal matrix with diagonal entries $Z(e)$.

$$\begin{cases} \left.\begin{array}{l} A^T j + i = 0 \\ Au + Zj = 0 \\ u(0) = U_N e^{j\psi} \end{array}\right\} \text{Linear constraints} \\ u(x)i(x)^* = s(x) \text{ for all } x \neq 0 \Big\} \text{Nonlinear constraints} \end{cases}$$

Implicit function

$$F(\underbrace{u, i, j}_{\text{states}}, \underbrace{s, U_N}_{\text{inputs}}) = 0$$

inputs $\begin{cases} U_N \text{ (nominal voltage)} \\ s(x), x \neq 0 \text{ (complex powers)} \end{cases}$ → Determine → $u, i, j \Big\}$ states

The cost function

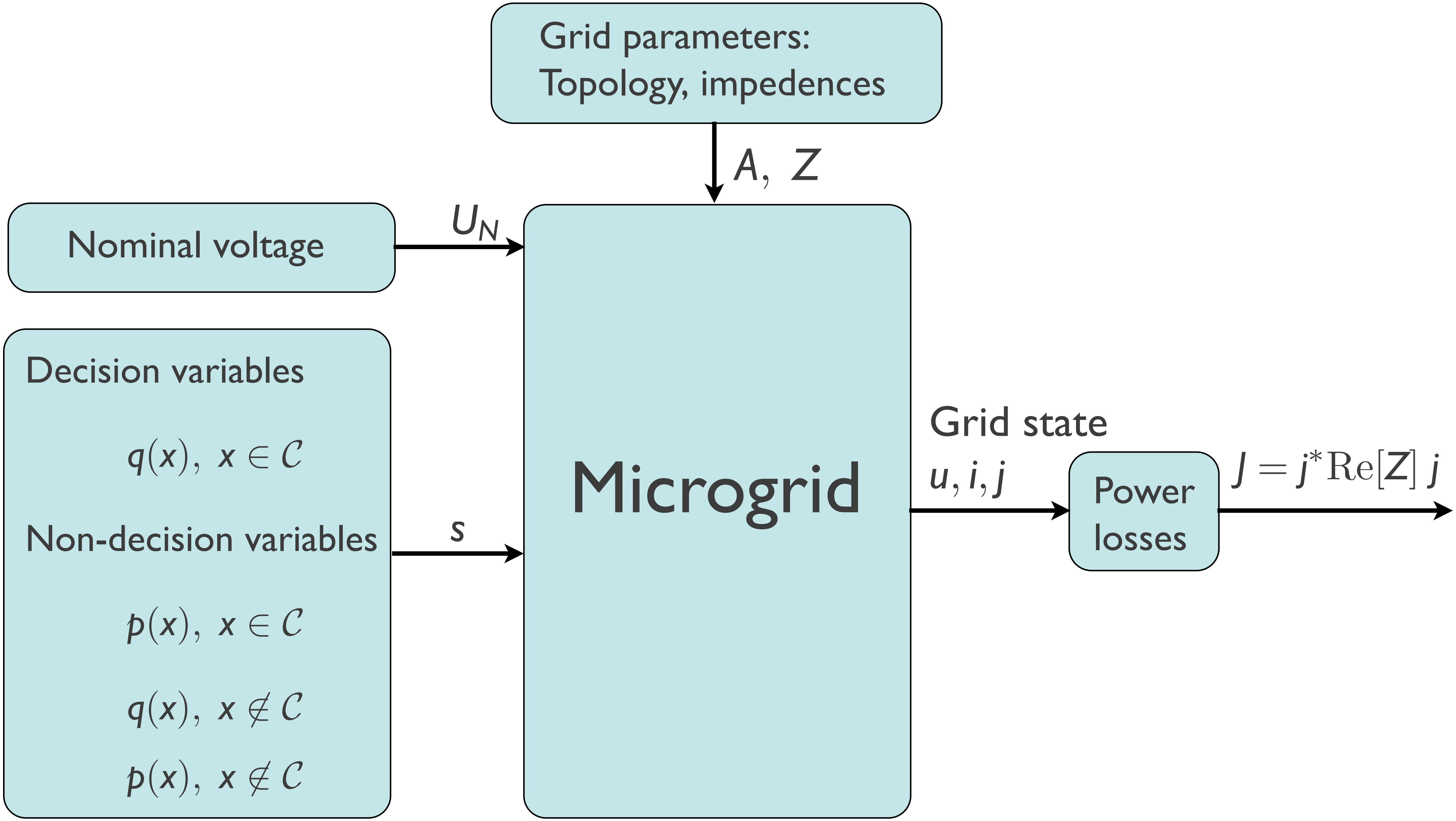
The power losses along the power lines

$$\boxed{J = \sum_{\mathrm{e}} \mathrm{Re}[Z(\mathrm{e})] |j(\mathrm{e})|^2 = j^* \mathrm{Re}[Z]\, j}$$

This is a quadratic function in j, but it is non quadratic as a function of the inputs s and U_N

$$J = J(s, U_N)$$

The grid model



THEOREM

Power series expansion

$$J(s, U_N) = J_2(s)U_N^{-2} + J_3(s)U_N^{-3} + \cdots$$

where

$$\boxed{J_2(s) = s^* M s}$$

and M is a suitable real matrix depending on the power network topology.

