ALGORITHMICALLY INDUCED ARCHITECTURES FOR MULTI-AGENT NETWORKS

A Dissertation
Presented to
The Academic Faculty

By

Thiagarajan Ramachandran

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in
Electrical and Computer Engineering

School of Electrical and Computer Engineering
Georgia Institute of Technology
May 2016

ALGORITHMICALLY INDUCED ARCHITECTURES FOR MULTI-AGENT NETWORKS

Approved by:

Dr Magnus Egerstedt, Advisor
Professor, School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr Yorai Wardi
Professor, School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr Fumin Zhang
Professor, School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr Shabbir Ahmed
Professor, School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr Santiago Grijalva
Associate Professor, School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: May 2016

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr Magnus Egerstedt, for being the best advisor and teacher and supporting me through this harrowing journey. It is highly unlikely that I would survived the past few years without his wisdom and advice. I will be forever grateful to him for letting me be part of his wonderful lab. I would also like to thank the members of his thesis committee, Dr Santiago Grijalva and Dr Fumin Zhang, for reading throug my proposal and the feedback that they provided. I would also like to thank Dr Shabir Ahmed and Dr Yorai Wardi for being part of my committee.

I am also very thankful to the members of the GRITS lab, especially Smriti, Yancy, Sung and JP, for providing moral support and suffering my company over a non-negligible period lasting four years.

Finally, I would like to thank my parents Mr and Mrs Ramachandran and my brother Siva for their patience, support and love. Thank you for everything!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

The objective of this thesis is to understand the interactions between the computational mechanisms, described by algorithms and software, and the physical world, described by differential equations, in the context of networked systems. Such systems can be denoted as cyber-physical nodes connected over a network. In this work, the power grid is used as a guiding example and a rich source of problems which can be generalized to networked cyber-physical systems. We address specific problems that arise in cyber-physical networks due to the presence of a computational network and a physical network as well as provide directions for future research.

CHAPTER 1

INTRODUCTION AND MOTIVATION

1.1 Introduction

The objective of this thesis is to understand the interactions between the computational mechanisms, described by algorithms and software, and the physical world, described by differential equations, in the context of networked systems. Such systems can be denoted as cyber-physical systems with a network. While there is a significant body of work addressing the characterization, control and verification of cyber-physical systems (see [5] and [6] and references herein), the networked nature of these systems has not received much attention. The work presented in this document addresses specific problems that arise due to the networked nature of such systems as well as provides directions for future research.

Within the context of this work, we view networked cyber-physical systems as a collection of spatially seperated entities, hereby referred to as nodes, interacting with each other over a network to make decisions. Each node in such a system is considered to be a computational unit controlling a physical device. This perspective allows us to decompose the system into two distinct but related networks.

The entire system of nodes, viewed as a collection of coupled physical devices, gives rise to a networked dynamical system whose evolution is governed by the laws of physics and is influenced by human designed control laws. The coupling between the physical devices dictates how their states affect each other and gives rise to a physical network.

The computational unit residing in each node is responsible for controlling the state of the underlying physical device and setting long term goals. In order to do so, it needs to acquire measurements and information from other nodes in the network. The information exchange network specifies which nodes can communicate with each other and defines the overall information flow through the network. A physical realization of such a network comprises of sensor suites and communication infrastructure which will enable each node

to communicate with other nodes or just observe their behaviour.

In this thesis, the power grid is used as a guiding example and a rich source of problems which can be generalized to networked cyber-physical systems. In Section 1.2, a literature survey which identifies the connection between the work in this thesis and existing literature on networked controls and smart grids is provided.

In Chapter 2, the connection between information exchange networks and algorithms is explored in the context of the power allocation problem. The information exchange network must possess certain properties to support the data flow required by the problem in hand and the algorithm chosen to solve the problem. Simply put, the information exchange network determines which nodes *can* communicate with each other, but the information requirements of an algorithm specifies which nodes *must* communicate with each other. The work presented in Chapter 2 explores two different algorithms which solves the power allocation problem in a distributed manner and identifies the tradeoffs between information requirements and performance.

In Chapter 3, the effect of communication loss/node failure on the controllability of the power grid is explored (i.e *Can the network survive loss of communication and retain certain properties?*). Two topological tests which provide conditions that need to be met in order to retain controllability under communication loss/node failure is provided.

In Chapter 4, we investigate how imposing time-constraints on the decision making process imposes constraints on the algorithms chosen to solve a task (i.e *How much should agents in a network talk before making a decision?*).The chapter introduces a method to compute a lower bound on the number of iterations required by two different optimization protocols used to compute the stabilizing control action for frequency regulation. It is empirically shown that this number depends on the size of the grid and the specific optimization protocol that is chosen.

In Chapter 5,a stability problem set in a multi-agent setting where the agents lack communication capabilities and have access only to limited state information obtained via sensing is considered. We provide conditions on the system matrices which allow us to stabilize the system in a distributed manner using the information available to them.

It is our intention to develop the work presented in this dissertation into a framework for tackling problems arising in general networked cyber-physical systems.

.

1.2 Literature Survey

The connection between topology and system properties has been explored under the name of networked controls. This survey will provide a representative sample of the literature that explores the connection between topology and networked controls. We provide a brief review of smart grids and prosumer-based power systems as it acts a testbed for our analysis. Finally, we also identify and review literature whch has previously dealt with the notion of information constraints.

1.2.1 Smart Grids and Prosumers

Majority of the work presented in this proposal uses a smart grid model which is populated by producer-consumer hybrids as a test bed. We present a brief survey explaining this setting. In recent years, changes in the power industry have been posing challenges to the power grid. As renewable resources drop in cost and approach price parity with fossil power, intermittent sources will become a larger part of total generation. Additionally, power generation will be more distributed, with residential customers more frequently having generation capacity. As this shift occurs, the lines between producer and consumer become less clear leading to a hybrid prosumer. In the near future, it is expected that any agent on the power grid will be able to have generation capacity, storage capacity, and loads [7, 8]. The agents, being economically motivated entities, would have balance their own personal needs while ensuring the stability of the grid. These entities are dubbed as

prosumers in [9]. The resulting power grid, populated by prosumers, is largely flat in that small prosumers, such as buildings with energy management systems, share the same functionality as large prosumers, such as regional utilities [10]. The prosumer power grid model is a prime example of a cyber-physical network (see [11]), a network of physical devices which are controlled by networked computational entities.

This setting is used in the work presented in Section 2 which considers a power balancing problem in a power grid populated with prosumers. The work done in Chapter 3 aims to explore the controllability problem in the same environment.

1.2.2 Networked Controls and the Graph Laplacian

The graph Laplacian is an important graph-theoritic object which encodes the structure of a given graph. Given a graph $\mathcal{G} = (V, E)$, The graph Laplacian $L_{\mathcal{G}}$ is a matrix which encodes structural and spectral properties of the graph $\mathcal{G}$. It is defined by the equation

$$L_{\mathcal{G}} = D_{\mathcal{G}} - A_{\mathcal{G}}$$

where $D_{\mathcal{G}}$ is the node-edge incidence matrix and $A_{\mathcal{G}}$ is the adjacency matrix of the graph $\mathcal{G}$. The Laplacian and its variants aid in the construction of several coordination protocols (See for e.g [12], [13] and [14]).

An important coordination algorithm whose functioning can be described using the graph Laplacian is the consensus protocol [15]. Consider a group of n agents modelled as a graph $\mathcal{G} = (V, E)$ where V is the vertex set and $(v_i, v_j) \in E$ if and only if the agents represented by v_i and v_j can communicate with each other. Let x_i represent a state, such as spatial location, associated with each agent $v_i \in V$. The consensus algorithm allows the system of agents to converge to the average of their state values. In continuous time, each agent executes the consensus protocol given by

$$\dot{x}_i = -\sum_{j \in N_i} (x_i(t) - x_j(t))$$

where N_i is the neighborhood set of agent $v_i \in V$ (i.e $v_j \in N_i \Leftrightarrow (v_i, v_j) \in E$). The

corresponding state space form of the consensus equation is given by

$$\dot{x} = -L_{\mathcal{G}} x$$

where $L_{\mathcal{G}}$ is the graph Laplacian associated with $\mathcal{G}$.

The topology of the graph $\mathcal{G}$ plays a significant role in the convergence of the consensus protocol. The rate of convergence of the consensus protocol depends on the smallest non-zero eigenvalue of $\mathcal{G}$. The smallest non-zero eigenvalue of the graph Laplacian $L_{\mathcal{G}}$ is known as the Fiedler value or the algebraic connectivity of the graph $\mathcal{G}$ and it measures the connectivity of the graph. It is observed in [16] that more connected the graph $\mathcal{G}$ is, faster the convergence of the consensus protocol. In Section 2.5, we study the connection relationship between information requirements, convergence and task formulation. It can be seen that different formulation of the same task can lead to different information requirements which in turn impacts rate of convergence.

1.2.3 Importance of Topology in Networked Systems

An area of research which highlights the connection between topology and system properties is that of the controllability of leader-follower networks where the follower dynamics are determined by consensus-type local interaction rules. A leader-follower network is constructed by segregating a group of agents, described by a graph $\mathcal{G} = (V, E)$, into leaders and followers. The leaders are subject to external input while the dynamics of the followers are governed by the consensus equation and influenced by the input supplied to the leaders. The dynamics of a leader-follower system is given by

$$\dot{x} = -L_f x(t) - lu(t)$$

where the matrices L_f and l are sub-matrices of the graph Laplacian $L_{\mathcal{G}}$. The controllability of leader-follower systems was first bought up by [17]. The paper characterized the controllabilty of leader follower systems in terms of the eigenvectors of the graph Laplacian and emphasized the need for a graph-theoretic characterization. A topological exploration

of the controllability problem, for the single-leader case, with an emphasis on symmetry and its impact on controllability can be found in [18]. Through the use of equitable partitions, a graph theoretic sufficient condition was presented in [19]. A further investigation, undertaken in [20], uses quotient graphs and bisimulations to show that the uncontrollable part of such leader-follower networks is asymptotically stable for all connected networks. Other results related to leader-follower networks with a single leader is presented in [21]. In case of multiple leaders, a tight lower bound for the rank of the controllability matrix, computed by calculating the distance of each node to the closest leader, is presented in [22] while [23] presents an upper bound.

From a design perspective, [24] presents a method to design completely controllable networks by splicing together smaller controllable networks by connecting the leaders and showing that controllability is preserved. Also, a method for constructing leader-asymmetric networks, a necessary topological condition for controllability, in a decentralized way using graph grammars can be found in [25].

Finally, the topology of the communication/interaction graph also plays a major role on the robustness of consensus/leader-follower based networked systems to noise. It is shown that the robustness of a leader-follower network with a single leader is very sensitive to the communication topology. Specifically, it is shown that the topology which is maximally robust with respect to certain robustness measures is the star topology with the leader acting as the hub node [26]. In the case of a leaderless system executing the consensus protocol, The *Kirchoff index* associated with communication graph can be used as a measure of robustness and can be used to rearrange tree strucutres in order to improve their robustness [27, 28]. Lastly, [29] unifes two different notions of robustness and demonstrates that they can me measured using a common graph invariant, the Kirchoff index.

The work presented in Chapter 3 aims to explore the controllability of a specific networked linear system, a power grid populated by producer-consumer hybrids, in the presence of communication failure and connect it to the underlying topology.

1.2.4 Networked Systems and Information requirements

Networked dynamical systems can be thought of as a group of autonomous agents cooperating with each other over a network in order to achieve a common task. Several examples of such tasks can be found in multi-agent literature (for e.g, see [30],[31] for multi-robot systems, [32], [33] for sensor networks and power systems). The information flow in such a system is dictated by the sensing and communication capabilities of the individual agents. Agents can obtain limited state information via sensors and the existence of a communication network can allow the agents to further coordinate and execute distributed algorithms (for e.g, see [33], [34]). In Section 51, we approach the problem of stabilizing a networked system described by linear dynamics using only sensor measurements without relying on communication. One motivation behind such an undertaking is to understand whether typical tasks like stabilization can be dealt with without relying on expensive communication infrastructure.

One of the idiosyncrasies of a networked dynamical system is the limited predictive power of the individual nodes in the system. This is typically due to the fact that the individual nodes are often seperated spatially and can observe and interact with only a small subset, referred to as its neighborhood, of the system. This imposes an *information constraint* which needs to be taken into account when designing controllers. The problem of stabilizing a power system modelled by linear dynamics under such an information constraint is approached by [33]. The information constraint is addressed by using a model predicitve controller with a prediction horizon set to one (dubbed as *one-step MPC* in [33]). The short prediction horizon allows the nodes to utilize the state information obtained by observing its neighborhood to compute a stabilizing control strategy using optimization methods. It is important to note that the execution of distributed optimization methods requires a communication network as each node has to make multiple information exchanges with its neighbors in order to compute the control signal. This approach was used in [33] in order to compute a stabilizing control strategy for a power system. We adopt the appraoch

of using a 1-step MPC in Section 5.1 but mitigate the need for a communication network in Section 5.2 by using a approximate model which eliminates that need.

Another concept that aids in the design of a feedback controller which respects information constraints is that of quadratic invariance. The information constraint is expressed as a sparsity constraint on the structure of the gain matrix K which defines the feedback law given by $u = -Kx$. It is then shown that it is possible to compute a stabilizing feedback controller using convex programming methods if the constraint set defined by the sparsity requirements is quadratically invariant with respect to the system [35, 36, 37, 38, 39]. The results pertaining to quadratic invariance are fairly general and applies to a wide variety of systems. The downside is that the controller design must be done offline and requires a centralized computation.

In Section 5.3 and Section 5.4, we show that it is possible to design a stabilizing controller in a online fashion for a limited class of systems. The notion of using limited information to solve tasks is also a theme that is explored in robotics under the name of minimality (for e.g, see [40, 41, 42]).

CHAPTER 2

DISTRIBUTED POWER ALLOCATION

In order to underline the connections between information requirements and task formulation, we consider an highly idealized version of the supply and demand problem which commonly arises in a power grid. The setting in which we consider this problem is that of a homogenous power grid populated by producer/consumer hyrbids (referred to as prosumers), a concept introduced in [10]. It is shown that the information requirements for solving this problem depends cruically on the choice of the constraints. We also identify that there exists an trade-off between information requirements and the rate of convergence. For a full account of the presented work, see [43].

2.1 Problem Statement

We consider a set of k prosumer agents, denoted by $V = \{v_1, \ldots, v_k\}$. These agents are assumed to be connected over a transmission network that they use for communications and power exchange. Formally, we represent the network by a connected, undirected graph $\mathcal{G} = (V, E)$ on V with m edges, in which the presence of an edge $(v_i, v_j) \in E$ indicates that agents v_i and v_j can communicate and exchange power bidirectionally. For the purpose of notational ease, we associate an arbitrary orientation to $\mathcal{G}$ which will serve to give meaning to positive and negative flows along edges. Indeed, for each edge $e_i \in E$, we define a *signed power flow* $r_i \in \mathbb{R}$ across e_i, and collect these flows in a vector $r = (r_1, \ldots, r_m)^T \in \mathbb{R}^m$. In the subsequent sections, we will present distributed algorithms that manipulate flow vectors to satisfy the energy needs of the various agents.

We assume that each agent i in the network has computed its desired power need $\mathcal{N}_i \in \mathbb{R}$, by taking into account its personal load, storage and generation capabilities. The value $\mathcal{N}_i$ is negative if agent i desires to generate/produce power and positive if it wants to consume power. Agents with positive $\mathcal{N}_i$ require more power than they produce, and

are requesting power from the network. Likewise, agents with negative $\mathcal{N}_i$ have excess power available, which they are offering for distribution. We collect the power needs of the individual agents in the vector $\mathcal{N} = (\mathcal{N}_1, \ldots, \mathcal{N}_k)^T$.

