

# Control of Prosumer Networks

Nicolas Gensollen, Monique Becker, Vincent Gauthier and Michel Marot

CNRS UMR 5157 SAMOVAR,

Telecom SudParis/Institut Mines Telecom

Email: {nicolas.gensollen, vincent.gauthier, michel.marot, monique.becker}@telecom-sudparis.eu

**Abstract**—The ability to control a given network by dynamically injecting inputs offers ways to make sure that it stays in stable regions. In the case of smart grids, where end users exhibit complex behaviors and renewable production is quite unstable, control of the system is paramount. In this paper, we study the case of a network composed of entities called prosumers. These agents have the ability to both consume and produce according to external conditions. We first model and simplify the underlying dynamic of such network using a second order coupled oscillators network model. Under some conditions, the system synchronize to a common frequency. However, in case of perturbations in the power distribution, the system might lose synchrony and require control to bring it back to the stable state. In this situation, control can be seen as energy absorbed or injected in the system at specific locations. Moreover, the power outputs of the prosumers are susceptible to change such that loads and generators are not fixed. In this context, it is important to select the right subset of nodes that yields, on average, the cheapest control in terms of energy.

## I. INTRODUCTION

Modernizing the power grids and increasing the share of renewables in the production are two substantial objectives of the energetic transition. Because of parallel progresses in communication, data management, and storage, the upcoming emergence of a power grid "2.0", often called smart grid, appears as a cross-disciplinary challenge of the 21st century [1].

It is believed that these new ways of producing and managing energy, will also impact how end users consume. More precisely, the consumer of the future is expected to be more involved in the system operation. Using dynamic signals (real time electricity prices for instance), load curves could be shaped to some extend as to agree with the production conditions of the main grid. Furthermore, the emergence of renewable distributed energy resources (DER) in the distribution networks, possibly hold by individuals, will completely modify the today end user agent model.

These new agents that can both consume and produce are often called prosumers [2]. This decentralized production is believed to lead to more resilient systems than today centralized power grid. Nonetheless, the difficulty of dealing with decentralized production is coupled with the high uncertainty of renewables. Ensure system stability in these conditions appears thus as a complex task.

Relying on the ability of the smart grid to obtain, communicate and analyse data, several solutions could improve the stability of such complex systems. Among these solutions, demand side management and energy storage are two of the

most popular ones. In this paper, we explore the use of storage in a network of prosumers whose power outputs are susceptible to change due to various external causes. More precisely, we are interested in the "best" locations for the storage equipments such that we are able to maintain stability when perturbations occur. Since all nodes can switch from generator to load and vice versa, the optimal location is susceptible to change with the prosumers power output. Our goal is then to find the one that, on average, provide the best results.

Since charging and discharging a battery is costly, we are particularly interested in the energy necessary to control the system when perturbations occur. The notion of performance of a given set of locations for using storage would then be inversely proportional to the energy used. One problem with this definition is that it depends explicitly on the state of the system and the prosumers power distribution. Besides, as the system size increases, the number of possible location sets increases exponentially. This forbid any algorithms using complete enumeration and evaluation of the sets. Moreover, we consider that the electrical lines have maximum capacities, and storage equipments have both a maximum charge / discharge rate and bounded capacities. Obviously, only solutions satisfying these constraints will be considered as feasible.

In this paper, we model the prosumer network as a coupled oscillator network. Inspired by [3], we use a second order Kuramoto model that can be seen as a simplified version of the underlying power grid dynamics (see Section II). We then use control theory [4] in order to model the impact of the storage equipments on the network (see Section III). Using the relationships between submodular set functions and the Gramian matrix [5], we use a modified version of a greedy algorithm [6] in order to find the smallest locations set that minimizes the average control energy while satisfying all the constraints (see section IV). Finally, we show some results in Section V.

## II. THE MODEL

In this section, we introduce the coupled oscillators network model used to simplify the power grid dynamic. The detailed derivation of the second order dynamics can be found in [3], and the key points are reproduced here for the convenience of the reader.

The objective is to achieve synchronization of the grid at the main frequency  $\Omega = 50\text{Hz}$ . Each oscillator  $i$  has a phase angle  $\delta_i$  and a frequency  $\dot{\delta}_i$ . Therefore, we seek an equilibrium of the form :  $\forall i, \dot{\delta}_i = \Omega$ . For convenience, we express the dynamic

of the oscillators in terms of the deviations from the main frequency :  $\delta_i(t) = \Omega t + \theta_i(t)$ . Let  $\omega_i = \dot{\theta}_i$ , such that  $\dot{\delta}_i = \Omega + \omega_i$ . The equilibrium, in terms of the deviations dynamics, is :  $\forall i, \omega_i = 0$

The next step consists in translating the dynamics of the generators and machines into equations involving the phase angles  $\theta_i$  and the frequencies  $\omega_i$ . Generators and machines are composed of a turbine that dissipates energy at a rate proportional to the square of the angular velocity ( $P_{diss} = FV \sim V^2$ ) :

$$P_{diss,i}(t) = K_{Di}(\dot{\delta}_i(t))^2 \quad (1)$$

where  $K_{Di}$  is the dissipation constant of entity i.