Proof: Implicit function theorem

Approximation of the cost

Approximated cost function

$$J(s, U_N) = J_2(s)U_N^{-2} + J_3(s)U_N^{-3} + \cdots, \qquad J_2(s) = s^* M s$$

If U_N is big, then $$\boxed{J(s, U_N) \simeq \frac{1}{U_N^2} s^* M s}$$

For fixed U_N, minimizing $J(s, U_N)$ is equivalent to minimizing

$$\boxed{J_2(s) = s^* M s}$$

Approximation of the cost

Decompose the complex vector s in its real and imaginary part

$$s = p + jq$$

where p is the active power and q is the reactive power, then

$$J_2(s) = p^T M p + q^T M q$$

⇓

Minimizing $J_2(s)$ is equivalent to minimizing $q^T M q$
(active power p is not controllable)

Quadratic optimization

Minimizing the cost $J(s, U_N)$ is equivalent to minimizing $q^T M q$.
Only the components of q belonging to the set of controllable nodes $\mathcal{C}$ are controllable.

Hence we need to solve the following optimization

Optimization problem

$$\min \quad q^T M q$$

$$\begin{aligned} &q(x) \quad x \in \mathcal{C} \text{ are free} \\ &q(x) \quad x \notin \mathcal{C} \text{ are fixed} \\ &\textstyle\sum_x q(x) = 0 \end{aligned}$$