In a physical power network, the power produced by any node must subsequently be consumed by some other node or nodes. The actual power which is being produced or consumed by an agent is a determined by the amount of power which is being injected and withdrawn from the transmission/distribution infrastructure represented as power flow along the edges. The power $p_i \in \mathbb{R}$ that is actually available to prosumer i is determined by the power flows along edges incident to v_i. Letting $\mathcal{D}(\mathcal{G}) \in \mathbb{R}^{k\times m}$ be the node-edge incidence matrix for $\mathcal{G}$ (given the arbitrary orientation) and given a flow $r \in \mathbb{R}^m$, we define the power vector $p \in \mathbb{R}^k$ by

$$p = \mathcal{D}r \,. \tag{1}$$

The vector r represents the power flows along the edges whose directionality is determined by the incidence matrix $\mathcal{D}$. We would like to compute a power-flow vector r such that the net weighted discrepancy between the power $p = \mathcal{D}r$ and the desired power $\mathcal{N}$ is minimized. This problem can be phrased as least-squares optimization problem,

$$\min_r \frac{1}{2}(\mathcal{D}r - \mathcal{N})^T W(\mathcal{D}r - \mathcal{N}), \tag{2}$$

where $W = W^T > 0$ is a diagonal, positive definite weight matrix.

The interpretation is that we are optimizing over power flows in order to ensure that agents' power needs are satisfied as closely as possible, in a least-squares sense. The weight matrix W captures the relative importance of each agent's need in the network. If agent j is a critical facility (e.g a hospital) , whose power needs are important, then the w_{jj} term is made larger. Also, smaller agents like electric vehicles have smaller tolerances and poor safety mechanisms and cannot handle large power fluctuations. This is also taken into account when assigning weights for the prosumers.

First of all, we immediately see that, as this is a standard linear least-squares problem, the optimal r can be found by computing an appropriate Moore-Penrose psuedo-inverse. Unfortunately, such a computation is inherently centralized and it will not endow the individual prosumers with distributed actions. Instead, one possible way to determine r is to differentiate the cost in (2) with respect to r and perform a (hopefully distributed) gradient descent to find the optimal value of r. But, since the dimension of the r is equal to the number of edges in the network and since the transmission infrastructure is a passive component and does not perform computations, this computation can unfortunately not be distributed among the prosumers in the network.

In the following sections, we present an alternative formulation of the above problem as a constrained optimization problem where the decision variable is actually p, the available power at each individual node, instead of the flows r used in the definition of (2). We will show that we can recover not only the flows r, but also obtain an alternative characterization of the flows as potential differences across adjacent nodes. We also point out the relationship between the structure of the solution and the constraints it imposes on the information topology of the grid. We give a helpful alternate characterization of feasible power flows, in Lemma 1, which is really just a direct consequence of the fundamental theorem of linear algebra:

Lemma 1 *A vector $p \in \mathbb{R}^k$ can be expressed as $p = \mathcal{D}r$ for some $r \in \mathbb{R}^m$, if and only if $\mathbf{1}^T p = 0$, where $\mathbf{1} \triangleq (1, 1, \ldots, 1)^T \in \mathbb{R}^k$.*

Proof: We must show that $\text{range}(\mathcal{D}) \perp \text{span}\,\mathbf{1}$. Since $\text{range}(\mathcal{D})^\perp = \text{null}(\mathcal{D}^T)$ and it is also known that $\text{null}(\mathcal{D}^T) = \text{span}(\mathbf{1})$ for weakly connected, directed networks, we have $\text{range}(\mathcal{D})^\perp = \text{span}(\mathbf{1})$.

■

In short, the requirement $\mathbf{1}^T p = 0$ expresses the conservation law that the power generated in the network equal the power consumed.

2.2 Anticipating Available Power

Using Lemma 1, it is possible to solve for p directly without computing the flow r, as the solution to the minimization problem

$$\begin{aligned} &\min_{p} \frac{1}{2}(p-\mathcal{N})^T W(p-\mathcal{N}) \\ &\text{s.t. } \mathbf{1}^T p = 0 \end{aligned} . \tag{3}$$

By Lemma 1, the constraint $\mathbf{1}^T p = 0$ is equivalent to asserting that p belongs to range($\mathcal{D}$). The Lagrangian for the above problem is given by

$$L(p, \upsilon) = \frac{1}{2}(p-\mathcal{N})^T W(p-\mathcal{N}) + \upsilon \mathbf{1}^T p \tag{4}$$

where $\upsilon \in \mathbb{R}$ being the Lagrange multiplier. The optimal solution is given by $p^* = \mathcal{N} - \upsilon W^{-1}\mathbf{1}$ and $\upsilon^* = \frac{\mathbf{1}^T\mathcal{N}}{\mathbf{1}^T W^{-1}\mathbf{1}}$. The term $\upsilon^* W^{-1}\mathbf{1}$ can be interpreted as the optimal disparity vector as it represents the difference between the power need $\mathcal{N}$ and the optimal power vector p^*.

Let $\mathcal{F} \triangleq \{p \mid p = \mathcal{N} - \alpha W^{-1}\mathbf{1} \text{ for } \alpha \in \mathbb{R}\}$. We can understand the role of the Lagrange multiplier υ by projecting the vectors in $\mathcal{F}$ onto the span($\mathbf{1}$). The projection of a vector p onto span($\mathbf{1}$), derived using normal equations, is given by $\frac{\mathbf{1}^T p}{\mathbf{1}^T\mathbf{1}}$.The projection $\Pi : \mathbb{R} \to \mathbb{R}$ of a vector $p \in \mathcal{F}$ parameterized by α onto span($\mathbf{1}$) is given by the following equation:

$$\Pi(\alpha) = \frac{\mathbf{1}^T}{\mathbf{1}^T\mathbf{1}}\mathcal{N} - \frac{\mathbf{1}^T}{\mathbf{1}^T\mathbf{1}}(\alpha W^{-1}\mathbf{1}) \tag{5}$$

Note that $\Pi(\alpha) = 0$ when $\alpha = \upsilon^* = \frac{\mathbf{1}^T\mathcal{N}}{\mathbf{1}^T W^{-1}\mathbf{1}}$. The Lagrange multiplier υ^* determines the magnitude of the error $\alpha W^{-1}\mathbf{1}$ such that the projection of the error term onto span($\mathbf{1}$) cancels out the projection of the power need $\mathcal{N}$ onto span($\mathbf{1}$) thus rendering p^* orthogonal to span($\mathbf{1}$).

2.3 Constrained Formulation

We reformulate (2) as an equivalent, constrained minimization problem which allows us to generate decentralized control laws depending on the characterization of the constraint $p \in \mathcal{F}$.

Consider the following optimization problem:

$$\begin{aligned} &\min_{p} \frac{1}{2} p^T W p \\ &\text{s.t. } p \in \mathcal{F} \end{aligned} \tag{6}$$

where $\mathcal{D}$ is the incidence matrix. We can show that the optimization problem defined in (6) is equivalent to (3) as follows.

Lemma 2 *Let p^* be the minimizer for (3) and $\tilde{p}^*$ be the minimizer for (6). Then $p^* = \tilde{p}^*$.*

Proof:

Let $p^* = \mathcal{N} - \upsilon^* W^{-1}\mathbf{1}$ where $\upsilon^* = \frac{\mathbf{1}^T\mathcal{N}}{\mathbf{1}^T W^{-1}\mathbf{1}}$ denote the minimizer for (3) and let $h(\alpha) = \mathcal{N} - \alpha W^{-1}\mathbf{1}$. The range of h is the constraint space $\mathcal{F}$ and the tangent to $\mathcal{F}$ at a point $p \in \mathcal{F}$ is given by $\frac{\partial h}{\partial \alpha} = W^{-1}\mathbf{1}$

Since $\tilde{p}^*$ is a minimizer for (6), it lies in $\mathcal{F}$.

Let $J = \frac{1}{2} p^T W p$. From the Karush-Kuhn-Tucker conditions, we know that, at $\tilde{p}^*$, the gradient $\nabla J(\tilde{p}^*)$ can be expressed as a linear combination of the surface normals to the constraint surface $\mathcal{F}$. Therefore, at $\tilde{p}^*$, the gradient $\nabla J(\tilde{p}^*)$ will be orthogonal to the tangent vector to $\mathcal{F}$ at $\tilde{p}^*$. The tangent vector to $\mathcal{F}$, $\frac{\partial h}{\partial \alpha}$ at any point is simply the vector $W^{-1}\mathbf{1}$.Then we have:

$$\nabla J(\tilde{p}^*)^T (W^{-1}\mathbf{1}) = \tilde{p}^{*T}\mathbf{1} = 0 \tag{7}$$

Since $\tilde{p}^*$ lies in $\mathcal{F}$, $\tilde{p}^* = \mathcal{N} - \alpha^* W^{-1}\mathbf{1}$ for some $\alpha^* \in \mathbb{R}$. Substituting $\tilde{p}^* = \mathcal{N} - \alpha^* W^{-1}\mathbf{1}$ in (7),we obtain

$$\tilde{p}^{*T}\mathbf{1} = \mathbf{1}^T\mathcal{N} - \mathbf{1}^T(\alpha^* W^{-1}\mathbf{1}) = 0 \tag{8}$$

and $\alpha^* = \upsilon^* = \frac{\mathbf{1}^T\mathcal{N}}{\mathbf{1}^T W^{-1}\mathbf{1}}$. Therefore, we have $p^* = \tilde{p}^*$. ■

2.4 Flows, Potentials and Lagrange multipliers

We now point out the connection between the unconstrained optimization problem where we optimize over flows r defined by (2) and the constrained optimization problem where we solve for the feasible power p defined by (6). We can recast (6) in a much more concrete fashion by noting that error space span($W^{-1}\mathbf{1}$) is the null space of the matrix $\mathcal{D}^T W$. Then we can rewrite (6) as

$$\begin{aligned} &\min_p \frac{1}{2} p^T W p \\ &\text{s.t. } \mathcal{D}^T W(\mathcal{N} - p) = 0 \end{aligned} \tag{9}$$

whose Lagrangian is given by

$$L(p, \eta) = \frac{1}{2} p^T W p + \eta^T \mathcal{D}^T W(\mathcal{N} - p) \tag{10}$$

where $\eta \in \mathbb{R}^m$ is the Lagrange multiplier. We can obtain p as a function of η by setting $\frac{\partial L}{\partial p}^T = Wp - W\mathcal{D}\eta = 0$. We do so, and obtain $p = \mathcal{D}\eta$. We can then determine the Lagrange dual function by substituting $p = \mathcal{D}\eta$ in (10) and obtain

$$g(\eta) = \inf_p L(p, \eta) = -\frac{1}{2} \eta^T \mathcal{D}^T W \mathcal{D} \eta + \eta^T \mathcal{D}^T W \mathcal{N}. \tag{11}$$

The minimizer for (10) is $p^* = D\eta^*$ where $\eta^* = \underset{\eta}{\text{argmax}}\ g(\eta)$. We can use this fact to define an update law for η by performing gradient ascent on (11) by letting

$$\dot{\eta} = \frac{\partial g}{\partial \eta} = -\mathcal{D}^T W \mathcal{D} \eta + \mathcal{D}^T W \mathcal{N}. \tag{12}$$

The Lagrange multiplier η^* is identical to the flow r^* with respect to the orientation defined by the incidence matrix $\mathcal{D}$, where r^* is the minimizer for the problem defined by (2). The $\mathcal{D}^T W \mathcal{D}$ matrix is called the weighted edge Laplacian. For a more detailed discussion of the weighted edge Laplacian, see [12].

So, the protocol defined by (12) in conjunction with $p = D\eta$, solves the problem defined by (2).

The Lagrange multiplier η determines the feasible power vector p the same way the flow vector r defined in (2) does. Since η is a Lagrange multiplier, its dimension is determined by the dimension of the constraints which is equal to m, where m is the number of edges in the network. As a result, the control law given by (12) becomes a edge oriented control law. We can use this observation to obtain a nodal control law by recasting the above m-dimensional constraint into a n-dimensional one,where n is the number of nodes.

An alternative way to characterize the constraint $p \in \mathcal{F}$ is to use the graph Laplacian $\mathcal{L}$ instead of the transpose of the node-edge incidence matrix $\mathcal{D}$. The graph Laplacian shares the same nullspace with $\mathcal{D}^{\mathcal{T}}$; i.e null$(\mathcal{D}^T)$ = null$(\mathcal{L})$ = $\mathbf{1}$. So, we can use this fact to reformulate (6) as follows:

$$\begin{aligned} &\min_{p} \frac{1}{2} p^T W p \\ &\text{s.t. } \mathcal{L}W(\mathcal{N} - p) = 0 \end{aligned} \tag{13}$$

whose Lagrangian is then

$$L(p, q) = \frac{1}{2} p^T W p + q^T \mathcal{L} W(\mathcal{N} - p) \tag{14}$$

where $q \in \mathbb{R}^n$ is the Lagrange multiplier. Let g be the Lagrange dual function associated with the Lagrangian given by (14). We can then proceed to solve this problem in a similar fashion to (9) and obtain $p = \mathcal{L}q$ and an update law for q given by

$$\dot{q} = \frac{\partial g}{\partial q} = -\mathcal{L}W\mathcal{L}q + \mathcal{L}W\mathcal{N}. \tag{15}$$

The Lagrange multiplier q^* here provides an alternate way to define the flow r^* as the potential difference between the nodes, where the role of the potentials is played by the Lagrange multiplier q^*. The flows are then defined as $r^* = \mathcal{D}^T q^*$ and since the feasible power at each node is defined as $p^* = \mathcal{D}r^*$, we have $p^* = \mathcal{D}\mathcal{D}^T q^* = \mathcal{L}q^*$.

2.5 Information requirements

In this section, we will identify the information topology which must be supported by the prosumer network to execute the gradient ascent protocols defined by (12) and (15).

Since the Lagrange multiplier q lies in $\mathbb{R}^n$, where n is the number of prosumers in the network, we can imagine prosumer i to be responsbile for updating the value q_i. We can then infer the information required by prosumer i to perform the q-update by inspecting the protocol given by (15). Then, the $\mathcal{L}W\mathcal{L}q$ term implies that agent i needs to keep track of the potentials q_j such that $j \in cl(N_i)$ where $cl(N_i)$ denotes the closure of the neighbourhood set of i. Therefore protocol (15) requires each prosumer to keep track of the q-values of its neighbours and the q-values of neighbours-of-its-neighbours (i.e neighbours that are 2-hops away or less).

To identify the information required by an individual prosumer i to execute the edge-oriented update law given by (12), we need to recast it into a nodal update law.

Define node-potentials to be a vector $q \in \mathbb{R}^n$. Then the flow η induced by the potential q is $\mathcal{D}^T q$. To obtain a update law for q, we note that $\dot{\eta} = \mathcal{D}^T \dot{q}$. Since (12) already defines an update law for the flows η, we have

$$\dot{\eta} = \mathcal{D}^T \dot{q} = -\mathcal{D}^T(W\mathcal{D}\eta + W\mathcal{N}) = -\mathcal{D}^T(W\mathcal{D}\mathcal{D}^T q + W\mathcal{N}) \tag{16}$$

This then gives us a update law for q as follows:

$$\dot{q} = W(-\mathcal{L}q + \mathcal{N}) \tag{17}$$

Clearly, we have $\dot{q} = 0$ when $\mathcal{L}q = \mathcal{N}$. The above update law defined for q is not always stable as $\mathcal{N}$ need not lie in the **range** $\mathcal{L}$. We can achive stabilization by simply projecting $\mathcal{N}$ into the **range** $\mathcal{L}$ as follows:

$$\begin{aligned} &\dot{q} = W(-\mathcal{L}q + \mathcal{N} - s) \\ &\text{where } \dot{s} = -\mathcal{L}Ws \text{ and } s(0) = \mathcal{N} \end{aligned} \tag{18}$$

The update on $s \in \mathbb{R}^N$ is defined such that s converges to $\alpha W^{-1}\mathbf{1}$ where $\alpha = \frac{\mathbf{1}^T\mathcal{N}}{\mathbf{1}^T W^{-1}\mathbf{1}}$. For more details on the convergence properties of ODEs involving graph Laplacian, see [44]. So, when $\dot{q} = 0$, we obtain $p = \mathcal{L}q = \mathcal{N} - \alpha W^{-1}\mathbf{1}$ as expected.