Furthermore, it also accumulates kinetic energy at a rate :

$$P_{acc,i}(t) = \frac{1}{2} I_i \frac{d}{dt} (\dot{\delta}_i(t)^2) \quad (2)$$

where  $I_i$  is the moment of inertia of entity i. For simplicity, we consider that all entities have the same dissipation constants( $K_D$ ) and moment of inertia (I).

The condition for the power transmission between i and j is that the two devices do not operate in phase : The phase difference between i and j is :  $\delta_j(t) - \delta_i(t) = \Omega t + \theta_j(t) - \Omega t - \theta_i(t) = \theta_j(t) - \theta_i(t)$ .

The transmitted power along the line can thus be written as :

$$P_{transmitted} = -P_{ij}^{MAX} \sin(\theta_j - \theta_i) \quad (3)$$

with  $P_{ij}^{MAX}$  being the maximum capacity of the line ( $i, j$ ). If i is connected to more than one other entities, this equation becomes ( $\mathcal{N}_i$  being the neighborhood of i) :

$$P_{transmitted} = - \sum_{j \in \mathcal{N}_i} P_{ij}^{MAX} \sin(\theta_j - \theta_i) \quad (4)$$

Each entity i is then described by a power balance equation of the type :

$$P_{S,i} = P_{diss,i} + P_{acc,i} + P_{transmitted,i} \quad (5)$$

By substituting and re-arranging the terms, we obtain the following non-linear coupled system of equations :

$$\begin{aligned} P_{S,i} &= I\Omega\ddot{\theta}_i + [I\dot{\theta}_i + 2K_D\Omega] \dot{\theta}_i + K_D\Omega^2 \\ &\quad + K_D\dot{\theta}_i^2 - \sum_{j \in \mathcal{N}_i} P_{ij}^{MAX} \sin[\theta_j - \theta_i] \end{aligned} \quad (6)$$

These non linear equations are obviously not very practical. We now use some approximations based on the fact that we consider only small deviations from the main frequency :  $\dot{\delta}_i \sim \Omega$  which means that  $\omega_i = \dot{\theta}_i \ll \Omega$ , such that the squared term  $K_D\dot{\theta}_i^2$  can be neglected. Moreover, we assume that the rate at which energy is stored in the kinetic term is much less of the rate at which energy is dissipated by friction, a common condition for mechanical systems :  $\dot{\theta}_i \ll \frac{2K_D}{I}$

The equation can thus be simplified to :

$$\ddot{\theta}_i \sim \psi_i - \alpha\dot{\theta}_i + \sum_{j \neq i} K_{ij}g_{ji} \sin[\theta_j - \theta_i] \quad (7)$$

Where  $\alpha = \frac{2K_D}{I}$  is the dissipation term,  $K_{ij} = \frac{P_{ij}^{MAX}}{I\Omega}$  are the coupling strengths,  $\psi_i = \left[ \frac{P_{S,i}}{I\Omega} - \frac{K_D\Omega}{I} \right]$ , and  $g_{ji}$  is the coefficient of the adjacency matrix G. By working in a rotating frame, we can simplify  $\psi_i = \frac{P_{S,i}}{I\Omega}$ .

The dynamic is still non linear because of the sine coupling. Therefore, we also assume that the phase angle differences are small such that  $\sin[\theta_j - \theta_i] \sim \theta_j - \theta_i$ . By using vector notations, the dynamic can be written in the following form :

$$\ddot{\theta} = \Psi - \alpha\dot{\theta} - KL\theta \quad (8)$$

Where L is the Laplacian matrix of the underlying topology ( $k_i$  is the degree of node i):

$$L_{ij} = \begin{cases} k_i & \text{if } i = j \\ -g_{ij} & \text{if } i \neq j \end{cases} \quad (9)$$

This is a continuous time second order linear system of  $N$  equations. We first transform this into a continuous time first order linear system of  $2N$  equations by introducing the following vector :  $X = \begin{pmatrix} \theta \\ \dot{\theta} \end{pmatrix}$  :

$$\dot{X} = \begin{pmatrix} 0 & I \\ -KL & -\alpha I \end{pmatrix} X + \begin{pmatrix} 0 \\ \Psi \end{pmatrix} \quad (10)$$

Which can be written in discrete time:

$$X(t + \Delta t) = MX(t) + \begin{pmatrix} 0 \\ \Psi\Delta t \end{pmatrix} \quad (11)$$

$$\text{With } M = \begin{pmatrix} I & I\Delta t \\ -KL\Delta t & (1 - \alpha\Delta t)I \end{pmatrix}.$$

By setting  $Y(t) = \begin{pmatrix} X(t) \\ 1 \end{pmatrix}$ , let the dynamic be :

$$Y(t + \Delta t) = AY(t) \quad (12)$$

With transition matrix A :

$$A = \begin{pmatrix} I & I\Delta t & 0 \\ -KL\Delta t & (1 - \alpha\Delta t)I & \Psi\Delta t \\ 0 & 0 & 1 \end{pmatrix} \quad (13)$$

This is a linear discrete time system of  $2N + 1$  equations. Note that the transition matrix A encodes all the system parameters, topology, and power distribution.