Motivation for distributed algorithm

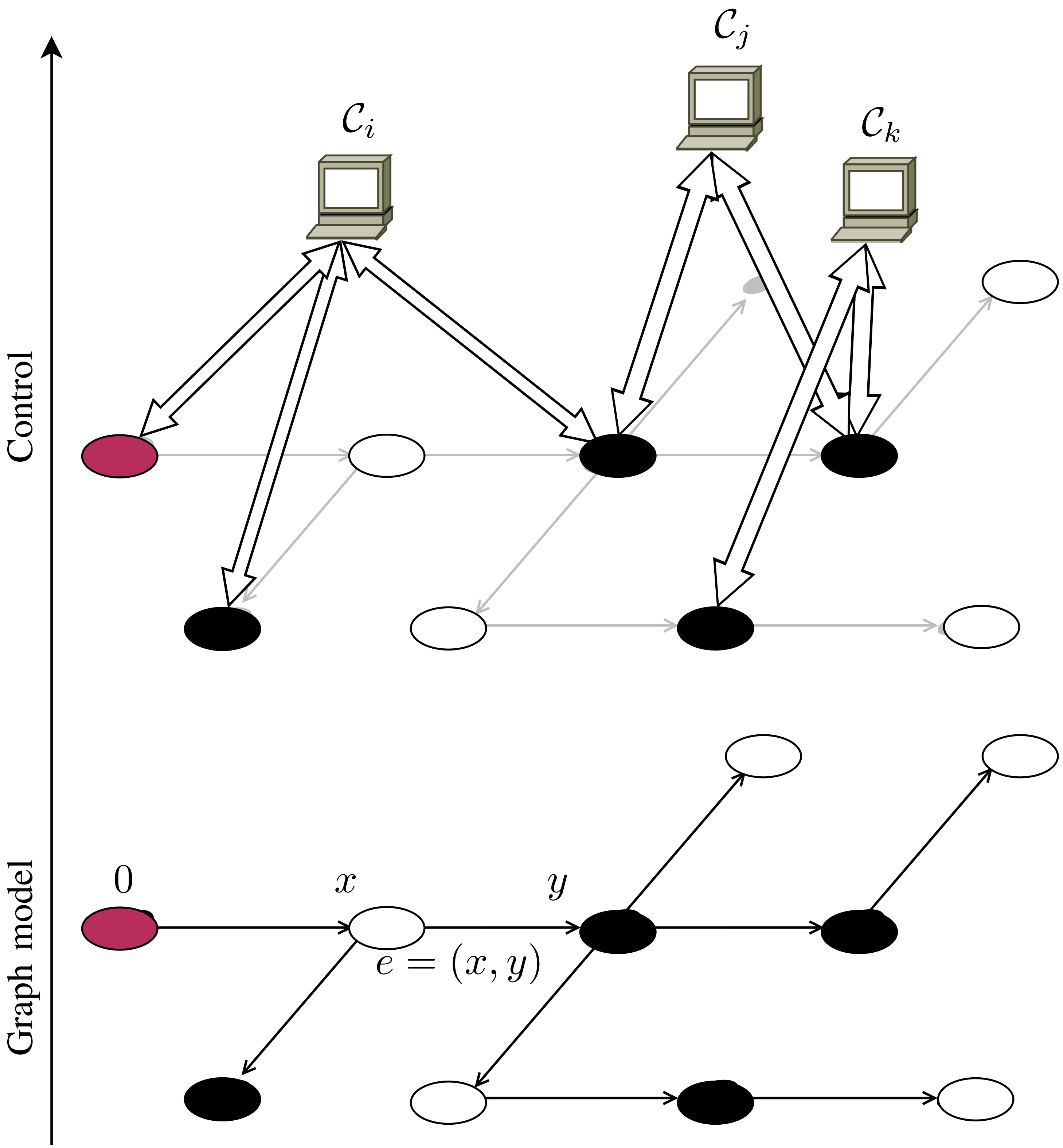
$$\min \quad q^T M q$$

$$\begin{array}{ll} q(x) & x \in \mathcal{C} \text{ are free} \\ q(x) & x \notin \mathcal{C} \text{ are fixed} \\ \sum_x q(x) = 0 & \end{array}$$

This optimization problem admits a simple closed form solution, but:

- complete knowledge of the system structure and of the system state is required
- coordination and communication among all agents is required
- compensators are
  - heterogeneous in size and characteristics
  - in large number
  - subject to partial availability, disappearance, replacement, new insertion