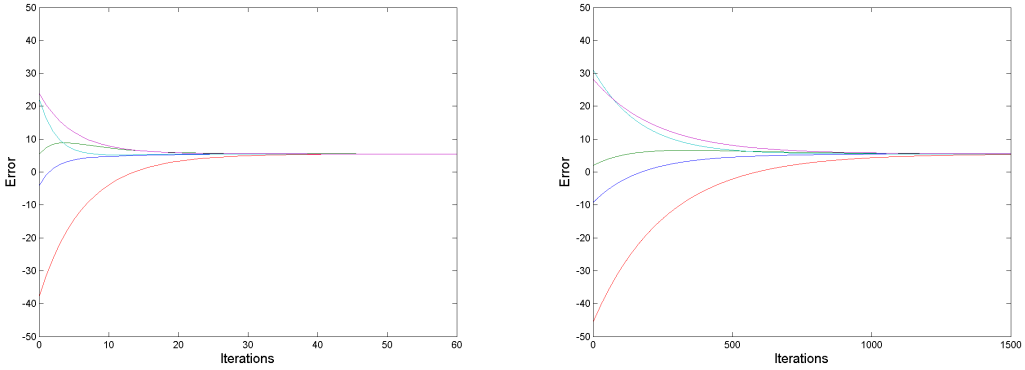
We can now determine the information required by individual prosumers by inspecting (18). Both the q-update law and the s-update law requires only that we know the state pair

(q_i, s_i) of the neighbouring prosumers (encoded by the $\mathcal{L}$ term). So, the update law given by (15) requires a one-hop information network as opposed to (15), but it requires that each prosumer maintain two states, s and q, instead of one.

Also, the update law (15) typically converges to the optimal solution faster than (18). This is because the convergence rate of (15) is proportional by λ_2^2 and the convergence of (18) is proportional by λ_2 where λ_2 is the second smallest eigenvalue of the Laplacian matrix.

2.6 Example - A 100 Prosumer system

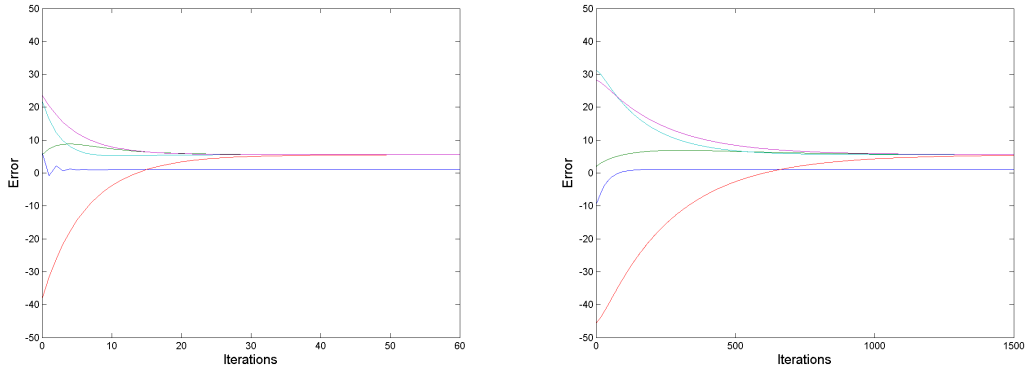
We compare the performance of the controllers given by (15) and (18) on a randomly generated prosumer network with two different weight functions.



(a) (b)

Figure 1: 1a tracks the error $\mathcal{N} - \mathcal{L}q_1$ while 1b tracks the error $\mathcal{N} - \mathcal{L}q_2$. q_1 is updated according to (15) while q_2 is updated according to (18). The weights are chosen to be the identity matrix.

A prosumer graph $\mathcal{G}$ with 100 prosumers was generated randomly using a Erdos-Renyi random graph model and the corresponding graph Laplacian $\mathcal{L}$ was constructed. The smallest non-zero eigen value of the generated graph, which determines the rate of convergence, was 34.6389. The power need vector $\mathcal{N}$ was also generated randomly with values ranging in between -70 to 70 watts. The average surplus in the network, given by $\frac{1}{100}\mathbf{1}^T\mathcal{N}$, is 5.5173 . The potentials q_1 and q_2 were randomly initialized and are modified according to



(a) Error convergence at nodes 1 through 5 using update law (15)

(b) Error convergence at nodes 1 through 5 using update law (18)

Figure 2: 2a tracks the error $\mathcal{N} - \mathcal{L}q_1$ while 2b tracks the error $\mathcal{N} - \mathcal{L}q_2$. q_1 is updated according to (15) while q_2 is updated according to (18). Agent 1(blue) suffers lower error than the others as it was weighted much higher than the other agents

the update law given by (15) and (18), respectively.

The weight function W_1 is the identity matrix $I_{100} \in \mathbb{R}^{100\times100}$. The power p converges to $\mathcal{N} - \alpha\mathbf{1}$ where $\alpha = \frac{1}{100}\mathbf{1}^T\mathcal{N} = 5.5173$. Figure 1a tracks the evolution of the error $\mathcal{N} - Lq_1$ for the first five nodes where q_1 is updated according to (18) while Figure 1b tracks $\mathcal{N} - Lq_2$ for the same set of nodes, where q_2 is updated according to (15). Note that the error does converge to 5.5173, average of the power needs $\mathcal{N}$. Also, the rate of the convergence of the update law (15) is much higher than that of the update law given by (18). This is due to the fact that the convergence rate of (15) is proportional to the square of the smallest non-zero eigen value of the graph Laplacian $\mathcal{L}$, due to the $\mathcal{L}W\mathcal{L}$ term. The rate of convergence of the update law given by (18) is determined by the smallest non-zero eigen value of $\mathcal{L}$ which is approximately 1120. As a result Lq_1 converges to the minimum in less that 50 iterations while Lq_2 takes almost 1500 iterations to converge.

The weight function W_2 is the matrix where $W_{2ii} = 1$ if $i \neq 1$ and $W_{211} = 5$. All the other entries, W_{2ij} where $i \neq j$, is set to 0. In short, prosumer 1 is weighted in such a fashion that it is considered 5 times more important the rest of the prosumers and will suffer 5 times less error than the others. Figure 2a tracks $\mathcal{N} - Lq_1$ where q_1 is updated according to (15)

while Figure 2b tracks $\mathcal{N} - Lq_2$ where q_2 is updated according to (15). The error suffered by prosumer 1(blue) is just 1.1125, while the other prosumers bear $5 \times 1.1125 = 5.5623$. The performance of both the update law is similar to the unweighted case for the same reasons.

2.7 Summary

In this chapter, The distributed power allocation problem was used to explore the connection between task formulation and information requirements in a producer-consumer (prosumers) hybrid environment. It was shown that different solutions have implications for the information exchange network that needs to accompany the underlying physical network. In the next chapter, we will further explore the connection between information exchange networks and physical networks in the same prosumer environment by exploring how loss of contact with a subset of nodes in the information exchange network can impact the controllability of the system.

CHAPTER 3

CONTROLLABILITY AND COMMUNICATION FAILURE

Networked dynamical systems rely on communication in order to coordinate and compute appropriate control actions. Loss of communication links can exclude key decision makers from providing input and can even alter the system properties. The work presented in this chapter explores the impact of communication loss on the controllability of a specific networked system, a homogeneous power-grid populated by producer-consumer hybrids. We provide results which relates the controllability of such a system with mute nodes to the topology of the underlying electrical network and show that under certain topological conditions, controllability is preserved. For more details, see [45].

3.1 Grid model

We briefly describe the power grid model that we shall be using throughout this Chapter before addressing the issue of communication failure and muteness.

We consider a set $V = \{1, 2 \dots n\}$ of n prosumers (introduced in Chapter 2) which are connected to each other electrically via a transimission line. Formally, we can represent the physical layout of this power network by a graph $\mathcal{G}_p = (V, E)$ where $|V| = n$. The presence of an edge (v_i, v_j) indicates that the nodes v_i and v_j are physically coupled. There is a state $x_i \in \mathbb{R}$, the deviation in output power with respect to a scheduled reference, associated with each node v_i in V. We collect the states in a vector associated with each node v_i to obtain the ensemble state given by $x = [x_1, x_2 \dots x_n]^T$. The evolution of x with respect to a discrete time parameter $k \in \mathbb{N}$ is given by the following dynamical model:

$$x(k+1) = A_p x(k) + B_p u(k) \tag{19}$$

where $u(k)$ is a vector of setpoints which controls the power output of each agent at time k. Furthermore,

$$A_p = I - T_s JS \tag{20}$$

$$B_p = T_s J\beta \tag{21}$$

where $J \in \mathbb{R}^{n\times n}$ is called the Jacobian matrix, I is the n-dimensional identity matrix, $T_s \in \mathbb{R}$ is the sampling time, S and β are diagonal matrices of dimension $n \times n$ which encodes certain electrical properties of the each individual agent. Note that A_p and B_p are derived from the Jacobian matrix J. This special structure inherent in the power grid model is what allows us to connect controllability to the grid's electrical layout. A much more detailed discussion of this model can be found in [33].

The Jacobian matrix has a sparsity structure which reflects the underlying electrical topology of the network. This is captured by the following relation

$$J_{ij} = 0 \Leftrightarrow (v_i, v_j) \notin E. \tag{22}$$

Note that both A_p and B_p share the same sparsity structure with J as the algebraic operations carried out to obtain them, multiplication by the diagonal matrices, S and β and subtraction with the identity matrix I, do not affect the sparsity structure. Furthermore, the Jacobian, for power systems, is invertible and posseses full rank (For more details, see section 3-D in [33]) . This allows us to establish that the pair (A_p, B_p) is completely controllable.

We also assume that the agents can communicate with each other over a communication network which is represented by a graph $\mathcal{G}_c = (V, E) = \mathcal{G}_p$. The equivalence between the physical and communication graphs imply that the nodes which are connected physically can communicate with each other.

3.2 Muteness

The work done in [33] provides a distributed method to stabilize the system described by (19) in a distributed manner where the agents iterate over control strategies by exchanging

information over the communication network to obtain an optimal stabilizing solution. As a result, any communication failure which isolates an agent from the rest of the system means that the isolated agent cannot participate in the decision making process and is forced to arrive at a control strategy without any information.

Since the isolated agent is still connected to the network physically, and is influenced by the states of the other agents, one approach would be to set its input to zero and relinquish the burden of stablizing the system to the rest of the agents. Consequently, it is important to understand how this loss of actuation affects the controllability of the system. In this section, we introduce the formal notion of muteness in order to discuss the system resulting from the isolated agents not participating in the decision making process.

Let the set $\mathcal{M} \subset V$ denote the set of agents which are isolated due to communication failure. We will refer to the agents belonging to the set $\mathcal{M}$ as **mute**. We assume that each agent in the set $\mathcal{M}$ is unable to communicate with the rest of the network and as a result adopts a "zero-bias" control strategy. The following equation summarizes this control strategy:

$$v_i \in \mathcal{M} \Leftrightarrow u_i(k) = 0 \; \forall k \in \mathbb{N} \tag{23}$$

For convenience, we will assume that $\mathcal{M}$ is given by the last m agents i.e, $\mathcal{M} = \{n - m + 1, n - m + 2, \dots n\}$ where $m = |\mathcal{M}|$ in order to simplify analysis.After substituting the control strategy adopted by the mute agents given by (23) in the dynamical model (19), we obtain

$$x(k+1) = A_p x(k) + \begin{bmatrix} B_{\mathcal{N}} & B_{\mathcal{M}} \end{bmatrix} \begin{bmatrix} u_{\mathcal{N}}(k) \\ 0_{m\times 1} \end{bmatrix} \tag{24}$$

where $B_{\mathcal{N}}$ is the matrix comprising of the first $n - m$ columns of the B_p, the $B_{\mathcal{M}}$ contains the rest of the columns of B_p which corresponds to the mute agents. Also, $u_{\mathcal{N}} = [u_1, u_2 \dots u_{n-m}]^T$ and $0_{m\times 1}$ is a column vector with m rows containing zeros. Since the inputs corresponding to the columns of $B_{\mathcal{M}}$ is zero, we can simplify the dynamics further and obtain

$$x(k+1) = A_p x(k) + B_{\mathcal{N}} u_{\mathcal{N}}(k). \tag{25}$$

The dynamics given by (25) represents the evolution of the power grid when the mute agents do not participate in the decision making process. It is the controllability of this system that we are interested in.

3.3 Controllability result

In this section, we will provide a rank test which allows us to establish the controllability of the system described by (25).

The controllability matrix of the the system defined by (25) is expressed as

$$\Gamma = \begin{bmatrix} B_{\mathcal{N}} & A_p B_{\mathcal{N}} & \dots & A_p^{n-1} B_{\mathcal{N}} \end{bmatrix}. \tag{26}$$

Furthermore, we can define the reduced controllability matrix as follows

$$\hat{\Gamma} = \begin{bmatrix} B_{\mathcal{N}} & A_p B_{\mathcal{N}} & \dots & A_p^{m} B_{\mathcal{N}} \end{bmatrix}, \tag{27}$$

where $m = |\mathcal{M}|$ is the number of mute prosumers.

We present a lemma which shows that it is enough to check the rank of $\hat{\Gamma}$ in order to establish controllability of the system described by (25).

Lemma 3 *Let Γ and $\hat{\Gamma}$ be as defined by Equations (26) and (27). Then $\rho(\Gamma) = \rho(\hat{\Gamma})$ where ρ is the rank operator.*

Proof: We begin by defining

$$\Gamma_k = \begin{bmatrix} A_p^0 B_{\mathcal{N}} & A_p^1 B_{\mathcal{N}} & \dots & A_p^k B_{\mathcal{N}} \end{bmatrix} \tag{28}$$

It can be easily seen that $\rho(\Gamma_{r+1}) \geq \rho(\Gamma_r)$. This is because Γ_{r+1} is constructed by adding more columns to Γ_r and adding more columns does not decrease the rank. Now, we claim

that if $\rho(\Gamma_{r+1}) = \rho(\Gamma_r)$ for some r, then $\rho(\Gamma_{r+n}) = \rho(\Gamma_r)$ for all n. This can be established via induction.

Let $\rho(\Gamma_{r+1}) = \rho(\Gamma_r)$ and let $h = 1$. We will now establish that $\rho(\Gamma_{r+2}) = \rho(\Gamma_r)$. Then,

$$\Gamma_{r+1+h} = \Gamma_{r+2} = \begin{bmatrix} \Gamma_r & A_p^{r+1}B_{\mathcal{N}} & A_p^{r+2}B_{\mathcal{N}} \end{bmatrix}. \tag{29}$$

Now, let v be a column of the matrix $A_p^{r+2}B_{\mathcal{N}}$. Then $v = A_p u$ where u is a column of the matrix $A_p^{r+1}B_{\mathcal{N}}$. Every column of the matrix $A_p^{r+1}B_{\mathcal{N}}$, which represents the last set of columns in the matrix Γ_{r+1}, can be written as a linear combination of the columns of the matrix Γ_r owing to the fact $\rho(\Gamma_{r+1}) = \rho(\Gamma_r)$. This implies that $u = \sum_{i=0}^{r} A_p^i B_{\mathcal{N}} v_i$ for some choice of vectors $v_i \in \mathbb{R}^n$. Then

$$v = A_p \sum_{i=0}^{r} A_p^i B_{\mathcal{N}} v_i = \sum_{i=1}^{r+1} A_p^i B_{\mathcal{N}} v_i \tag{30}$$

$$\implies v = \underbrace{\sum_{i=1}^{r} A_p^i B_{\mathcal{N}} v_i}_{\text{Linear combinations of columns of } \Gamma_r} + A_p^{r+1} B_{\mathcal{N}} v_r \tag{31}$$

Since every column of $A_p^{r+1}B_{\mathcal{N}}$ is a linear combination of the columns of the matrix Γ_r, we can conclude the v, a column of the matrix $A_p^{r+2}B_{\mathcal{N}}$ is a linear combination of columns of the matrix Γ_r. This establishes that every column of the matrix Γ_{r+2} is a linear combination of columns of Γ_r allowing us to conclude that $\rho(\Gamma_{r+2}) = \rho(\Gamma_r)$.