### III. CONTROL AND SUBMODULARITY

#### A. Submodularity

In this section we provide a very brief introduction to what submodular functions are, and how their maximization could be achieved in reasonable time. More information can be found in [7].

A set function  $F : 2^V \rightarrow \mathbb{R}$  defined over a finite set V is said to be submodular if it satisfies :

$$\forall S, T \in V, F(S) + F(T) \geq F(S \cup T) + F(S \cap T) \quad (14)$$

Another equivalent definition is that for all sets  $X, Y \in V$ , such that  $X \subseteq Y$  and for all element  $x \in V \setminus Y$ ,

$$F(X \cup \{x\}) - F(X) \geq F(Y \cup \{x\}) - F(Y) \quad (15)$$

This basically means that submodular functions exhibit a diminishing return property which makes them particularly interesting for optimization.

Finding the set  $S_k$  of size  $k$  that maximizes a set function  $F$  is a difficult problem because the number of sets grows exponentially with the number of nodes. Therefore complete enumeration and evaluation is only a feasible solution on very small examples. Nevertheless, if the set function is submodular, a simple greedy heuristic returns a solution  $S_k^*$  such that, in the worst case,  $\frac{F(S_k^*)}{F(S_k^{OPT})} \sim 63\%$  (where  $S_k^{OPT}$  is the optimum set of size  $k$ ) [7].

Since the evaluation of  $F$  can be costly, a well-known lazy-greedy variation has been proposed by [6]. This smart implementation uses the submodular structure of the marginal gains in order to reduce the number of calls to  $F$ . This can result in speedups of several order of magnitudes.

### B. Control

In this section, we introduce basic tools from control theory and relate them to the submodular functions introduced above.

We assume the control of a subset of the nodes that we call driver nodes. In our particular power grid situation, a driver node can be seen as a prosumer where some additional equipment (battery, backup generator,...) will be used as to inject or absorb energy from the system. This control assumption modifies the system dynamic :

$$Y(t + \Delta t) = AY(t) + Bu(t) \quad (16)$$

Where the control matrix  $B$  is an indicator of the driver nodes, and  $u(t)$  represents the control vector input at time  $t$ . The system is said to be controllable in  $T$  steps if it can be steered from any initial state  $Y_0$  to any final state  $Y_f$  through a sequence of control inputs  $u(t)$ . If the system is controllable, there might be more than one sequence of control inputs that could achieve such a thing. Among all these possibilities, we are interested in the one that requires the minimum amount of energy [8]. Let  $\mathcal{E} = \int_{t_0}^{t_0+T} \|u(t)\|^2 dt$  be the energy used for the control of the system.

It can be shown [4] that the control input sequence that minimizes the control energy can be written as :

$$u^*(t) = B^T (A^T)^{T-t-1} W^{-1}(T) \nu(T) \quad (17)$$

where  $\nu(T) = Y_f - A^T Y_0$  is the difference between the desired final state and the final free state, and  $W(T) = \sum_{k=0}^T A^k B B^T (A^T)^k$  is called the Gramian matrix of the system. The Gramian is a popular tool in control theory since

it is deeply linked to the controllability of the system. It can indeed be shown that the system is controllable if and only if the Gramian is not singular, and that its rank indicates the dimension of the controllable subspace. Besides, the minimum control energy associated with the inputs  $u^*(t)$  can be written as :

$$\mathcal{E}_{min} = \nu(T)^T W^{-1}(T) \nu(T) \quad (18)$$

Not surprisingly, the control energy depends on the initial and final states as well as on the inverse of the Gramian. In our prosumer case, we want to find the best battery placement, whose quality should not depend on the initial and final states. Indeed, these states as well as the power distribution  $\Psi$  are susceptible to change, and we would like to perform well, on average, whatever the states and distributions. Therefore, we concentrate on the Gramian as a way to quantify the average controllability. It can indeed be verified that the Gramian does not depend on  $\Psi$ .

A key point to design a low consumption control is thus to understand the relationships between the control node set and the Gramian matrix. In [], the authors introduce several Gramian-based metrics that have concrete meanings in terms of the system controllability. For example, the trace of the inverse Gramian ( $Tr[W^{-1}(T)]$ ) quantifies the average control energy necessary to move the system around the state space. These metrics measures some notions of the control allowed by a given driver node set. But they can be understood the other way around : what drivers should one select as to optimize a given control-based metric ?