Distributed optimization algorithm



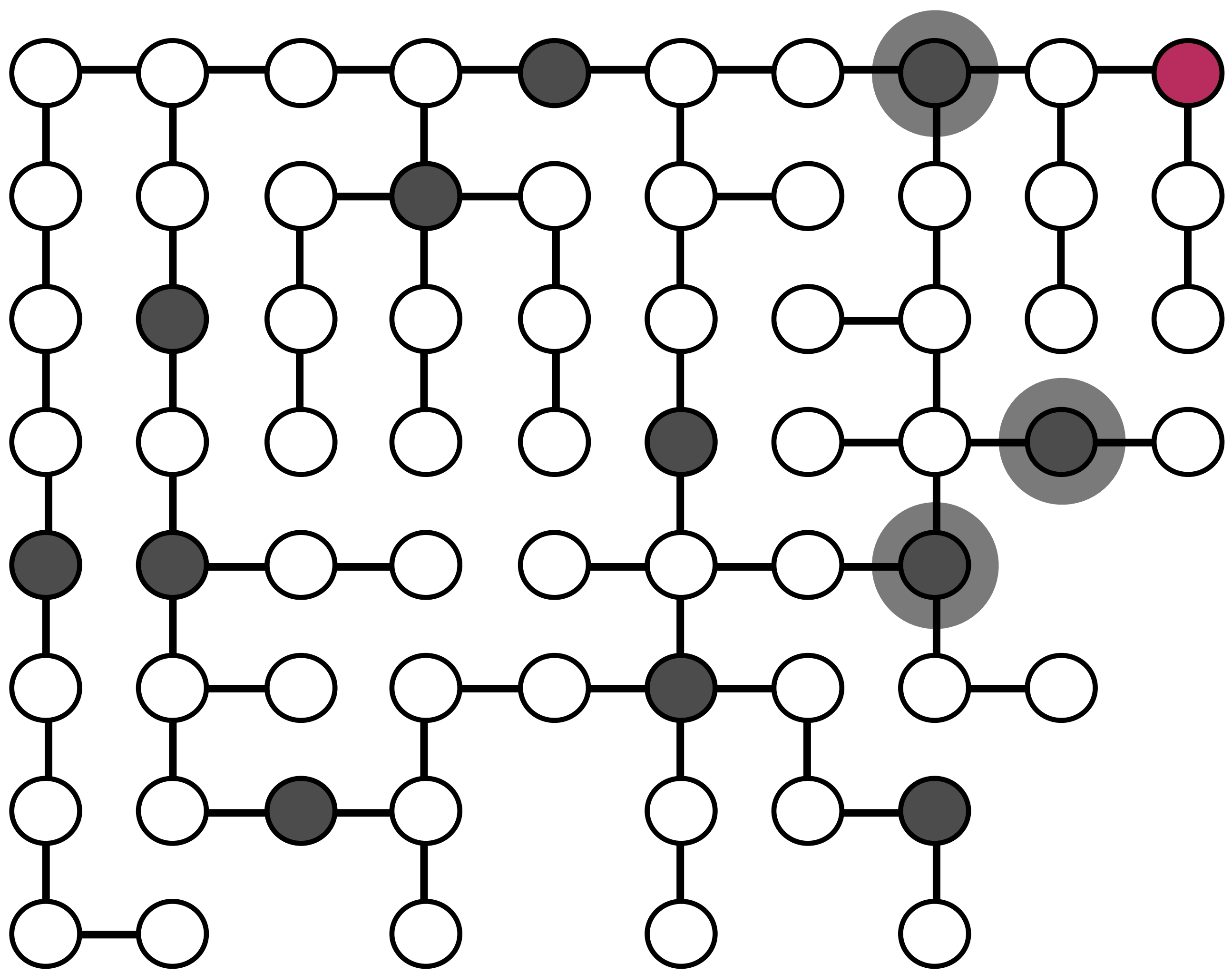
Consider the family of subsets of $\mathcal{C}$