Now, assume $\rho(\Gamma_{r+h}) = \rho(\Gamma_r)$ for some $h > 1$. We will proceed to show that $\rho(\Gamma_{r+h+1}) = \rho(\Gamma_r)$. Note that $\rho(\Gamma_{r+h}) = \rho(\Gamma_r)$ for some $h > 1$ implies that $\rho(\Gamma_{r+h-1}) = \rho(\Gamma_r)$. This can be established by the following argument. We know that $r + h - 1$ is bounded by $r + 1$ as $h > 1$. Then since, the rank operator ρ is monotonic, we have

$$\rho(\Gamma_r) = \rho(\Gamma_{r+1}) \leq \rho(\Gamma_{r+h-1}) \leq \rho(\Gamma_{r+h}) = \rho(\Gamma_r) \tag{32}$$

allowing us to infer that $\rho(\Gamma_{r+h}) = \rho(\Gamma_{r+h-1}) = \rho(\Gamma_r)$. Now, setting $c = r + h - 1$, we have $\rho(\Gamma_c) = \rho(\Gamma_{c+1})$. We can apply the same argument which we used above to establish that $\rho(\Gamma_c) = \rho(\Gamma_{c+2})$ which was to be shown. This combined with base case argument shows that if $\rho(\Gamma_{r+1}) = \rho(\Gamma_r)$ for some r, then $\rho(\Gamma_{r+n}) = \rho(\Gamma_r)$ for all n.

Now, we know that $B_p = T_s J\beta$ is $n \times n$ matrix of rank n as it is a product of two full rank matrices, J and β. This implies all its columns are linearly independent. Then the rank of $B_{\mathcal{N}}$, comprised of the first $n - m$ columns, is $n - m$. Then, $\rho(\Gamma_0) = \rho(B_{\mathcal{N}}) = n - m$. Now, if the $\rho(\Gamma_m) = \rho(\hat{\Gamma}) < n$, then there exists $k < m$ such that $\rho(\Gamma_k) = \rho(\Gamma_{k+1})$. Then, we can establish that $\rho(\Gamma_k) = \rho(\Gamma_m) = \rho(\hat{\Gamma}) = \rho(\Gamma)$ from our previous argument.

If $\rho(\hat{\Gamma}) = n$, then clearly $\rho(\Gamma) = n$, as n is the upper bound for the rank of the controllability matrix. This lets us conclude that $\rho(\hat{\Gamma}) = \rho(\Gamma)$. ■

The above lemma simplifies the problem by allowing us to truncate the controllability matrix and this would allow us to discard a lot of columns if the number of agents n is much greater relative to the number of mute agents m. We can further simplify the problem and draw connections to the underlying physical topology of the system by further exploiting the strucuture of the matrices A_p and B_p.

We recall from Section 2 that $A_p = I - T_p JS$ and $B_p = T_s J\beta$. We can then write $B_{\mathcal{N}} = T_s J\hat{\beta}$ where $\hat{\beta} = [b_1, b_2 \dots b_{n-m}] \in \mathbb{R}^{n \times (n-m)}$ where b_i is the i'th column of the matrix β and $m = |\mathcal{M}|$ is the number of mute agents. In order to proceed, we define a matrix P_m as follows:

$$P_m = \begin{bmatrix} SJ\hat{\beta} & \dots & S(JS)^{m-1}J\hat{\beta} \end{bmatrix} \tag{33}$$

The structure of the P_m contains information about the underlying physical graph $\mathcal{G}_p$ and its higher powers. This will later allow us to connect the controllability of the power network to its underlying physical topology. For now, we express P_m as the following block matrix

$$P_m = \begin{bmatrix} G \\ F \end{bmatrix}, \tag{34}$$

where G is a matrix of dimension $(n - m) \times m(n - m)$ and F is a matrix of dimension $m \times m(n - m)$. The matrix F is of quite some importance as it captures the interaction between the mute nodes $\mathcal{M}$ and the non-mute nodes contained in the set $V \setminus \mathcal{M}$.

Theorem 1 *The pair (A_p, B_N) is completely controllable if the rank of the matrix F is equal to m, where m is the number of mute agents.*

Proof: We know that the rank of the controllability matrix Γ is equal to the rank of $\hat{\Gamma}$ from Lemma 1. We also know that $B_p = T_s J\beta$. Then $B_N = T_s J\hat{\beta}$ where $\hat{\beta} = [b_1, b_2 \dots b_{n-m}] \in \mathbb{R}^{n\times(n-m)}$ where b_i is the i'th column of the matrix β.

The sampling time T_s does not affect the rank analysis. So, we set $T_s = 1$ in the following derivation to simplify analysis. Now, we have

$$\rho(\hat{\Gamma}) = \rho(\begin{bmatrix} J\hat{\beta} & (I - JS)J\hat{\beta} & \dots & (I - JS)^m J\hat{\beta} \end{bmatrix})$$

We can expand the terms $(I - JS)^r$ using binomial expansion and obtain

$$\rho(\hat{\Gamma}) = \rho(\begin{bmatrix} J\hat{\beta} & \dots & \sum_{k=0}^{m} c_k^n (-JS)^k J\hat{\beta} \end{bmatrix})$$

where $c_n^k = \frac{n!}{k!(n-k)!}$. Since the coefficients do not contribute the rank analysis, we can drop them and obtain a further simplified expression as follows:

$$\rho(\hat{\Gamma}) = \rho(\begin{bmatrix} J\hat{\beta} & \sum_{k=0}^{1}(-JS)^k J\hat{\beta} & \dots & \sum_{k=0}^{m}(-JS)^k J\hat{\beta} \end{bmatrix}).$$

Note that the first term of the sum $\sum_{k=0}^{r}(-JS)^k J\hat{\beta}$ is always equal to $J\hat{\beta}$. This allows us to drop that term as it is also equal to the first set of columns of the matrix $\hat{\Gamma}$. So, we then obtain

$$\rho(\hat{\Gamma}) = \rho(\begin{bmatrix} J\hat{\beta} & \dots & \sum_{k=1}^{m}(-JS)^k J\hat{\beta} \end{bmatrix}).$$

We can see that the sum $\sum_{k=1}^{r+1}(-JS)^k J\hat{\beta}$ can expressed as the sum of $\sum_{k=1}^{r}(-JS)^k J\hat{\beta}$ and $(JS)^{r+1} J\hat{\beta}$. This allows us to eliminate the summation and only retain the last term further simplifying the expression for the rank as follows:

$$\rho(\hat{\Gamma}) = \rho(\begin{bmatrix} J\hat{\beta} & (-JS)^1 J\hat{\beta} & \dots & (-JS)^m J\hat{\beta} \end{bmatrix}).$$

Finally, we can drop the negative signs (as they do not have an impact on the rank of a matrix) and factor out the Jacobian to obtain

$$\rho(\hat{\Gamma}) = \rho(J\begin{bmatrix}\hat{\beta} & S J\hat{\beta} & \dots & S(JS)^{m-1}J\hat{\beta}\end{bmatrix}).$$

Since, the Jacobian J is a full rank matrix, it does not reduce the rank of the controllability matrix. So the rank is purely determined by the second term of the product. That is

$$\rho(\hat{\Gamma}) = \rho(\begin{bmatrix}\hat{\beta} & S J\hat{\beta} & \dots & S(JS)^{m-1}J\hat{\beta}\end{bmatrix}).$$

We set

$$M = \begin{bmatrix}\hat{\beta} & S J\hat{\beta} & \dots & S(JS)^{m-1}J\hat{\beta}\end{bmatrix}. \tag{35}$$

Note that $\hat{\beta}$ is a truncated diagonal matrix of the form

$$\hat{\beta} = \begin{bmatrix} D \\ 0_{m\times(n-m)} \end{bmatrix}$$

where D is a diagonal matrix of dimension $(n-m)\times(n-m)$. We can use this to express M as a block matrix of the form

$$M = \begin{bmatrix} D & G \\ 0_{m\times(n-m)} & F \end{bmatrix}.$$

where G is a matrix of dimension $(n-m)\times m(n-m)$ and F is a matrix of dimension $m \times m(n-m)$.

Since D is a diagonal matrix, we can take linear combinations of the its columns to eliminate the entries of the matrix G. This allows us to conclude that

$$\rho(M) = \rho(\begin{bmatrix} D & 0_{(n-m)\times m(n-m)} \\ 0_{m\times(n-m)} & F \end{bmatrix}) \tag{36}$$

$$\implies \rho(M) = \rho(D) + \rho(F) = (n-m) + \rho(F). \tag{37}$$

In order for the system to be completely controllable, we require $\rho(F) = m$ which was to be shown. ■

Theorem 1 provides us with a rank test as opposed to topological. One of the primary advantages of a topological characterization as opposed to a rank test is that it aids in the design of the network topology and is therefore of interest. In the next section, we will use the results of Theorem 1 to connect the controllability of (25) to the topology given by $\mathcal{G}_p$.

3.4 Controllability and Topology

The rank test provided by Theorem 2 involves inspecting the matrix P_m defined by (33). The matrix P_m possess a rich topological structure which encodes information about the physical network G_p and its higher graph powers. In this section, we establish controllability by extracting specific linear submatrices of the matrix P_m and interpret the results from a graph-theoritic viewpoint.

We will seperate our analysis into two cases : $|\mathcal{M}| = 1$ and $|\mathcal{M}| > 1$.

When there is a single mute agent (i.e $|\mathcal{M}| = 1$), we can show that controllablity of (25) can be directly related to the connectivity of the physical network represented by the graph $\mathcal{G}_p$.

Theorem 2 *If the graph G_p is strongly connected and $|\mathcal{M}| = 1$, then the pair (A_p, B_N) is always completely controllable.*

Proof: When the number of mute agents is equal to 1, we can write the matrix P_1 defined by (33) as follows:

$$P_1 = S J\hat{\beta} = \begin{bmatrix} G \\ F \end{bmatrix} \tag{38}$$

where F is a $1 \times (n-1)$ matrix. Note that F is just the last row of the matrix $S J\hat{\beta}$. Since both $\hat{\beta}$ and S are diagonal matrices, they do not affect the sparsity structure of the product $P_1 = S J\hat{\beta}$. So, P_1 inherits its sparsity structure from that of J. Let $u = n$ denote the single element of the set $\mathcal{M}$. Since $\mathcal{G}_p$ is strongly connected, there exists atleast one node v in $V \setminus \mathcal{M}$ such that $(v, u) \in E$. This implies that $\alpha \neq 0$ where α is the element in the v'th position in the vector F. Since, F is a row vector with a non-zero entry α, we can conclude

that the rank of F is equal to 1 which is the number of mute agents in the system. This allows us (by Theorem 1) to conclude that the pair $(A_p, B_{\mathcal{N}})$ is completely controllable. ■

So, as long as the isolated node is connected to the system electrically, we can use the other nodes to control the state of the isolated node irrespective of the node's position in the network topology. This shows that the pair $(A_p, B_{\mathcal{N}})$ is always controllable, irrespective of the network topology, when only a single node suffers from communication failure.

For the case $|\mathcal{M}| > 1$, we provide a sufficient condtion under which controllability is preserved. In order to do so, we define the set $\mathcal{N} = V \setminus \mathcal{M}$ where V is set of all agents and $\mathcal{M}$ is the set of mute agents in the network. In the following theorem, we identify topological conditions on the set $\mathcal{M}$ which renders the pair $(A_p, B_{\mathcal{N}})$ controllable.

Theorem 3 *Let $G_p = (V, E)$ be the graph representing the electrical network. Let $\mathcal{M} \subset V$ be the set of mute agents. If there exists an injective map $\phi : \mathcal{M} \to \mathcal{N}$ such that*

$$\phi(m) = n \Leftrightarrow (m, n) \in E \wedge (v, n) \notin E \ \forall v \in \mathcal{M} \setminus \{m\}, \tag{39}$$

then the pair $(A_p, B_{\mathcal{N}})$ is completely controllable.

Proof: Let $M \subset V$ be a set of mute agents. Assume that there exists $\phi : \mathcal{M} \to \mathcal{N}$ which satisfies the condition given by (39). Physically speaking, the existence of the map ϕ implies that every mute node $v \in \mathcal{M}$ is electrically connected non-mute node $\phi(v) = m$ which is not connected to any node in the set $\mathcal{M} \setminus \{v\}$.

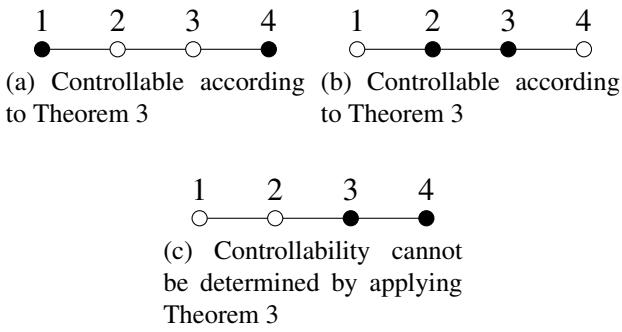
Once again, we restrict our attention to matrix $\hat{P}_m = S J\hat{\beta}$ and express it as

$$\hat{P}_m = S J\hat{\beta} = \begin{bmatrix} \hat{G} \\ \hat{F} \end{bmatrix} \tag{40}$$

where $\hat{F} \in \mathbb{R}^{|\mathcal{M}| \times |\mathcal{N}|}$.

As in the case of the single mute agent case, the matrix $\hat{F}$ encodes the relationship between mute nodes and non-mute nodes. Allowing $\hat{F}_{(i,j)}$ to stand for the element located along the ith row and the jth column of the matrix $\hat{F}$, we can say that

$$\hat{F}_{(i,j)} = 0 \Leftrightarrow (i, j) \notin E \wedge (i \in \mathcal{M}) \wedge (j \in \mathcal{N}). \tag{41}$$



(a) Controllable according to Theorem 3

(b) Controllable according to Theorem 3

(c) Controllability cannot be determined by applying Theorem 3

Figure 3: Three different configurations of a line graph with different nodes muted (represented by white nodes).

Since ϕ satisfies the condtion (39), for every mute node $m \in \mathcal{M}$, there exists a node $n = \phi(m) \in \mathcal{N}$ such that the column $\hat{F}_{\phi(m)}$ contains zero at all locations except $\hat{F}_{m,\phi(m)}$. Then, the collection of columns $\{F_{\phi(m)} \mid m \in \mathcal{M}\}$ are all mutually orthogonal and therefore linearly independent. Therefore, the matrix $\hat{F}$ contains $|\mathcal{M}|$ linearly independent columns and the rank of $\hat{F}$ is equal to $m = |\mathcal{M}|$. Writing the matrix P_m as defined by (33) as

$$P_m = \begin{bmatrix} S J\hat{\beta} & \dots & S(JS)^{m-1} J\hat{\beta} \end{bmatrix} = \begin{bmatrix} \hat{G} & \tilde{G} \\ \hat{F} & \tilde{F} \end{bmatrix} \tag{42}$$

we can see that the rank $\rho(F) = \rho(\begin{bmatrix} \hat{F} & \tilde{F} \end{bmatrix}) = m$. Appealing to Theorem 1, we can conclude that the pair $(A_p, B_{\mathcal{N}})$ is completely controllable. ■

Theorem 2 and Theorem 3 are results which connect the topology of the physical network to the controllability of the underlying system. While the rank tests establish in Section 4 are more definitive tests for controllability, the topological tests established in this section can be a valuable aid when it comes to designing network topologies as they can be used to identify problematic node configurations and restructure them so that the system is more controllable.

In the next section, we consider different examples of network topologies and apply our results to them in order to determine the controllability of a power grid with that physical topology.

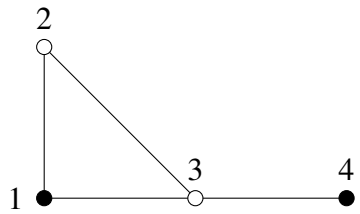


Figure 4: The controllability of this configuration cannot be detected by applying Theorem 3, but can be inferred from theorem 1

3.5 Examples

We consider different network topologies and present a brief controllability analysis for each one of them using the results derivied in Sections 5 and 4.

3.5.1 A path graph with 4 nodes

The physical topology under consideration is a line graph with four nodes. Figure 3 displays 3 different configurations of line graphs to which we can apply our results. The nodes marked in white in the figures are the muted nodes and the black ones are the unmuted nodes.