Using the Gramian-based metric  $\mathcal{M}$ , let  $F(S) = \mathcal{M}(W_S(T))$ , where  $W_S(T)$  is the Gramian matrix based on the driver node set  $S$ . [] showed that for some metrics,  $F$  is a submodular set function. This means that, for these metrics, we can use the greedy heuristic discussed above in order to maximize  $F$ .

## IV. CONTROL OF THE PROSUMER NETWORK

### A. Controllability Constraints

Considering the dynamic of equation (), we notice that the Gramian based on the transition matrix  $A$  is never invertible, meaning that the system should not be controllable. It is easy to see that, indeed, we do not have full control because of the last row of the matrix. Nevertheless, as this row was only added in order to include the power distribution inside  $A$ , we do not need to control the system along this "fake" dimension. That is, we only seek the control of the system in the first  $2N$  dimensions of the space ( $N$  phase angles  $\theta_i$  and  $N$  frequencies  $\omega_i$ ). Therefore, whenever  $rank[W(T)] = 2N$ , we will use the Moore-Penrose pseudo inverse of the gramian  $W^\dagger(T)$ .

### B. Flow Constraints

Recall that the simplified dynamic (see section II) implies that, the power transmitted on line  $(i, j)$  is  $P_{i \rightarrow j} = -P_{ij}^{MAX}(\theta_j(t) - \theta_i(t))$ . Therefore, we write the flow constraints at time  $t$  as :

$$\forall(i,j) \in V^2, \quad g_{ij} | \theta_j(t) - \theta_i(t) | \leq 1 \quad (19)$$

If this constraint is verified for all instant t in the control time range, then the control inputs satisfy the flow constraints.

### C. Power Distribution Constraints

As explained above, we wish to obtain results that do not depend on the power distribution  $\Psi$  of the prosumers. However, in order to achieve synchronization along with feasible power flows, we need to define some constraints on this distribution. As shown in [], a condition for achieving synchronisation in a coupled oscillators network is  $\|L^\dagger \omega\|_\infty \leq \sin(\gamma)$ , where L is the Laplacian matrix of the network,  $\omega$  is the vector of the natural frequencies of the oscillators, and  $\|x\|_\infty = \max|x_i - x_j|$ . If this condition is satisfied, the oscillators will synchronize at the common frequency  $\omega_{SYNC}$  with the phase lock  $g_{ij} |\theta_i - \theta_j| \leq \gamma \in [0, \frac{\pi}{2}]$ ,  $\forall(i,j) \in V^2$ .

Therefore, we can write the following synchronization constraint :

$$\| (L \circ P^{MAX})^\dagger \Psi \|_\infty \leq \sin(1) \quad (20)$$

Where  $A \circ B$  represents the Hadamard product between matrices A and B. If constraint 20 is satisfied, the oscillators will synchronize to a common frequency  $\omega_{SYNC} = \frac{\sum_{k=1}^N \Psi_k}{\sum_{k=1}^N \alpha_k} = \frac{\sum_{k=1}^N P_{S,k}}{I\Omega N \alpha}$ , and the flows at steady state will be feasible. Note that  $\omega_{SYNC}$  should be zero in order to synchronize to the main frequency  $\Omega$  (recall that the phase angles and frequencies represent deviations from  $\Omega$ ). This gives the following intuitive zero sum constraint :

$$\sum_{k=1}^N P_{S,k} = 0 \quad (21)$$

Constraint 21 simply states that production should match demand in order for the system to remain stable.

### D. Battery Constraints

Let i be a battery. In the simple model we adopt here, it is characterised by its maximum charge/discharge rate  $r_i$ , the amount of energy stored at time t  $\Lambda_i(t)$ , and its maximum capacity  $\Lambda_{i,MAX}$ . For simplicity, we will consider all maximum charge/discharge rates to be the same ( $r$ ), and all maximum capacities to be the same  $\Lambda_{Max}$ . Furthermore, we denote by  $\Lambda(t)$  the vector of energy level at time t.

Since the control input vector  $u^*(t)$  specifies the control inputs, the energy level dynamic of the batteries can be written as :

$$\Lambda(t + \Delta t) = \Lambda(t) - u^*(t)I\Omega \quad (22)$$

Which can also be written as :

$$\Lambda(t) = \Lambda(0) - I\Omega \sum_{k=0}^t u^*(k) \quad (23)$$

Where  $\Lambda(0)$  is the vector of initial levels in the batteries.

Obviously, the energy level in a battery is bounded. This leads to the following set of constraints :

$$\forall t, \quad 0 \leq \Lambda(t) \leq \Lambda_{MAX} \quad (24)$$

Which can be written in terms of the control inputs :

$$\forall t, -\frac{\Lambda(0)}{I\Omega} \leq -\sum_{k=0}^t u^*(k) \leq \frac{\Lambda_{MAX} - \Lambda(0)}{I\Omega} \quad (25)$$

Finally, during each time slot, the battery cannot charge or discharge at a rate higher than  $r$  :

$$\forall t, \quad |u^*(t)I\Omega| \leq r \quad (26)$$

If, for a given  $u^*$ , the constraints of eq. 25 and 26 are satisfied, then the control inputs satisfy the storage constraints.