$$\{\mathcal{C}_1, \ldots, \mathcal{C}_\ell\}$$

such that

$$\bigcup_{i=1}^{\ell} \mathcal{C}_i = \mathcal{C}$$

Distributed optimization algorithm



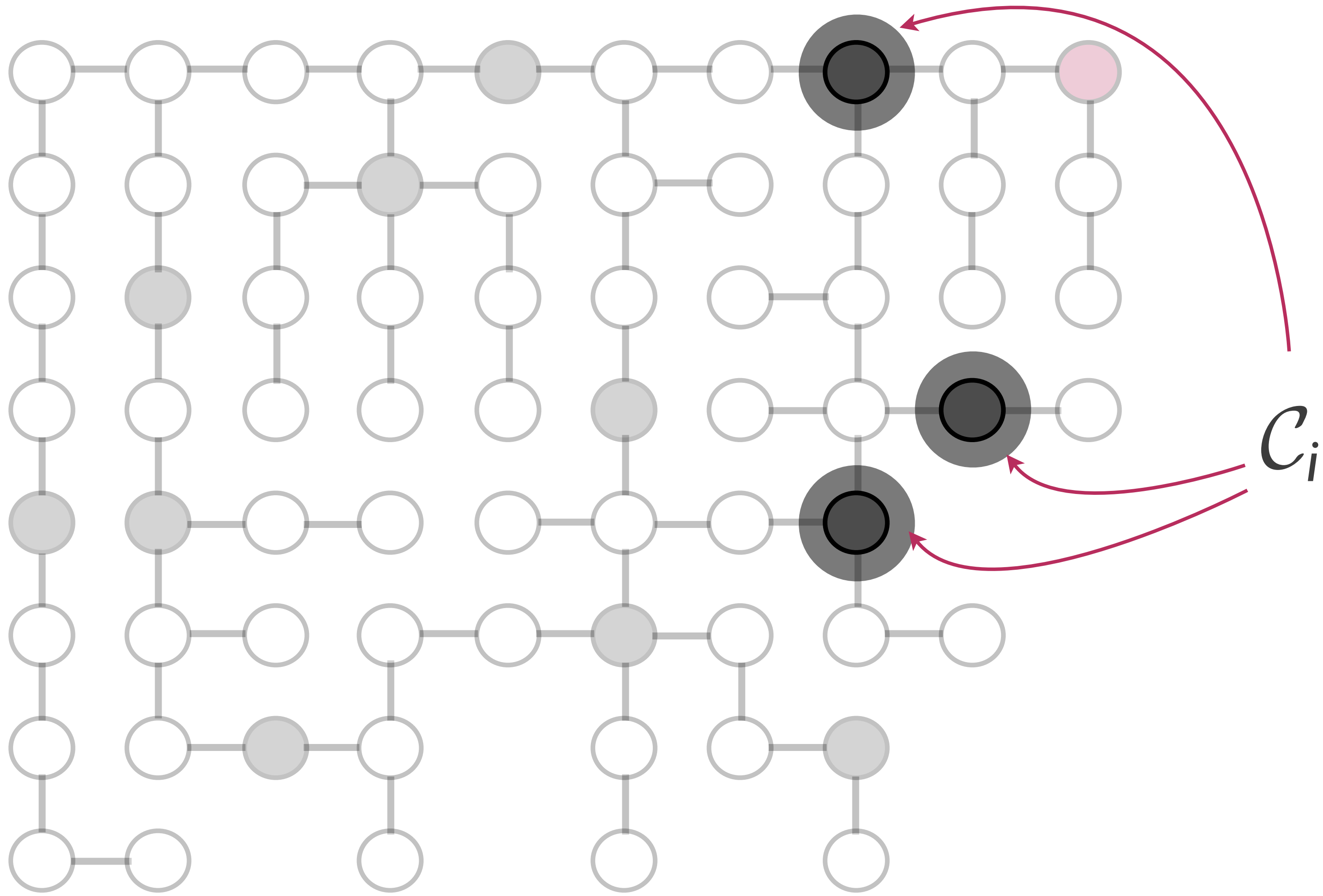
Consider the family of subsets of $\mathcal{C}$

$$\{\mathcal{C}_{\mathbf{1}}, \ldots, \mathcal{C}_\ell\}$$

such that

$$\bigcup_{i=\mathbf{1}}^{\ell} \mathcal{C}_i = \mathcal{C}$$

Distributed optimization algorithm



Distributed optimization algorithm

i-th optimization subproblem

Start from q and compute an update q' minimizing the cost by updating only the components of q belonging to $\mathcal{C}_i$, while keeping the others constant. Namely

$$\boxed{\min_{q'} q'^T M q' \quad \text{subject to } q' \in q + \mathcal{S}_i}$$

where

$$\mathcal{S}_i = \left\{ q \in \mathbb{R}^N \ : \ \sum q(x) = 0 \, , q(x) = 0 \ \forall x \notin \mathcal{C}_i \right\}$$

Distributed optimization algorithm

i-th optimization subproblem

Let

$$\boxed{T_i(q) := \operatorname*{argmin}_{q'} q'^T M q' \quad \text{subject to } q' \in q + \mathcal{S}_i}$$