In the configuration shown in Figure 3a, we can define a map ϕ as follows:

$$\phi(2) = 1$$

$$\phi(3) = 4$$

The above map satisfies the requirements given by (39) allowing us to apply Theorem 3 and infer that the power grid model with the configuration given 3a is controllable. We can also define a similar ϕ for the configuration given in 3b by mapping the node 1 to 2 and node 4 to 3.

It turns out that there exists no ϕ which satisfies the requirement (39) for configuration 3c. We cannot apply Theorem 3 in this situation. But, it turns out that the configuration given by 3c is actually controllable. This can be seen by computing the higher powers of the matrix P_m given by equation 33 and applying Theorem 1.

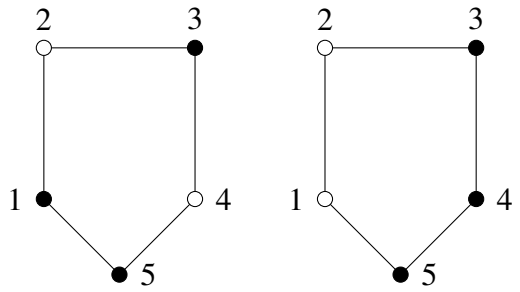


Figure 5: A cycle graph with 3 non-muted nodes and 2 muted nodes is always controllable

3.5.2 A 4 node asymmetrical graph

The example shown in the Figure 4 is chosen to illustrate the conservative nature of the result presented in theorem 3. Note that the matrix $\hat{F}$ defined in Theorem 3 would have the following structure:

$$\hat{F} = \begin{bmatrix} f_{21} & 0 \\ f_{31} & f_{34} \end{bmatrix} \tag{43}$$

.

It can be seen that the rank of the matrix $\hat{F}$ is clearly equal to 2 implying that the configuration shown in Figure 4 is controllable. Yet, there exists no map ϕ which will satisfy the requirement given by (39).

3.5.3 C_5 : Cycle graph with 5 nodes

Finally, we present an controllablility analysis of a cycle graph with 5 nodes (see fig 5) with any 2 nodes muted as an example of a topology which is resilient to a certain degree of muteness. Note that any 2 nodes in a cycle graph with 5 nodes have a neighboring node which is not connected to the other node. This allows us to construct a ϕ satisfying the requirements given by 39 quite easily and infer that a cycle graph with 5 nodes is always controllable if only 2 nodes are muted.

3.6 Summary

In this chapter, we explore the connection between the topology and the controllability of a homogenous power grid consisting of prosumers in presence of communication failures. We provide topological tests which can be used to check whether a grid experiencing communication failures is controllable when underlying physical network has certain substructures. These tests can serve as an alternative to rank tests and can simplify the problem of determining controllability significantly. In the next chapter, we explore communication delays, another aspect of information exchange, in the prosumer environment and derive results which provide conditions under which frequency regulation can be achieved in the presence of communication delays.

CHAPTER 4

DISTRIBUTED FREQUENCY REGULATION AND COMMUNICATION DELAYS

The networked nature of the systems considered in this work means that the communication delay associated with exchanging information is an important factor that needs to be considered. For instance, we could envision agents in a multi-agent system relying on iterative protocols to compute an optimal control strategy. Each iteration might require the agents to exchange information with their neighbors. In such cases, the amount of time taken by the agents to compute a control action depends not only on the number of iterations required to compute the optimal solution, but also on the communication delay associated with exchanging information with their neighbours.The work presented in this chapter approaches this problem in the power systems setting. We utilize the linear system model for prosumer based power systems introduced in the Chapter 3 in order to consider the problem of distributed frequency regulation. It provides an answer to the question *"How much information needs to exchanged?"* when certain optimization algorithms are used. For a more detailed account of this work, see [46].

4.1 Overview of Distributed Frequency Regulation

4.1.1 DFR Framework

Frequency regulation is the process of driving frequency to the desired value, 60 or 50 Hz depending on the country [1], using minimal control effort. This is an optimal control problem whose objective is to drive the power deviations to zero using minimal control effort. This work presented here bases its analysis on the frequency regulation problem for prosumer-based energy systems as it is formulated in [33]. For convenience, the problem

[1] Note that most countries operates at 50Hz.

is summarized as follows:

$$\min_{u} J(x(t_c), u) = \min_{u} \sum_{i \in N} p_i x_i(t_c + 1)^2 + r_i u_i^2, \tag{44}$$

subject to coupling constraints

$$\begin{aligned} x_i(t_c + 1) = \quad & a_{ii} x_i(t_c) + b_{ii} u_i(t_c) \\ & + \sum_{j \in N_i} a_{ij} x_j(t_c) + b_{ij} u_j \end{aligned} \tag{45}$$

where $u = \left[u_1, \ldots u_n\right]^T$ and $x(t_c) = \left[x_1(t_c), \ \ldots x_n(t_c)\right]^T$, N is the set of all prosumers ($n = |N|$ is the number of prosumers), x_i and u_i are the power deviation and control variable of prosumer i, $P = diag(p_i)$ and $R = diag(r_i)$ are cost coefficients, and N_i is the set of prosumer i's neighbors. In addition, the system matrices are $A = [a_{ij}]$ and $B = [b_{ij}]$, which have the same sparsity structure as the Laplacian of the grid.

In today's industry, this problem is solved by neglecting coupling between prosumers. Therefore, the problem becomes much simpler and each prosumer solves its sub-problem in a fully decentralized way as:

$$\begin{aligned} \min_{u_i} \quad & \left[p_i x_i(t_c + 1)^2 + r_i u_i^2\right] \\ s.t. \quad & x_i(t_c + 1) = a_{ii} x_i(t_c) + b_{ii} u_i \end{aligned} \tag{46}$$

As discussed in the preceding section, neglecting coupling can cause critical technical problems for the grid. Advanced frequency regulators take into account the effect of tie-line flows [47], but still neglect the effect of neighbors' control strategy.

In [48], a distributed framework for frequency regulation is proposed, under which prosumers have a perception of the decision variables of their neighbors and through a consensus-based ADMM method they achieve agreement on their control strategy. This framework is denoted as "One-Step DFR", because only one-hop communication between prosumers is sufficient to achieve stabilizing optimal control strategies. Under the one-step

DFR framework, the frequency regulation problem is recast as follows [33]:

$$\begin{aligned} \min_{U_1,\cdots U_n} \ & \sum_{i=1}^{n} (p_i \left[A_i^T X_i + B_i^T U_i\right]^2 + r_i U_{ii}^2) \\ \text{s.t.} \ & U_{ij} = U_{jj}, \ \forall i \in N, j \in \mathcal{N}_i, \end{aligned} \tag{47}$$

where A_i, B_i, and U_i are the ith rows of A, B and U matrices, where $U = [U_{ij}]$, $\forall i \in N$ and $\forall j \in \mathcal{N}_i \cup \{i\}$, and U_{ij} is the perception of prosumer i from the control action of its neighbor, prosumer j. In addition, X_i is a column vector, which includes x_j, $j \in \mathcal{N}_i \cup \{i\}$.

In order to solve the DFR problem, the constraints are augmented in the objective function and the ADMM method is used to produce the augmented Lagrangian function as [33], [49], and [50]:

$$\begin{aligned} \mathcal{L}_{\rho,i}(U_i, \bar{U}_i^h, \lambda_i^h) = & \ p_i \left[A_i^T X_i + B_i^T U_i\right]^2 + r_i U_{ii}^2 \\ & + \lambda_i^{h^T} (U_i - \bar{U}_i^h) + \frac{\rho}{2} \|U_i - \bar{U}_i^h\|_2^2, \end{aligned} \tag{48}$$

where, $\rho > 0$ is a given penalty factor, and $\bar{U}_i^h$ is a column vector, which includes the average control strategy of prosumer i and that of its neighbors, defined as:

$$\bar{U}_{ij}^h := \frac{\sum_{l \in \mathcal{N}_j \cup \{j\}} U_{lj}^k}{|\mathcal{N}_j| + 1}, \ \forall j \in \mathcal{N}_i \cup \{i\}. \tag{49}$$

In each iteration, prosumer i computes its optimal control strategy by solving a self-contained problem of the following form:

$$U_i^{h+1} = \underset{U_i}{\text{argmin}} \ \mathcal{L}_{\rho,i}(U_i, \bar{U}_i^h, \lambda_i^h), \tag{50}$$

Next, prosumers share their perceptions with their neighbors and continue this process until errors in power deviations and errors in perceptions become smaller than a desired value.

As shown in [48], DFR is a distributed method for frequency regulation, which can guarantee system-wide stability using minimal control effort. It can also address inter-area oscillations problems because coupling between prosumers is considered in computing minimizing control strategies.

4.1.2 Modelling communication delays

DFR relies on communication between prosumers, which can potentially pose limitations for the convergence of the algorithm, if the communication delays become large. In general, the communication architecture in smart grids supports the functionalities of the DFR algorithm, as the cyber layer has a similar sparsity pattern as the prosumer-based power grid and it connects neighboring prosumers, which are located at seperate geographical regions. The communication networks for prosumer-based energy systems have three classes: ISO-level network for communication between independent system operators (ISO); utility-level network to connect various devices within utilities and interconnect neighboring utilities; and, micro-level network to form a backbone for communication between microgrids, facilities, homes, etc [3].

Furthermore, NERC A1 criterion requires that a prosumer brings power and frequency deviations (area control error, ACE) to zero once every 10 minutes and NERC B2 criterion requires that a prosumer begins to return ACE to zero within 1 minute after the beginning of a disturbance [51]. These reliability criteria enforce a clear time limit for the DFR algorithm. If a prosumer cannot compute stabilizing control strategy within 1 minute, it is a violation, which can lead to system-wide stability problems.

Increasing the communication delay between prosumers increases the risk of violating the NERC reliability criteria as it slows down the convergence of the DFR algorithm (iterations take more time). It can also be noted as the size of the grid increases, the number of iterations required to reach the minimizing control action increases for the DFR algorithm. This has been illustrated in Table I for three practical prosumer-based energy systems with different size and connectivity.

IEEE and the International Electrotechnical Commission (IEC) have defined rigorous standards for communication delay requirements in smart grids in order to ensure reliable operation of the grid and avoid potential stability problems [1]. Table II illustrates a summery of the expected packet delays in different communication categories. It is shown that

Table 1: Convergence of the DFR algorithm for three practical energy systems

Number of prosumers	Number of buses	Number of iterations	Duality gap (pu)
3 (Flores Island)	46	10	10^{-4}
10 (IEEE System)	24	47	10^{-4}
15 (Sao Miguel Island)	1900	151	10^{-4}

Table 2: IEEE and IEC standards for communication timing requirements in smart grids

Information category	Delay requirment
Protection	4 ms
Monitoring and control	16 ms
Medium speed control functions	100 ms
Slow speed auto-control functions	500 ms
Operations and maintenance	1 s

the communication networks are responsible for delivering diverse categories of messages. In addition, some of the messages, such as the monitoring and control information, have critical delay requirements [3].

In reality, the communication networks are not always able to meet the strict communication delay requirements of IEEE and IEC. For instance, experimental results on communication delays between substations, reported in [1] and [2] and summarized in Table III, show that in many scenarios the packet delays exceed the maximum required limit for the most critical messages.

In order to overcome the limitations of the communication delays, in the next section, a new framework is proposed to estimate a lower bound on the number of DFR iterations.

Table 3: Experimental delay measurement for messages requiring immediate actions [1, 2], and [3]

Test scenario for critical messages	Delay rang (ms)
1	0.2 - 0.7
2	3.2 - 17
3	12 - 86
4	32 - 173
5	18 - 97

This algorithm may not be the most efficient but it allows us to explicitly ask questions about the computation budget. This analysis presented in the following section focusses on one-step Model predictive control in a distributed setting. A relatively general problem has been studied in [52]. [52] focusses on identifying optimization algorithms which can solve N-step Model Predictive Control problem within specified time constraints, albeit in a centralized setting.

4.2 Obtaining a Lower Bound for DFR Iterations

4.2.1 Steepest Descent-based DFR

In this section, a gradient descent-based approach is proposed to obtain the computation budget of DFR iterations. The general structure of the gradient descent-based DFR algorithm is formulated as follows.

$$u^{l+1} = -\frac{\gamma}{2}\frac{\partial J(x(t_c), u)}{\partial u} + u^l = Cx(t_c) + Du^l \tag{51}$$

where γ_{t_c} is the step size at time t_c and C and D matrices are defined as:

$$C = -\gamma B^T PA \tag{52}$$

$$D = I - \gamma\left(R + B^T PB\right) \tag{53}$$

It follows from (51), (52), and (53) that the predicted control strategy is related to the square of the Laplacian of the grid, which implies that each prosumer needs to communicate with its neighbors and neighbors' neighbors to estimate its control strategy for the next step.

Equation (54) shows the predicted control strategy for two iterations and L iterations scenarios.

$$\begin{aligned} u(t_c)^2 &= (C + DC)x(t_c) + D^2u(t_c - 1) \\ &\vdots \\ u(t_c)^L &= \sum_{j=0}^{L-1} D^jCx(t_c) + D^Lu(t_c - 1), \end{aligned} \tag{54}$$

In order to determine how many iterations need to be taken to obtaining stabilizing control strategies, one needs to first calculate power deviations at time $t_c + 1$ based on the predicted control strategy at time t_c

$$x(t_c + 1) = \left(A + B \sum_{j=0}^{L-1} D^j C \right) x(t_c) + BD^L u(t_c - 1), \tag{55}$$

combining (54) and (55) leads to:

$$\begin{aligned} \begin{bmatrix} x(t_c + 1) \\ u(t_c) \end{bmatrix} &= A_{steepest} \begin{bmatrix} x(t_c) \\ u(t_c - 1) \end{bmatrix} \\ &= \begin{bmatrix} A + B \sum_{j=0}^{L-1} D^j C & BD^L \\ \sum_{j=0}^{L-1} D^j C & D^L \end{bmatrix} \begin{bmatrix} x(t_c) \\ u(t_c - 1) \end{bmatrix}, \end{aligned} \tag{56}$$

where $A_{steepest}$ is defined as the composite system matrix whose spectral properties determines the stability of the system.

In fact, it is possible to define a formal characterization of the sufficient number of iterations based on the stability of the composite system matrix. The number of iterations (L) needs to be large enough to satisfy the following condition:

$$\left| eig(A_{steepest}(L)) \right| < 1 \tag{57}$$

4.2.2 Stability Condition for Steepest Descent-based DFR

Since (57) guarantees the stability of the composite system, it is important to first understand whether there exists any L for which the stability condition holds. Intuitively, it should be possible to find L if the optimal solution to DFR stabilizes the system. Note that when performing gradient descent starting from an arbitrary initial point, the distance to the optimal solution after L iterations depends on the step size and the distance between the initial estimate and optimal solution.

The following theorem shows that as long as the step-size for the gradient descent process is chosen appropriately, there exists L such that the composite system is always stable regardless of how the initial estimates for the gradient descent process is chosen.

Theorem 4 *Recalling from* (52) *and* (53)*, if* γ *is such that the spectral radius* $\rho(D) < 1$ *and the spectral radius* $\rho(I - B(R + B^T PB)^{-1} B^T P)A) < 1$*, there exists* L *such that* $\rho(A_{steepest}(L)) < 1$.

Note that the spectral radius of D determines the stability of the gradient descent process, while the spectral radius of $I - B(R + B^T PB)^{-1} B^T P)A$ determines the stability of the closed-loop system.

Proof: The key idea behind the proof lies in the following observation. Since $\rho(D) < 1$, the expression $\sum_{i=0}^{L-1} D^i$ corresponds to a convergent geometric sum and therefore converges to $(I - D)^{-1}$ as L approaches ∞. The same assumption also implies that D^L must converge to 0.