### E. Algorithm

We propose the following greedy algorithm in order to find the smallest set of locations that minimizes the average control energy and satisfy the constraints. We assume here that the system is in a given initial state  $Y_0$  and we want to reach a desired final state  $Y_f$  in T time steps. The algorithm ?? starts with an empty set and, as long as that constraints are not satisfied, it increases the size of the set. For a given size k, the set  $S_k$  is found using lazy greedy submodular optimization of a Gramian-based set function F. In practice we use  $F(S) = -Tr[W_S^\dagger(T)]$  which quantifies the average energy required to move the system around the controllable subspace. The algorithm ?? returns then the tuple  $(k, S_k, u^*)$  such that  $S_k$  is the smallest set that minimizes the average control energy and, in this case, satisfies the constraints with control  $u^*$ .

---

```

while Not  $\left\{ \begin{array}{l} \text{rank}[W_{S_k}(T)] < 2N \text{ and} \\ \forall t, \forall(i,j) \in V^2, g_{ij} |\theta_j(t) - \theta_i(t)| \leq 1 \text{ and} \\ \forall t, -\frac{\Lambda(0)}{I\Omega} \leq -\sum_{k=0}^t u^*(k) \leq \frac{\Lambda_{MAX} - \Lambda(0)}{I\Omega} \text{ and} \\ \forall t, |u^*(t)| \leq \frac{r}{I\Omega} \end{array} \right.$ 
do
   $k \quad \leftarrow k + 1$ 
   $S_k \quad \leftarrow Subopt(F, k)$ 
   $u^*(t) \quad \leftarrow OptControl(S_k)$ 
end while

```

---

## V. RESULTS

For a direct illustrative application, we chose a small system of 15 nodes connected with an erdos-renyi topology. The power distribution and the capacities of the lines are selected randomly such that the system is at equilibrium. Therefore, the production matches the consumption, no line is overloaded and all nodes are synchronized to the main frequency. We use the greedy algorithm (X) based on the trace of the inverse Gramian as to select the driver set  $S$ . At a given time  $t_k$ , we introduce a small perturbation in the form of a mismatch between the production and consumption :  $P_i(t_k) = P_i(t_{k-1}) + \delta P$  for

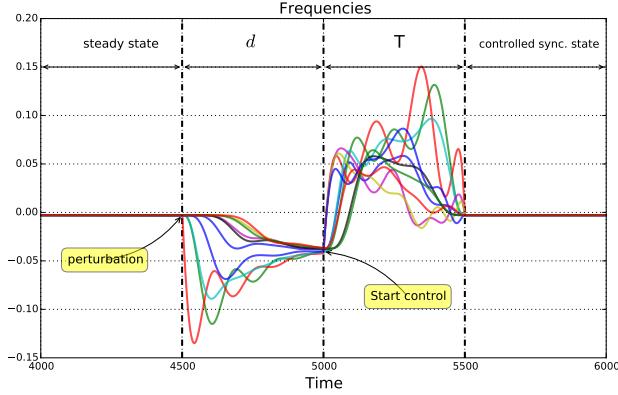


Figure 1: Time series of the  $N$  frequencies  $\omega_i$  (phase angles are not shown for clarity).  $d$  is the delay between the perturbation and the begining of the control phase that bring the system bak in the synchronized state in  $T$  time steps. Here,  $d$  was chosen quite large for the clarity of the figure. Note that, if the perturbation is still active at this point, the system still needs control in order to stay synchronized (batteries should still compensate the power imbalance)

some random node  $i$ . Consequently, the frequencies  $\dot{\theta}_i \forall i$  of the oscillators start to deviate from the synchronized state. At time  $t_k + d$ , we apply to the drivers the optimal control of equation (X) such that the system is brought to the synchronized state at time  $t_k + d + T$ , where  $T$  is the control time. At this time, if the mismatch between production and consumption is still active, we maintain synchrony by using optimal control with  $T = 1$ . Figure 1 shows the evolution of a subset (for clarity) of the nodes' frequencies during the control phase.

The purpose of selecting the drivers according to a Gramian related metric is to minimize, on average, the amount of control energy needed. In other words, if we select the drivers randomly, we expect to need, on average, more energy to control the system. In figure 2, we draw 200 example systems of 15 nodes with random topologies, power distributions, and line capacities. These systems are initially at equilibrium and are perturbed as explained previously. For each system we find the driver set  $S$  using algorithm (X) and we draw a random driver set  $S'$  such that  $S$  and  $S'$  have the same size, and such that  $S'$  also provide full control of the system. We then compute the control signals and the corresponding control energies  $\mathcal{E}_S$  and  $\mathcal{E}_{S'}$  for both  $S$  and  $S'$ . Each system can therefore be represented as a point with coordinates  $(|S|, \mathcal{E}_S)$ . On figure 2, we plot the densities of these points for the random case (left pannel) and the optimized case (right pannel). Regions in dark red contains a lot of points while regions in dark blue have very few points. Figure 2 shows that most of the systems considered tend to be controlled by the drivers of algorithm X with less energy and/or drivers than for the random case.