Then $T_i(q)$ is a linear operator

$$T_i(q) = F_i q$$

Notice that

$$F_i := I - N_i M$$

and where N_i is a matrix depending on the local electrical properties of the power lines network. Therefore

$$\boxed{T_i(q) = q - N_i M q}$$

Distributed optimization algorithm

i-th optimization subproblem

$$T_i(q) = q - N_i M q$$

Notice that, while N is a sparse matrix

$$(N_i)_{hk} = 0 \text{ if } h, k \notin \mathcal{C}_i$$

Consequence: multiplying by the matrix N_i is a computation that each node can do "locally", namely from the information coming only from the nodes in $\mathcal{C}_i$.

$$q_h(t+1) = q_h(t) - \sum_{k \in \mathcal{C}_i} (N_i)_{hk} (Mq)_k$$

The node h needs to know only $(N_i)_{hk},\ (Mq)_k$ for $k \in \mathcal{C}_i$.

Distributed optimization algorithm

$$T_i(q) = q - N_i M q$$

On the contrary, the matrix M is not sparse and so the term Mq can not be computed locally, since it would require the knowledge of entire M (and so the global power network topology) and q.

i-th optimization subproblem

HYPOTHESIS: $Z(e) = e^{j\theta} R(e)$ where $R(e)$ is a real number.

In this case it can be shown that for $k \in \mathcal{C}_i$

$$\boxed{(Mq)_k \simeq \sum_{h \in \mathcal{C}_i} |u_k||u_h| \sin(\angle u_h - \angle u_k - \theta)}$$

namely $(Mq)_k$ can be computed locally from the voltages on the nodes in $\mathcal{C}_i$.

Distributed optimization algorithm

Consider a sequence $\{\sigma(t)\}$, $\sigma(t) \in \{1, \ldots, \ell\}$. When the symbol i appears in the sequence, the i-th subproblem is solved:

$$\boxed{q(t+1) = T_{\sigma(t)}[q(t)] = F_{\sigma(t)}q(t)}$$

The proposed optimization algorithm therefore consists of the following, repeated steps:

- a set $\mathcal{C}_i$ is chosen according to a sequence of symbols $\sigma(t) \in \{1, \ldots, \ell\}$;
- every agent j in $\mathcal{C}_i$ senses the network and obtain, directly or via some filtering, an estimate of $(Mq)_j$;
- they determine a feasible update step that minimizes the given cost function, coordinating their actions and communicating;
- they actuate the system by updating their state (the injected reactive power).

Convergence of the algorithm

Assumption

The sequence $\sigma(t)$ is a sequence of independently, identically distributed stochastic process valued in $\{1, \ldots, \ell\}$ with $\mathbb{P}[\sigma(t) = i] = p_i$.

Results

- We have conditions on $\{\mathcal{C}_1, \ldots, \mathcal{C}_\ell\}$ ensuring that the algorithm converges to the global minimum.
- We analyzed the convergence rate of the algorithm.
- In case the power network topology is a tree, we have conditions on $\{\mathcal{C}_1, \ldots, \mathcal{C}_\ell\}$ and on the probabilities $p_1, \ldots, p_\ell$ yielding the fastest convergence rate.