Therefore,

$$\begin{bmatrix} A + B\sum_{j=0}^{L-1} D^j C & BD^L \\ \sum_{j=0}^{L-1} D^j C & D^L \end{bmatrix} \rightarrow \begin{bmatrix} A + B(I - D)^{-1} C & 0_{n\times n} \\ (I - D)^{-1} C & 0_{n\times n} \end{bmatrix} \tag{58}$$

as L approaches ∞, where $(I - D)^{-1} = \frac{1}{\gamma}(R + B^T PB)^{-1}$. Substituting for $(I - D)^{-1}$ and C in (58), we obtain

$$A_\infty = \begin{bmatrix} A - B(R + B^T PB)^{-1} B^T PA & 0_{n\times n} \\ -(R + B^T PB)^{-1} B^T PA & 0_{n\times n} \end{bmatrix} \tag{59}$$

The eigenvalues of the block lower triangular matrix A_∞ are the eigenvalues of $A - B(R + B^T PB)^{-1} B^T PA$ and $0_{n\times n}$ (due to the zero matrix on the bottom right corner of A_∞). Thus, all the eigenvalues of A_∞ are contained in the unit circle as the spectral radius of $A - B(R + B^T PB)^{-1} B^T PA$ is less than 1 by assumption [33].

This implies that $\rho(A_\infty) < 1$. Since the spectral radius of a matrix is a continuous function of its entries, we have $\rho(A_{steepest}(L)) \rightarrow \rho(A_\infty) < 1$. This shows that for large enough L, $\rho(A_{steepest}(L)) < 1$.

■

The above theorem guarantees that as long as a "large enough" L budget is chosen, the system represented by (56) would be asymptotically stable. However, the size of the L budget is quite dependent on the optimization procedure and the spectral characteristics of the system matrix (A). This implies that the L budget, found by the gradient descent-based method, can be conservative for many real-world power grids.

In the next section, an alternative and much faster approach is proposed, called Nesterov's accelerated method, which converges to the optimal solution with a quadratic rate as opposed to the steepest descent, which has linear convergence.

4.2.3 Nesterov's Accelerated-based DFR

In this section, the Nesterov's accelerated gradient descent method is applied to obtain an effective L budget for the DFR algorithm. The Nesterov method is a variation of the gradient descent, which uses a variable step-size to accelerate convergence.

The following equations outline the theory of the accelerated gradient method:

$$y^{l+1} = u^l - \gamma \nabla J(u^l) \tag{60}$$

$$u^{l+1} = \eta_l y^l + (1 - \eta_l) y^{l+1} \tag{61}$$

where $u^l, y^l \in \mathbb{R}^n$ (n is the dimension of the system), γ is the step-size and the sequence η_l is defined as:

$$\eta_0 = 0 \tag{62}$$

$$a_l = \frac{1+\sqrt{1+4\eta_{l-1}^2}}{2} \qquad \eta_l = \frac{1-a_{l-1}}{a_l} \tag{63}$$

The process is initialized such that $y^0 = u^0$. Next, it will be shown that y^l converges to the minimum of the DFR cost function (J) for all initial estimates u^0. Recalling from (51), the gradient of the cost function is recast as:

$$\nabla J(x(t_c), u) = B^T P A x(t_c) + (R + B^T P B) u \tag{64}$$

Therefore, the equations (60) and (61) can be re-formulated as follows:

$$y^{l+1} = Du^l + Cx(t_c) \tag{65}$$

$$u^{l+1} = \eta_l y^l + (1 - \eta_l) y^{l+1} \tag{66}$$

Equations (65) and (66) constitute a time-varying linear system, which can be expanded as:

$$\begin{bmatrix} w^{l+1} \\ y^{l+1} \\ u^{l+1} \end{bmatrix} = M_l \begin{bmatrix} w^l \\ y^l \\ u^l \end{bmatrix} + N_l x(t_c) \tag{67}$$

where the state variable w_l is used to keep track of previous values of y^l. In addition, matrices M_I and N_1 are defined as:

$$M_l = \begin{bmatrix} 0_{n\times n} & I_n & 0_{n\times n} \\ 0_{n\times n} & 0_{n\times n} & C \\ \eta_l & 0_{n\times n} & (1-\eta_l)C \end{bmatrix} \tag{68}$$

and

$$N_l = \begin{bmatrix} 0_{n\times n} \\ D \\ (1-\eta_l)D \end{bmatrix}. \tag{69}$$

Using (65) to (69), the Nesterov method can be casted as a linear time-varying system driven by a constant input $x(t_c)$. The response of such a system at time L is given by

$$\begin{bmatrix} w^L \\ y^L \\ u^L \end{bmatrix} = \Phi(0, L) \begin{bmatrix} w^0 \\ y^0 \\ u^0 \end{bmatrix} + F_L x(t_c) \tag{70}$$

where $\Phi(0, L)$ is the state transition matrix and F_L is the discrete time convolution operator, defined as follows:

$$\Phi(0, L) = M_{L-1} M_{L-2} \dots M_0 I_n \text{ when } L > 1 \tag{71}$$

$$F_L = \sum_{k=0}^{L-1} \prod_{i=0}^{L-1-k} M_{L-1-i} N_k \tag{72}$$

Recalling from Section III.A, the composite system matrix for the Nesterov-based DFR algorithm takes the following structure:

$$\begin{aligned}\begin{bmatrix} x(t_c+1) \\ u(t_c) \end{bmatrix} &= A_{Nesterov} \begin{bmatrix} x(t_c) \\ u(t_c-1) \end{bmatrix} \\ &= \begin{bmatrix} A+BPF_L & BT\Phi(0,L)G \\ PF_L & T\Phi(0,L)G \end{bmatrix} \begin{bmatrix} x(t_c) \\ u(t_c-1) \end{bmatrix}\end{aligned} \tag{73}$$

where

$$T = \begin{bmatrix} 0_{n\times n} & I & 0_{n\times n} \end{bmatrix} \tag{74}$$

and

$$G = \begin{bmatrix} I \\ 0_{n\times n} \\ I \end{bmatrix}. \tag{75}$$

The matrix G is used to generate initial conditions for the Nesterov's update equations and T is used to recover the vector of interest (i.e y^L). Since y^L converges to the minimizer of the DFR cost function as L approaches ∞, it can be shown that for large enough L, the composite system will stabilize to the origin. The system given in (73) is asymptotically stable if the spectral radius of the Nesterov's composite matrix is less than 1. Equation (76) illustrates a formal characterization for the stability of the Nesterov method.

$$\rho\left(\begin{bmatrix} A+BPF_L & BT\Phi(0,L)G \\ PF_L & T\Phi(0,L)G \end{bmatrix}\right) < 1 \tag{76}$$

4.3 Simulation Results

In this section, the Steepest descent-based and Nesterov's accelerated-based DFR algorithms are demonstrated on two practical power systems. The first system is the electric power system on Sao Miguel Island, the capital of Azores Archipelago, and the second system is the IEEE 24-bus system. The results show that the L budget depends on the optimization procedure, the spectral characteristics of the system matrix, and the size of the

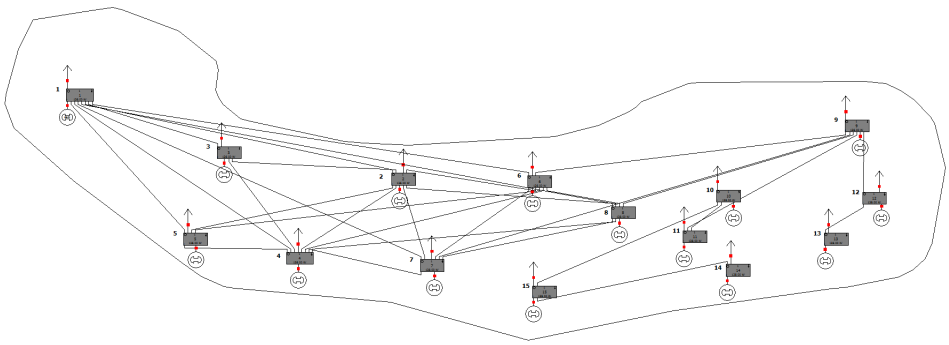


Figure 6: Schematics of the equivalent power grid on Sao Miguel Island

grid.

4.3.1 Computing L budget for Sao-Miguel Island

Sao Miguel is the largest and capital of Azores Archipelago, islands of Portugal. The electric power system on Sao Miguel has more that 2000 lines, around 1900 buses, and 15 generators. The average demand of the island is 70 MW. The detailed description of the Sao Miguel system is presented in [53], [54].

The power system of Sao Miguel is clustered into a prosumer-based structure, where each prosumer represents a control area for frequency regulation. Figure 6 illustrates the schematics of the equivalent power grid on Sao Miguel, in which each node represents a prosumer, which has a generator and a load. The loads are representing the equivalent demand on the prosumers [55].

The DFR cost is chosen such that the minimizer to the cost, which takes on the form of $u^* = -Kx$, becomes an stabilizing control strategy for the system. The state transition matrix and the convolution matrix corresponding to the gradient descent process for different values of L are computed using the following recursive equations:

$$\Phi(0,k) = D\Phi(0,k-1) \tag{77}$$

$$F_k = DF_{k-1} + C \tag{78}$$

As shown in Figure 8, it takes at least 4300 iterations for the gradient descent process to obtain a stabilizing control strategy, which can bring the spectral radius of $A_{steepest}(L)$ to

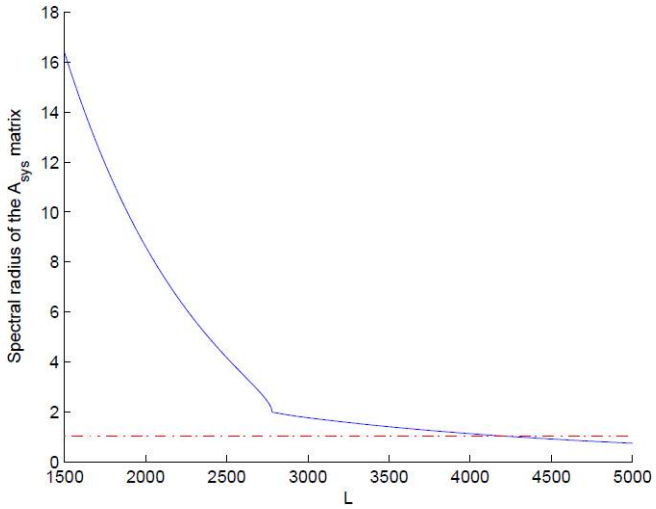


Figure 7: The plot of the spectral radius of the $A_{steepest}$ matrix for the Sao-Miguel island system.

less than 1. On the other hand, for the Nesterov's accelerated method the number of iterations (shown in Figure 9) is drastically less (L is approximately 700) due to the quadratic convergence of the Nesterov process[2].

Assuming that the expected communication delay for each iteration is δ, by increasing the number of iterations the overall time[3] taken to attain a stablizing controller is $\Delta = L\delta$.

Recalling from Section 4.1.1, NERC B2 criterion requires that prosumers start regulating frequency within 1 minute after the disturbance. Therefore, if $\Delta \geq 1$ minute, prosumers will violate the NERC reliability criteria and the DFC algorithm will fail to converge. Computing L allows prosumers to estimate whether they are able to stabilize power and frequency deviations after any arbitrary perturbations within the acceptable time window.

4.3.2 Computing L budget for the IEEE 24-bus system

The next case study is the IEEE 24-bus system, which has 38 lines and 32 generators. The average demand of the system is 2,577 MW. The detailed description of the IEEE 24-bus

[2]Unlike gradient descent, Nesterov's accelerated gradient descent is not a descent method and exhibits oscillations, called Nesterov's ripples, around the optimal solution. This is reflected in the osciallatory behaviour of the spectral radius of the $A_{Nesterov}$ matrix.

[3]Note that the computation delays are considered to be negligible compared to the communication delays. This is a reasonable assumption as the update law just requires taking linear combinations of state measurements, which can be done quite quickly.

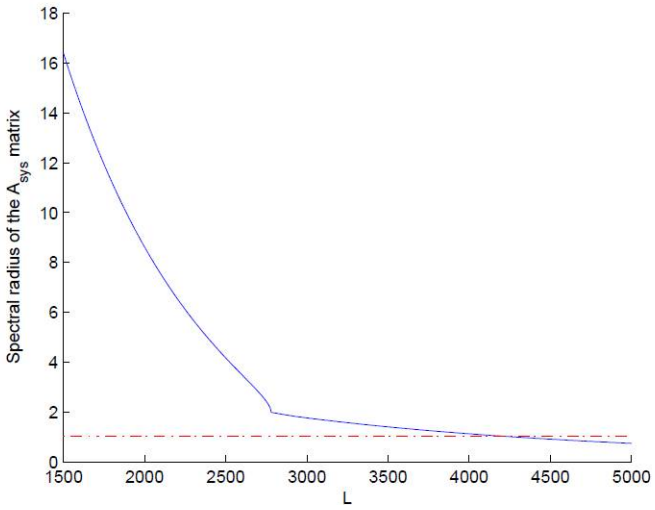


Figure 8: The plot of the spectral radius of the $A_{steepest}$ matrix for the Sao-Miguel island system.

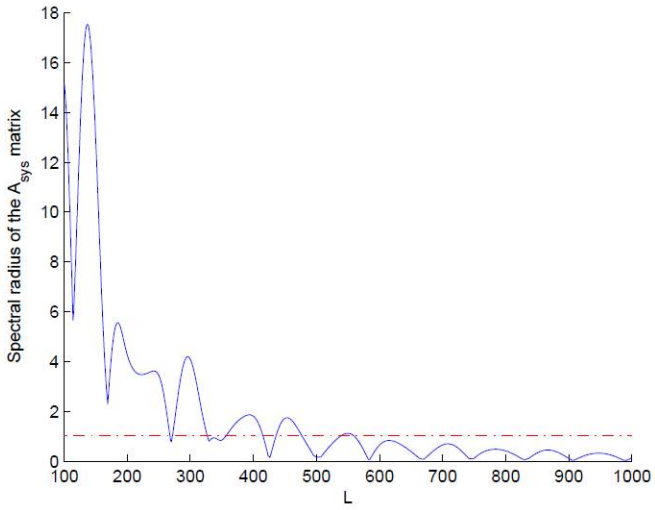


Figure 9: The plot of the spectral radius of the $A_{Nesterov}$ matrix for the Sao-Miguel island.

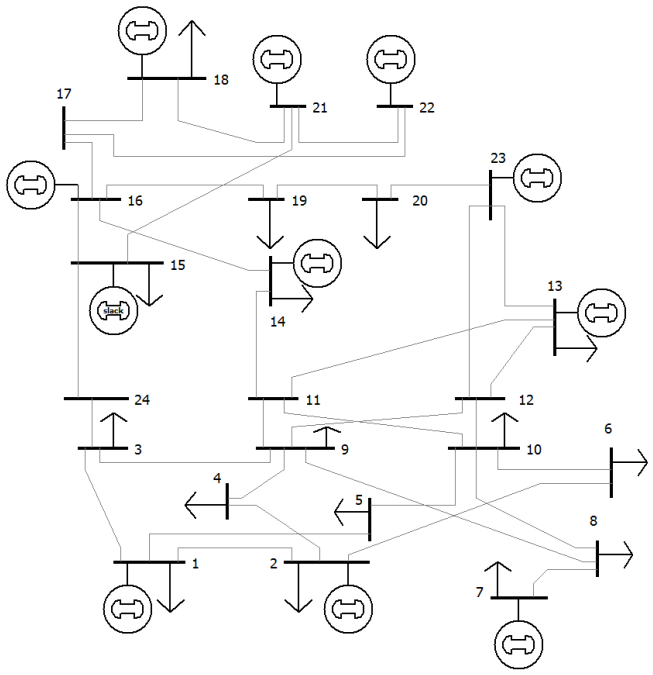


Figure 10: Schematics of the power grid of the IEEE 24-bus system [4]

system is presented in [4].

The power system is clustered into 10 prosumers, where each prosumer represents a utility or area balancing authority. Figure 10 illustrates the schematics of the power grid of the IEEE 24-bus system and Figure 11 demonstrates the cyber-physical network of the prosumer-based IEEE 24-bus system. It is shown in Figure 11 that the cyber-layer has the same sparsity structure as the physical grid.

Figures 12 and 13 illustrate the results of applying the Steepest descent-based and Nesterov's accelerated-based DFR algorithms to the IEEE 24-bus system. It can be observed that the Nesterov's accelerated gradient method outperforms the gradient descent-based approach by a large margin.