We investigate next how the topology of the grid affects

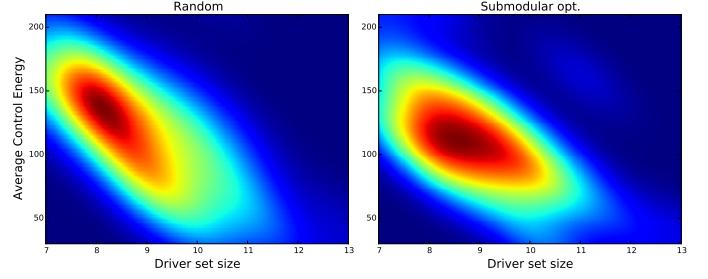


Figure 2: test

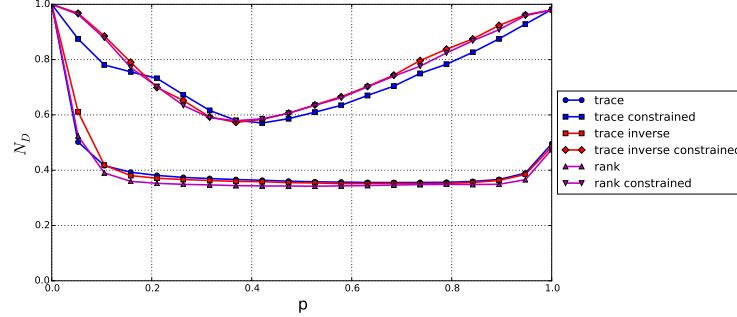


Figure 3:  $n_D$  against link probability  $p$  for erdos-renyi topologies ( $N = 100$ ). Curves are averaged over 100 realizations. The top three curves show the results for the three metrics taken into consideration when all constraints are considered (see Level 2 in the main text). The bottom three curves exhibit the results for the same metrics when only the full controllability constraint is considered (see Level 1 in the main text).

the size of the control set. We consider first the case of an erdos renyi topology with probability of connection  $p$ . We show on Figure 3 how the size of the driver set evolves with  $p$  for different Gramian based metrics and for two levels of constraints :

- Level 1 : full control
- Level 2 : full control + line capacities constraints + battery constraints (see algorithm X)

When  $p$  is close to zero, nodes tend to be very poorly connected such that we need to control almost all nodes in the grid. As  $p$  increases, the connectivity of the graph augments and the number of drivers decreases. At some point, the connectivity of the graph starts to harm its controllability, and more drivers are needed. This effect is in accordance with the litterature although it is less marked here because the topology has less influence in our model (recall that matrix  $A$  in the dynamics is not the adjacency matrix of the graph but a block matrix where one of the block is the Laplacian matrix of the graph). As expected, the driver sets for the level 2 of constraints are larger than for level 1 (for all metrics) because we impose far more constraints.

Because transmission power grids are spatially and politically constrained (each country has its own transmission grid

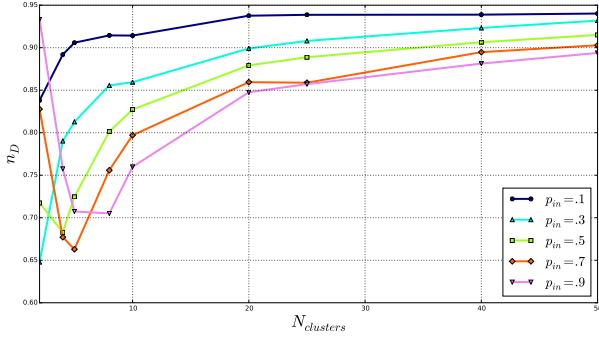


Figure 4:  $n_D$  against  $N_{clusters}$  for different values of  $p_{in}$  and  $p_{out} = 0.1$ .  $N = 200$  and curves are averaged over 100 realizations.

which is interconnected with neighboring grids at some specific locations), we study the control in clustered graphs. More precisely, we use a block model ( $N, N_{clusters}, p_{in}, p_{out}$ ) to generate random topologies where  $N$  is the number of nodes,  $N_{clusters}$  is the number of clusters,  $p_{in}$  is the probability that two nodes within the same cluster are linked, and  $p_{out}$  is the probability that two nodes in two distinct clusters are connected.