Convergence of the algorithm

Convergence to the global optimum

Let

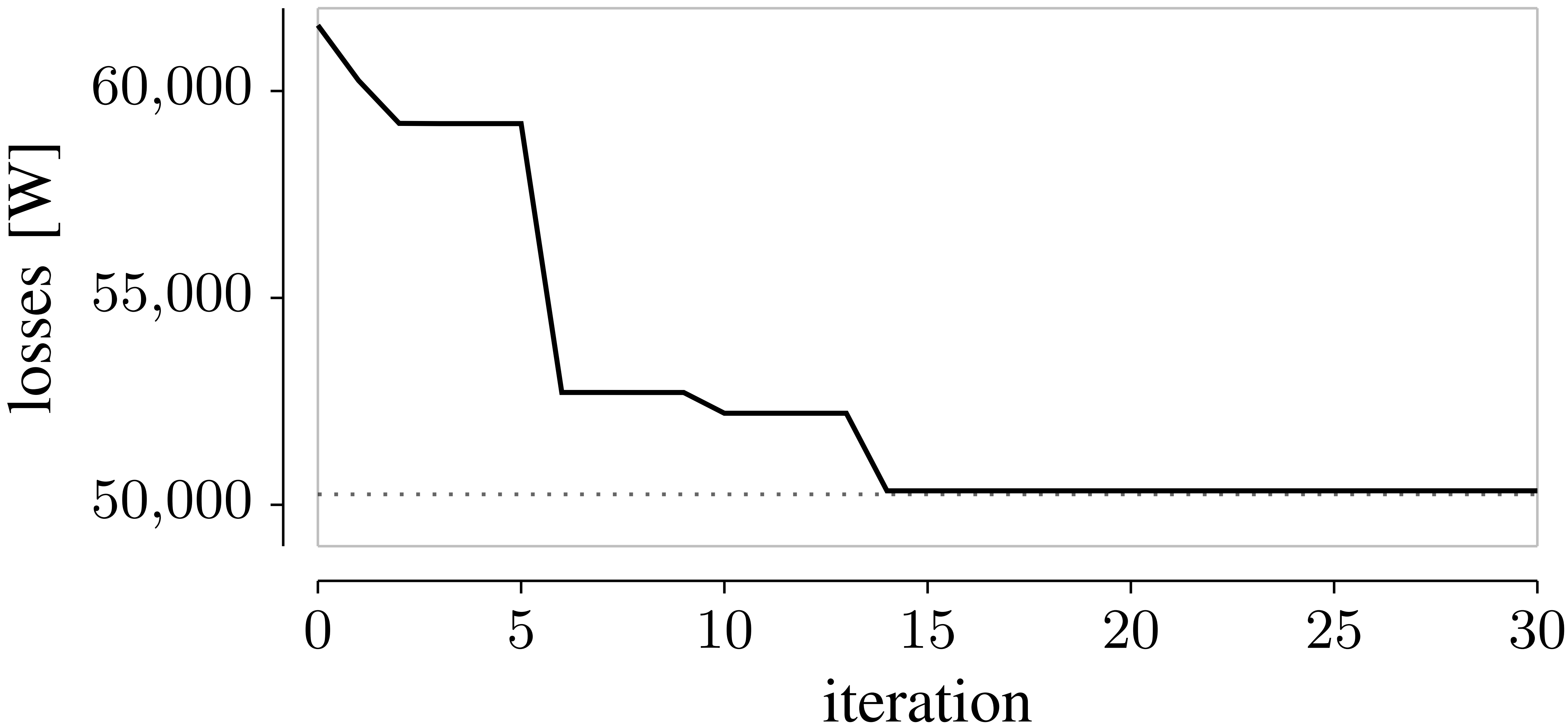
$$q^* := \operatorname{argmin} q^T M q \qquad \text{subject to } \mathbf{1}^T q = 0 \text{ and } q(x) = \bar{q}(x),\ \forall x \in \mathcal{U}$$

Then

$$q(t) \longrightarrow q^*$$

if and only if the hypergraph with hyperedges $\mathcal{C}_1, \ldots, \mathcal{C}_\ell$ is connected.

Simulation



Future research

- proving that clustering neighbor nodes is indeed the best choice for general graphs;
- introducing operating limits for compensators;
- considering a dynamical optimization problem that includes stochastically varying demands;
- obtaining a dynamic nonlinear model and its linearization which capture the same features of the static models presented here;
- designing dynamic filtering algorithms that allow local estimation of the gradient of the cost function from voltage measurements.

Conclusions

- The consensus algorithm is an instance of a completely distributed design. This is an extreme design paradigm.
- It is intrinsically robust to external changes and highly self-adaptive so that a limited initial configuration and tuning effort is necessary.
- None or limited information about the global structure of the system is necessary to the units.
- Graceful performance degradation.
- Importance of the interaction network topology.