Note that, the number of iterations required by the gradient descent DFR for the IEEE 24-bus system is approximately 3300, while it takes more than 4300 iterations to find an stabilizing control strategy for the Sao Miguel system. This is mainly due to the fact that the IEEE system has 10 prosumers and the Sao Miguel Island has 15 prosumers. A similar trend can be observed when comparing Nesterov's accelerated method for the two test

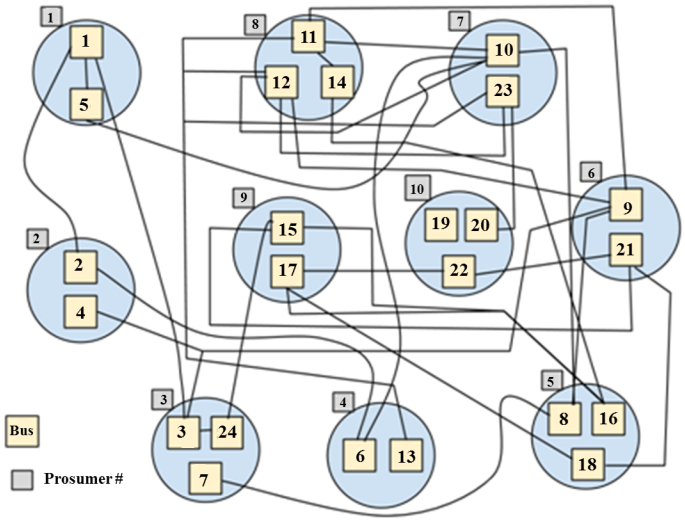


Figure 11: Schematics of the cyber-physical grid of the prosumer-based IEEE 24-bus system

systems.

The findings also show that the Nesterov method provides an acceptable lower bound for the computation budget of the test systems. Assuming that the cyber networks satisfy the communication delay requirements of IEEE ($\delta < 16$ ms), the overall delay for the convergence of DFR for both systems would be with in the acceptable time window ($\Delta <$ 11.2 s for Sao Miguel and $\Delta <$ 5s for the 24-bus system).

4.3.3 Frequency Regulation Performance

The spectral radius of the $A_{steepest}$ or $A_{Nesterov}$ matrices determines the rate at which the power deviations decay down to zero (closer the spectral radius is to 1, slower the convergence) and has a direct impact on the performance. At an execution level, each prosumer improves its initial estimate of the control action by executing L steps of a pre-determined optimization protocol and then applies the improved control action to the system. This process is repeated until the power deviations, and consequently the frequency deviations, are reduced to zero. The simulation results presented thus far demonstrates that the spectral radius of $A_{steepest}$ and $A_{Nesterov}$ depends directly on the number of iterations (L) spent improving the initial estimate.

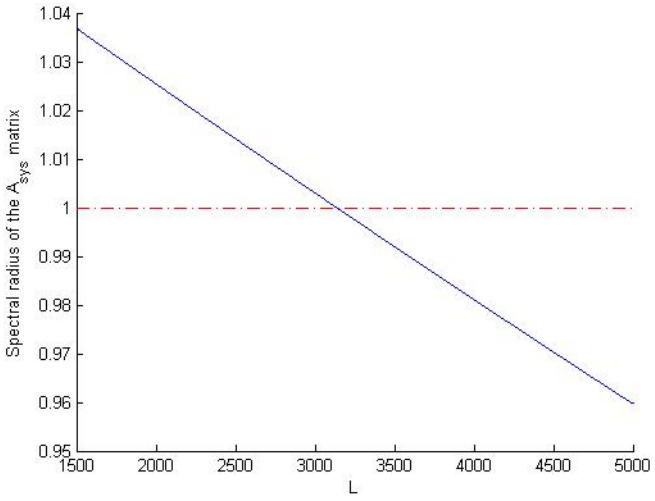


Figure 12: The plot of the spectral radius of the $A_{steepest}$ matrix for the IEEE 24-bus system.

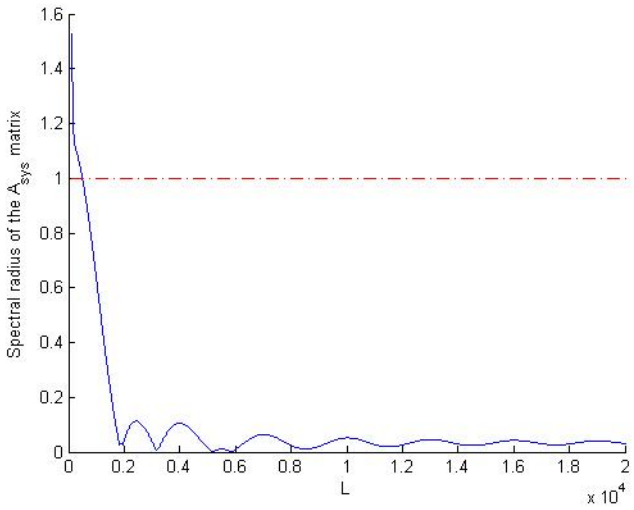


Figure 13: The plot of the spectral radius of the $A_{Nesterov}$ matrix for the IEEE 24-bus system.

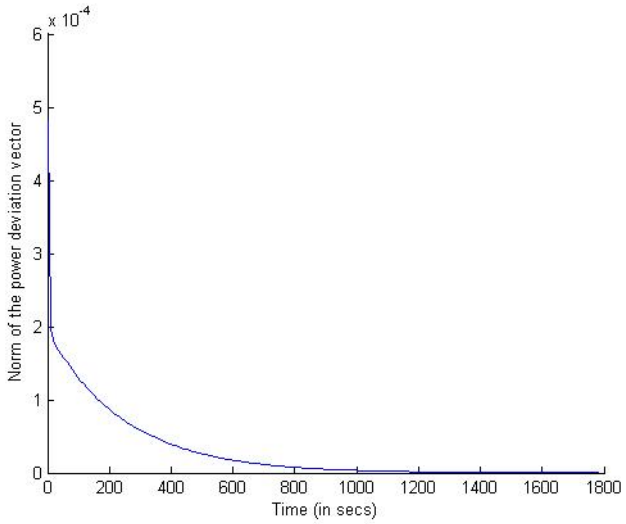


Figure 14: Plot of power deviations (L = 560 , optimization procedure = Nesterov's accelerated descent)

In this section, further simulation results are presented to illustrate and compare the performance of the optimization protocols at different values of L on the IEEE 24-bus system. The dynamics used to simulate the evolution of the power deviations and the input vector is given by (56) (steepest descent) and (73) (Nesterov's accelerated gradient descent). The plots presented in this section track the evolution of power deviations assuming a communication delay δ of 16 ms.

According to Figure 13, the spectral radius of the $A_{Nesterov}$ matrix for the IEEE 24-bus system dips below 1 for $L = 560$. Figure 14 shows the norm of the power deviation when the number of optimization steps used to compute the control action is $L = 560$ when the optimization protocol used is Nesterov's accelerated gradient descent. It takes $L\delta = 8.96$ seconds to compute a control action. The largest eigenvalue of the matrix $A_{Nesterov}$ determines the rate at which the power deviations converge to zero. This can be seen in Figure 14. The spectral radius of $A_{Nesterov}$ is 0.9646 and as such it takes roughly 1200 seconds (20 minutes) for the system to converge to zero.

Figure 15 is a plot of the power deviations when the number of optimization steps is $L = 3600$ when using the steepest descent method. Figure 16 shows a plot of power

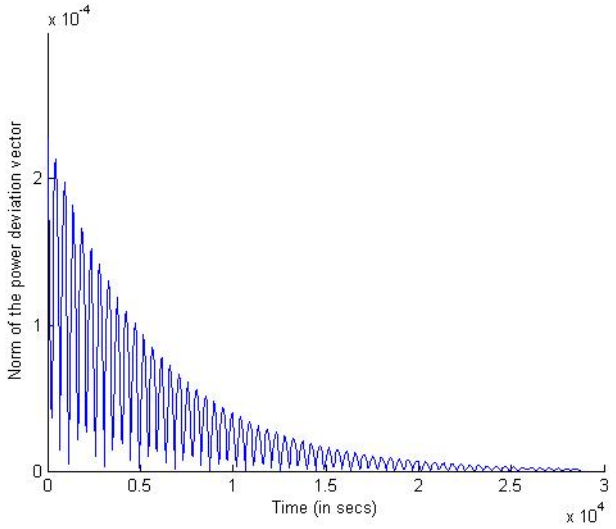


Figure 15: Plot of power deviations (L = 3600 , optimization procedure = steepest descent)

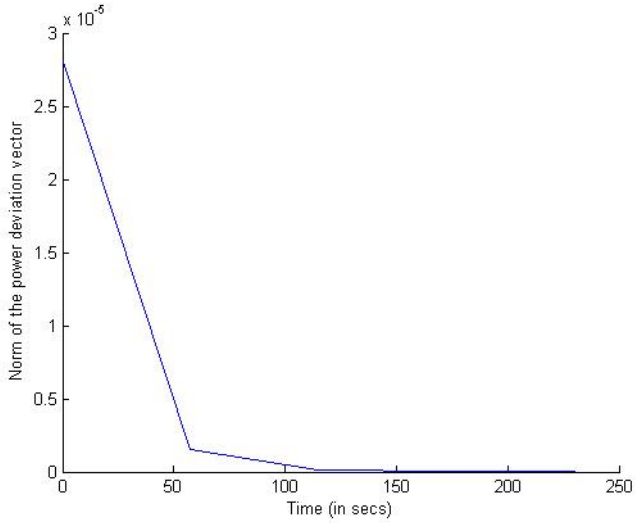


Figure 16: Plot of power deviations (L = 3600 , optimization procedure = Nesterov's accelerated descent)

deviations when the number of optimization steps is $L = 3600$, but for Nesterov's gradient descent method. With $L = 3600$, the delay between the application of control actions is $L\delta = 57.6$s which is still within the one minute limit, imposed by the NERC criteria.

Note that it takes about 500 minutes (8.5 hrs) for the power deviations to stabilize to zero when using steepest descent (Figure 15) as opposed to 4 minutes (approximately) required by Nesterov's accelerated gradient descent (Figure 16). This is due to the fact that the spectral radius of the $A_{steepest}$ matrix is 0.9898, which slows down the convergence. The oscillatory behavior is due to the fact that the largest eigenvalue of the system matrix $A_{steepest}$ happens to be complex when using steepest descent. The fast convergence exhibited in (Fig 16) is due to the extremely small spectral radius of the $A_{Nesterov}$ matrix (0.0783) when $L = 3600$.

It is of importance to note that the 8.5 hrs required for the steepest descent algorithm to stabilize is absolutely not realistic. Nesterov's accelerated gradient algorithm, on the other hand, does stabilize sufficiently fast. However, we do not claim that other algorithms, such as ADMM, would not do better. But, it has been shown that it is possible to connect the computing budget (L) to the system performance in an explicit way.

4.4 Summary

This chapter introduces a method to estimate a lower bound for the computation budget of the two different optimization protocols used to compute the control action for frequency regulation. Under the proposed method, prosumers are able to predict how many iterations they need to take to obtain a stabilizing control strategy in a distributed manner.

In this chapter, the ability of the prosumers to freely exchange measurements and arbitrary information is what allows us to compute a stabilizing control strategy. In the next chapter, we explore the other extreme by studying the stability of linear systems with sparse system matrices while allowing limited information exchange.

CHAPTER 5

ONE-STEP MODEL PREDICTIVE CONTROL AND STABILIZATION

In order to understand the importance of communication, we approach the problem of stabilizing a networked linear system where each agent has access to limited state information obtained via sensing. It is assumed that the agents do not possess any communication capabilities. Instead of addressing the stabilization problem directly, we approximate the original model in such a way that the information constraints are automatically satisfied. We design a model predictive controller which respects the imposed information constraints and stabtilizes the *approximation* of the system. We provide conditions on the original system matrices under which the controller, designed for the approximation, will stabilize the original system. We outline the work done on this problem in the following sections. For a detailed account, see [56].

5.1 System Model and 1-step Model Predictive Control

This section provides a brief discussion of the networked dynamical system which we shall be using throughout this work. The model that we are adapting is quite similar to the power system model being used by [33].

Let $V = \{1, 2 \dots n\}$ represent a group of n autonomous agents connected to each other via some physical infrastructure. Formally, we can represent this by using a graph $\mathcal{G} = (V, E)$ where $|V| = n$ and $(v_i, v_j) \in E$ implies that v_i and v_j are physically coupled. We associate a state x_i with each agent $i \in V$. The state vector x of the system is then given by $x = [x_1, x_2 \dots x_n]^T$. Then, the evolution of the state vector x with respect to a discrete time parameter $k \in \mathbb{N}$ is given by:

$$x(k+1) = A_{\mathcal{G}} x(k) + B_{\mathcal{G}} u(k). \tag{79}$$

where $x(k) \in \mathbb{R}^n$, $A_{\mathcal{G}} \in \mathbb{R}^{n \times n}$ and $B_{\mathcal{G}} \in \mathbb{R}^{n \times n}$. The system matrices $A_{\mathcal{G}}$ and $B_{\mathcal{G}}$ posses a

sparsity structure which reflects the structure of the graph $\mathcal{G}$. More formally, we have

$$(A_{\mathcal{G}})_{ij} = 0 \Leftrightarrow (i, j) \notin E$$

$$(B_{\mathcal{G}})_{ij} = 0 \Leftrightarrow (i, j) \notin E.$$

We assume that the magnitude of the largest eigenvalue λ_{max} of $A_{\mathcal{G}}$ is greater than 1 (i.e the system given by equation (79) is unstable). It is also assumed that the matrix $B_{\mathcal{G}}$ is invertible and as such, the system given by the pair $(A_{\mathcal{G}}, B_{\mathcal{G}})$ is controllable.

In order to describe the information available to a agent, we define the neighborhood set $N_i \subset V$ of an agent represented by $i \in V$ as follows:

$$v \in N_i \Leftrightarrow (v, i) \in E. \tag{80}$$

It is assumed that each agent $i \in V$ is equipped with sensors which allows it to observe the state of all the nodes in its neighborhood set N_i. Due to the sparsity structure of $A_{\mathcal{G}}$, this allows each agent to predict one step into the future under zero input conditions.

We briefly describe the model predictive control methodology used by [33] in order to stabilize a power system whose structure is similar to equation (79).

One approach to stabilizing (79) is Model Predictive Control (MPC) with a time horizon N which attempts to stabilize the state with minimal control effort. The cost function which is typically used takes the form

$$\begin{aligned} J_k(u) = & \sum_{i=k}^{k+(N-1)} x(i)^T Q x(i) + u^T R u \\ & + x(N)^T P x(N) \\ \text{s.t } & x(j+1) = A_{\mathcal{G}} x(j) + B_{\mathcal{G}} u, \;\; j \in \{k, \ldots, k + (N-1)\} \end{aligned}$$

where P, Q and R are positive-definite diagonal matrices and k is the current time instant. The minimizer for the cost function $J_k(u)$ takes the form $u = -Kx(k)$. If there exists some central coordinating unit which has access to the complete state of the system, then the optimal control law for the above system can be computed using numerical methods. In

the absence of such a presence, it is necessary to seek out distributed solutions to the MPC problem so that each agent can use the information that is available to it to compute a stabilizing control action.

One idea proposed in [33] is to limit the amount of information required by each agent by setting the time horizon for the MPC problem to be equal to one. Since the system matrices have the same sparsity structure as that of the graph $\mathcal{G}$, each agent i only needs to observe the current state of the agents in its neighborhood set N_i to predict one step forward in time.The cost function then becomes

$$J_k(u) = x(k+1)^T P x(k+1) + u^T R u \tag{81}$$

$$\text{s.t } x(k+1) = A_{\mathcal{G}} x(k) + B_{\mathcal{G}} u(k). \tag{82}$$

The control law which we obtain by minimizing the above cost takes the form

$$u^* = -(B_{\mathcal{G}}^T P B_{\mathcal{G}} + R)^{-1} B_{\mathcal{G}}^T P A_{\mathcal{G}} x(k) \tag{83}$$

It might not be possible to approximate or compute the matrix $(B_{\mathcal{G}}^T P B_{\mathcal{G}} + R)^{-1}$ in a distributed manner as it might not have the sparsity structure of $\mathcal{G}$. But the cost given by (81) is convex and there exists optimization methods which allows for a distributed computation of u^*. Computing the optimal control $u^* = -Kx(k)$ using distributed optimization methods involves each node making multiple information exchanges with its neighbors. In the absence of a communication network, it might not even be possible to compute the optimal solution to the cost given by equation (81).