Figure 4 shows how the number of drivers evolves with the number of clusters for different values of  $p_{in}$  when  $p_{out}$  is fixed ( $p_{out} = 0.1$ ). For small values of  $p_{in} \sim p_{out}$  clusters are poorly marked, and the connectivity is low. The number of drivers in these conditions is large and increases slowly when the number of clusters augments. As  $p_{in}$  increases, the clusters becomes more densely connected such that within a cluster less drivers are required in order to control it. As the number of clusters grows, more drivers are required. For large values of  $p_{in}$ , the behaviour is more complex. We see indeed that for  $p_{in} = 0.9$  and  $N_{clusters} = 2$ , almost 94% of the nodes are needed for control, but this quantity first decreases with the addition of a few more clusters (71% for  $N_{clusters} = 5$ ), before increasing. This behavior can be explained by the fact that controlling a very dense network requires, in our settings, a very large portion of nodes to be drivers. Controlling one cluster alone thus requires to control almost all nodes within this cluster. Nevertheless, when a relatively small (compared to the number of nodes in the graph) number of clusters are interconnected with a few links, nodes in one cluster are able to control nodes in other clusters to which they are connected. As the number of clusters grows, the global connectivity increases rapidly, such that the number of drivers also rises. We show this behavior in more details in figure 5, where the number of drivers is plotted against  $p_{in}$  for  $N = 200$  and  $N_{clusters} \in \{2, 4, 5\}$ . For  $N_{clusters} = 2$  and  $p_{in} = 0.1$  the number of drivers is large since the graph is poorly connected. When  $p_{in}$  increases,  $n_D$  decreases until a minimum value  $n_D \sim 0.64$  at  $p_{in} \sim 0.6$ . After this point,  $n_D$  augments with  $p_{in}$ . For  $N_{clusters} = 5$ , we do not see this behavior :  $n_D$  decreases

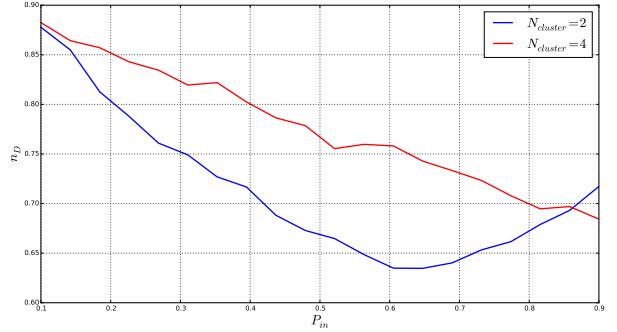


Figure 5:  $n_D$  against  $p_{in}$  for systems of  $N = 200$ ,  $p_{out} = 0.1$ , and for different values of  $N_{clusters}$ . The curves are averaged over 100 realizations.

continuously as  $p_{in}$  increases as expected from the curves of figure 4.

Until now, we have considered all batteries to be equal, i.e they have the same capacity and charge/discharge rate. Although simple, this assumption might not be verified in practice where different types of batteries exist. Making a cost-oriented study where the location os various types of batteries is studied such that the tradeoff between costs and reliability is out of the scope of this paper. Nevertheless, we studied how the number of drivers behaves when the batteries have different capacities. More precisely, we draw the capacities of the batteries from a normal distribution  $\mathcal{N}(\mu_\lambda, \sigma_\lambda)$  and keep a simple erdos-renyi topology for the power grid. Note that each node is assigned a capacity regardless of its characteristics (degree, betweenness, and so on...). In figure 6, we plot  $n_D$  as a function of the relative standard deviation of the battery capacity distribution  $\sigma_\lambda/\mu_\lambda$ . We see that  $n_D$  increases with the variance of the capacity distribution for all three gramian based metrics considered. Moreover, this rise is abrupt since  $n_D$  goes from 60% for  $\frac{\sigma_\lambda}{\mu_\lambda} \sim 0.15$  to 100% for  $\frac{\sigma_\lambda}{\mu_\lambda} \sim 0.27$ .

Real power grids topologies are known to be far from random such that using the erdos-renyi model probably do not give a realistic insight into the control of these systems. Depending on the voltage level

In this section, we provide some results in order to illustrate the work described above. We use a small 10 prosumers example, connected with a random topology. The line maximum capacities are selected according to a normal distribution  $\mathcal{N}(\mu_l, \sigma_l)$ . The power distribution is chosen as a zero mean normal distribution  $\mathcal{N}(0, \sigma_p)$ , and the samples should satisfy the constraints 20 and 21. Obviously,  $\mu_l$ ,  $\sigma_l$ , and  $\sigma_p$  should be chosen such that, without any perturbation, the lines are not already overloaded.

Once the synchronized state is reached, we apply a perturbation to the power distribution such that there is a mismatch between production and demand (constraint 21 is not satisfied). We denote by  $d$  the delay between the perturbation and the start of the control phase (see figure ??).  $d$  depends on how

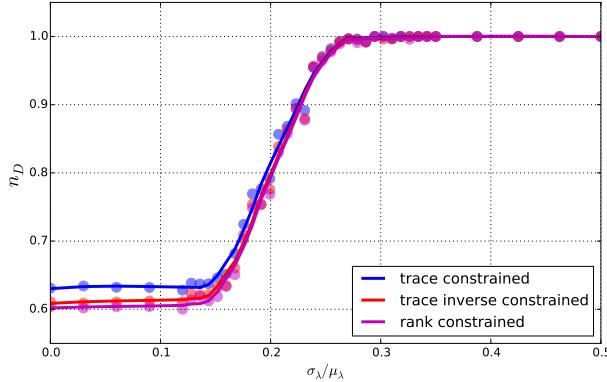


Figure 6:  $n_D$  against the relative standard deviation of the capacity distribution of the batteries  $\frac{\sigma_\lambda}{\mu_\lambda}$ . The dots are averages over 100 realizations and the curves are obtained using a Savitzky-Golay filter, each color shows the results for a given metric. The graphs are erdos-renyi with  $N = 100$  and  $p = 0.3$ , and  $\mu_\lambda = 100$  units.

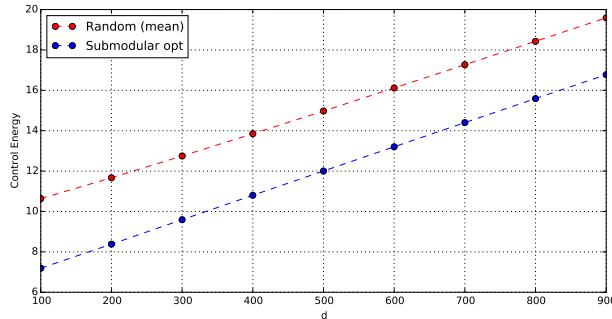


Figure 7: Control energy used in order to bring the system back to the synchronized state when the delay between perturbation and control increases, while the driver node set is fixed. The blue dotted curve shows the results for  $S_k^*$  our algorithm ???. The red line shows (average and standard deviation) the results when the driver set is sampled uniformly from all possible sets of the size  $k$ .

fast the perturbation can be identified and located, the control inputs calculated and communicated without error through the communication network.

If the driver set is fixed, we expect that the higher  $d$ , the higher the control energy required to drive the system back to the synchronized state. Eventually, if  $d$  is too large, the minimum energy control inputs will result in some constraints being broken (battery levels, line flows, and so on...).

As visible on figure ???, when  $d$  increases, the system deviates more from the equilibrium and this results in more energy being used to steer it back to synchrony. We also investigate here how the driver node set selected with algorithm ?? performs against other driver sets of the same size. we use a Monte-Carlo sampling of the drivers sets of the same size

as our solution, and filter out the ones that do not meet all the constraints. Figure ?? shows that our solution (blue dotted curve) uses less energy than other driver sets of the same size (red curve).

## VI. CONCLUSION

C'est fini !

## REFERENCES

- [1] S. D. Ramchurn *et al.*, “Putting the ‘smarts’ into the smart grid: A grand challenge for artificial intelligence,” *Commun. ACM*, vol. 55, no. 4, pp. 86–97, Apr. 2012.
- [2] a. J. D. Rathnayaka, V. Potdar, and M. H. Ou, “Prosumer management in socio-technical smart grid,” *Proceedings of the CUBE International Information Technology Conference on - CUBE ’12*, p. 483, 2012. [Online]. Available: <http://dl.acm.org/citation.cfm?doid=2381716.2381808>
- [3] G. Filatrella, a. H. Nielsen, and N. F. Pedersen, “Analysis of a power grid using a Kuramoto-like model,” *The European Physical Journal B*, vol. 61, no. 4, pp. 485–491, mar 2008. [Online]. Available: <http://www.springerlink.com/index/10.1140/epjb/e2008-00098-8>
- [4] Y.-Y. Liu and A.-L. Barabási, “Control Principles of Complex Networks,” pp. 1–55, 2015. [Online]. Available: <http://arxiv.org/abs/1508.05384>
- [5] T. Summers, F. Cortesi, and J. Lygeros, “On Submodularity and Controllability in Complex Dynamical Networks,” *arXiv preprint arXiv:1404.7665*, pp. 1–10, 2014. [Online]. Available: <http://arxiv.org/abs/1404.7665>
- [6] M. Minoux, “Accelerated greedy algorithms for maximizing submodular set functions,” in *Optimization Techniques*, ser. Lecture Notes in Control and Information Sciences, J. Stoer, Ed. Springer Berlin Heidelberg, 1978, vol. 7, pp. 234–243. [Online]. Available: <http://dx.doi.org/10.1007/BFb0006528>
- [7] A. Krause and D. Golovin, “Submodular function maximization,” *Tractability: Practical Approaches to Hard Problems*, vol. 3, pp. 71–104, 2014.
- [8] G. Yan, J. Ren, Y. C. Lai, C. H. Lai, and B. Li, “Controlling complex networks: How much energy is needed?” *Physical Review Letters*, vol. 108, no. 21, pp. 1–5, 2012.