One of the key issues with this approach is guaranteeing the stability of the closed loop system obtained by using the above feedback law (83). The closed loop system is obtained by substituting $u = -(B_{\mathcal{G}}^T P B_{\mathcal{G}} + R)^{-1} B_{\mathcal{G}}^T P A_{\mathcal{G}} x(k)$ in (79) and is given by

$$x(k+1) = (I - B_{\mathcal{G}}(B_{\mathcal{G}}^T P B_{\mathcal{G}} + R)^{-1} B_{\mathcal{G}}^T) P A_{\mathcal{G}} x(k).$$

It is necessary to choose the cost matrices P and R such that the state $x(k)$ stabilizes to the origin. One well known condition (see [57]) which guarantees stability requires the P and

R matrices such that they satisify the following Lyapunov inequality

$$(A_{\mathcal{G}} - B_{\mathcal{G}}L)^T P(A_{\mathcal{G}} - B_{\mathcal{G}}L) - P \leq -L^T RL \tag{84}$$

The matrix L is such that $u(k) = -Lx(k)$ is a stabilizing controller. The approach taken in [33] is to choose diagonal P and R matrices such that the diagonal entries of the control cost R are relatively small compared to the diagonal entries of the terminal state cost P which will allow for large control signals.

In the next section, we consider approximations of the model presented in (79) which can be stabilized with relative ease in the absence of a communication network. The rest of the paper focusses on stabilizing this approximation and identifying conditions under which the control signal which stabilizes the approximation can stabilize the system given by (79).

5.2 Approximations to the System Model

Note that equation (83) in Section 2 provides a closed form solution which minimizes the cost function $J_k(u)$ which is of interest. But the inverse term $(B_{\mathcal{G}}PB_{\mathcal{G}} + R)^{-1}$ might not have the sparsity pattern which reflects the structure of the graph $\mathcal{G}$. This makes the direct computation of the the control signal u difficult and we need rely on distributed optimization methods to compute the solution. In the absence of a communication network, it might not possible for the agents to execute such optimization protocols. As such, an alternate approach to the stabilization problem is required.

The main obstruction to the direct computation of the feedback signal given by (83) is the inverse term $(B_{\mathcal{G}}PB_{\mathcal{G}} + R)^{-1}$ which occurs in the solution. The difficulty in inverting the matrix $B_{\mathcal{G}}PB_{\mathcal{G}} + R$ stems from the non-diagonal nature of the matrix $B_{\mathcal{G}}$. Therefore, it might be of interest to construct stabilizing control strategies for a different model with a diagonal input matrix B_d, consisting of the diagonal terms of the matrix $B_{\mathcal{G}}$, and find means to relate it back to the original system.

Formally speaking, we are interested in computing a stabilizing control strategy for the following system:

$$x(k+1) = A_{\mathcal{G}}x(k) + B_d u(k) \tag{85}$$

where $x(k) \in \mathbb{R}^n$, $A_{\mathcal{G}} \in \mathbb{R}^{n\times n}$ has the sparsity structure of $\mathcal{G}$ and the matrix $B_d \in \mathbb{R}^{n\times n}$ is defined as follows:

$$i = j \Leftrightarrow (B_d)_{ij} = (B_{\mathcal{G}})_{ij}$$

$$i \neq j \Leftrightarrow (B_d)_{ij} = 0$$

The pair $(A_{\mathcal{G}}, B_d)$ is controllable as the matrix B_d is a diagonal matrix with non-zero diagonal entries. We would like to stabilize the system given by (85) with minimal control effort at each time instant. The one-step MPC cost function is given by

$$J_k(u) = x(k+1)^T x(k+1) + u^T R u \tag{86}$$

$$\text{s.t } x(k+1) = A_{\mathcal{G}}x(k) + B_d u \tag{87}$$

where the matrix R is positive definite and diagonal in nature. The minimizer for $J_k(u)$ is of the from $u^* = Kx(k)$ where K is given by $-(B_d^T B_d + R)^{-1} B_d^T A_{\mathcal{G}}$. We can immediately observe that the control u^* can be computed in a fast decentralized manner if each agent has access to its neighbor's state. This is because the inverse term $(B_d^T B_d + R)^{-1}$ is a diagonal and $A_{\mathcal{G}}$ has the sparsity structure of the $\mathcal{G}$. As a result, their product K also has the sparsity structure of $\mathcal{G}$ which implies that each agent only needs to measure or observe the power deviations of their neighbors to compute the control action. The state of each agent evolves as follows:

$$x_i(k+1) = \frac{r_i}{(r_i + b_i^2)} \sum_{j\in N_i} a_{ij} x_j(k) \tag{88}$$

where $x_i(k+1)$ denotes the state of agent i and N_i denotes the neighborhoodset of agent i. The terms b_i and r_i corresponds to the ii'th term of the matrices B_d and R.

Note that eventhough the control signal $u(k) = -(B_d^T B_d + R)^{-1} B_d^T A_{\mathcal{G}} x(k)$ can be computed in a distributed manner, it need not be a stabilizing control signal. The stability

depends on the how strict the input penalty is and this can be addressed by allowing the matrix R to be time-varying and dependant on the current state of the system. The question then becomes : Is it possible to choose for each agent i to choose $r_i(k)$ such that the state $x_i(k+1)$ stabilizes to the origin?

5.3 Stability Via Time-varying Cost matrices

In this section, we provide a method to choose the input penalty matrix R in a time-varying and decentralized fashion such that the system given by (88) stabilizes to the origin. Since the matrix $R(k)$ now varies with time, we rewrite the dynamics for agent i as

$$x_i(k+1) = g_i(x_i(k), r_i(k), N_i) = \frac{r_i(k)}{(r_i(k) + b_i^2)} \sum_{j \in N_i} a_{ij} x_j(k). \tag{89}$$

where $x_i(k)$ denotes the current state of agent i, N_i is the closed neighborhood of agent i, and $r_i(k)$ stands for the current input penalty. The ensemble dynamics is then given by

$$x(k+1) = (I - B_d(B_d^T B_d + R(k))^{-1} B_d^T) A_{\mathcal{G}} x(k) \tag{90}$$

The approach we take to guaranteeing stability is to choose $r_i(k)$ such that $V(x) = x^T x$ is a Lyapunov function for (90). For $V(x) = x^T x$ to be a Lyapunov function, we require $V(0) = 0$ and

$$V(x(k+1) - V(x(k)) < -c\|x\|^2 \tag{91}$$

$$\implies \sum_{i=1}^{n} (g_i(x_i(k), r_i(k), N_i)^2 - x_i(k)^2) < -c\|x\|^2 \tag{92}$$

for some $c \in (0, 1)$. If each agent i can ensure that $(g_i(x_i(k), r_i(k), N_i)^2 - x_i(k)^2) < -cx_i^2$ by picking $r_i(k)$, then stability follows as $V(x) = x^T x$ is a Lyapunov function for (90). We present a theorem that shows it is possible to pick $r_i(k)$ such that $(g_i(x_i(k), r_i(k), N_i)^2 - x_i(k)^2) < 0$ if certain conditions are met.

Theorem 5 *For any given time instant k and any agent i, if* $(\sum_{j \in N_i} a_{ij} x_j(k))^2 - x_i(k)^2 > -cx_i^2$, *then there exists* $\mu > 0$ *such that* $g_i(x(k), \mu, N_i)^2 - x_i(k)^2 < -cx_i^2$ *where* $c \in (0, 1)$.

Proof: Let x_i denote the state of the ith agent and k denote a time instant. Let $(\sum_{j\in N_i} a_{ij}x_j(k))^2 - x_i(k)^2 > -cx_i^2$. We have

$$g_i(x_i(k), 0.N_i) - x_i(k)^2 = -x_i(k)^2 < 0. \tag{93}$$

We also obtain

$$\lim_{r\to\infty} g_i(x_i(k), r, N_i)^2 - x_i(k)^2 = \sum_{j\in N_i} a_{ij}x_j(k))^2 - x_i(k)^2. \tag{94}$$

The above equality follows from the fact that

$$\lim_{r\to\infty} \frac{r}{r+b_i^2} = \lim_{r\to\infty}(1 - \frac{b_i^2}{r+b_i^2}) = 1. \tag{95}$$

Taking the derivative of g_i^2 with respect to r, we can infer that

$$\frac{dg_i(x_i(k), r, N_i)^2}{dr} = \frac{2rb_i^2}{(r+b_i^2)^3} \sum_{j\in N_i} (a_{ij}x_j(k))^2 > 0 \tag{96}$$

when $r > 0$.

The above equations can be used to establish the following facts

1. $g_i(x_i(k), r, N_i)^2 - x_i(k)^2$ is a monotonically increasing function of r by (96)

2. $g_i(x_i(k), 0, N_i)^2 - x_i(k)^2 < 0$ by (93)

3. $g_i(x_i(k), p, N_i)^2 - x_i(k)^2 > -cx_i^2$ for sufficiently large p by (94).

Then, by the intermediate value theorem, there exists $r^* \in [0, p]$ such that $g_i(x_i(k), r^*, N_i)^2 - x_i(k)^2 = -cx_i^2$ and $r^* > 0$ (i.e $g_i(x_i(k), r, N_i)^2 - x_i(k)^2$ has a positive square root). We can choose $\mu = r^* - \frac{r^*}{D} < r^*$ where D is a natural number. Since, $g_i(x_i(k), r, N_i)^2 - x_i(k)^2$ is a monotonically increasing function of r, we can establish that

$$g_i(x_i(k), \mu, N_i)^2 - x_i(k)^2 < -cx_i^2 \tag{97}$$

which was to be shown. ■

When $(\sum_{j\in N_i} a_{ij}x_j(k))^2 - x_i(k)^2 > 0$, the value μ can be computed explicitly as follows

$$\mu = max\{p_1(k), p_2(k)\} - \frac{max\{p_1(k), p_2(k)\}}{D} \tag{98}$$

where D is a natural number and $p_1(k)$ and $p_2(k)$ are the roots of the equation $g_i(x_i(k), \mu, N_i)^2 - x_i(k)^2 = -cx_i^2$. The roots can be extracted by using the quadratic formula and are given by

$$p_1(k) = \frac{-b_i^2(1-c)x_i(k)}{(1-c)x_i(k) + \sum_{j\in N_i} a_i j x_j(k)}, \tag{99}$$

$$p_2(k) = \frac{b_i^2(1-c)x_i(k)}{-(1-c)x_i(k) + \sum_{j\in N_i} a_i j x_j(k)}. \tag{100}$$

We can see that both the roots are functions of the system parameters a_{ij} and the neighbors states $x_j(k)$. This allows each area to compute $r_i(k)$ in a decentralized manner. It is also easy to verify the condition $\sum_{j\in N_i} a_{ij}x_j(k)^2 - x_i(k)^2 < 0$ as it only requires us to observe the neighbor's state which we are assuming that we have access to. On the other hand, when $\sum_{j\in N_i} a_{ij}x_j(k)^2 - x_i(k)^2 \leq -cx_i^2$,

$$\lim_{r\to\infty} g_i(x_i(k), r, N_i)^2 - x_i(k)^2 < -cx_i^2. \tag{101}$$

By (93), (96) and (101), we can infer that $g_i(x_i(k), r, N_i)^2 - x_i(k)^2 < 0$ for all values of r and does not have any positive roots. Now we set $r_i(k) = max\{\mu, C\}$

Now, we choose $r_i(k)$ as follows:

$$r_i(k) = max\{\mu, C\} \tag{102}$$

where C is a large positive number. The reason C is required is to handle the situation when the equation $g_i(x_i(k), r, N_i)^2 - x_i(k)^2 = 0$ does not have any positive roots. In such situations, any value of positive value of $r_i(k)$ will cause descent. So, $r_i(k)$ can be chosen to be as large as possible and since $r_i(k)$ cannot be set to infinity, we set it to a large positive number C. We now establish the stability of the system (90) by a Lyapunov argument.

Theorem 6 *The closed loop system described by (90) is asymptotically stable.*

Proof: Let $V(x) = x^T x$. We will demonstrate that V is Lyapunov function for (90). We have $V(0) = 0$ which implies that if V is a Lyapunov function, the system described by (90) would stabilize to the origin. We also have

$$\sum_{i=1}^{n} g_i(x_i(k), r_i(k), N_i)^2 - x_i(k)^2 < -c \sum_{i=1}^{n} x_i^2 \tag{103}$$

$$\implies V(x(k+1) - V(x(k)) < -c \sum_{i=1}^{n} x_i^2 < 0 \tag{104}$$

Also, note that the above equation implies that

$$\|x_{k+1}\| \leq \sqrt{(1-c)}\|x_k\| \tag{105}$$

if we choose $r_i(k)$ is chosen as prescribed by equations (102). Therefore V is a Lyapunov function for (90).

We can then conclude that (90) is asymptotically stable. ■

This allows us to stabilize the approximation (90) in a decentralized manner with minimal computation. In the next section, we provide conditions under which the input signal given by (102) stabilizes the actual control system given by (79).

5.4 Impact of Diagonal Dominance On Stability

The control signal given by (102) stabilizes the approximation (90). In this section, we establish some conditions under which such control signals will stabilize the actual system. We want to understand how the control system given by

$$x_{k+1} = A_{\mathcal{G}} x_k + B_{\mathcal{G}} u_k \tag{106}$$

behaves when driven by the input signal $u_k = -(R(k) + B_d^T B_d)^{-1} B_d^T A_g x_k$.

Note that we can write $B_{\mathcal{G}} = B_d + B_{od}$ where B_d consists of the diagonal entries of $B_{\mathcal{G}}$ and B_{od} contains the off-diagonal terms.

If the entries of B_{od} are small in magnitude, then it might be possible for an agent in the network to ignore the inputs of its neighbors. This in turn allows the controller

$u_k = -(R(k) + B_d^T B_d)^{-1} B_d^T A_g x_k$ to stabilize the system. The following theorem formalizes this notion.

Theorem 7 *Let* $x_{k+1} = A_{\mathcal{G}} x_k + B_{\mathcal{G}} u_k$, *where*

$$u_k = -(R(k) + B_d^T B_d)^{-1} B_d^T A_g x_k. \tag{107}$$

Let $B_{od} = B_{\mathcal{G}} - B_d$. *If*

$$\frac{||B_{od}||}{||B_d||}||A|| + \sqrt{1-c} < 1, \tag{108}$$

then $lim_{k\to\infty} x_k = 0$.

Note : The matrix norm used here is the induced 2-norm given by $||D|| = max_{||x||=1}||Dx||^2$.

Proof: Assume $\frac{||B_{od}||}{||B_d||}||A|| + \sqrt{1-c} < 1$. Applying the triangle inequality to $x_{k+1} = A_{\mathcal{G}} x_k + B_d u_k + B_{od} u_k$, we get

$$||x_{k+1}|| \leq ||A_{\mathcal{G}} x_k + B_d u_k|| + ||B_{od} u_k||. \tag{109}$$

Substituting $u_k = -(R(k) + B_d^T B_d)^{-1} B_d^T A_g x_k$ and using the fact that matrix norms are submultiplicative, we can see that

$$||x_{k+1}|| \leq ||A_{\mathcal{G}} x_k + B_d u_k|| + \frac{||B_{od}||||B_d||}{||R(k)+B_d^2||}||A_{\mathcal{G}}||||x_k||. \tag{110}$$

Since both B_d^2 and $R(k)$ are positive definite diagonal matrices, $||R(k) + B_d^2|| > ||B_d^2|| = ||B_d||^2$. This in turn implies that

$$||x_{k+1}|| \leq ||A_{\mathcal{G}} x_k + B_d u_k|| + \frac{||B_{od}||}{||B_d||}||A_{\mathcal{G}}||||x_k||. \tag{111}$$

We also know that by construction of R_k, $||A_{\mathcal{G}} x_k + B_d u_k|| \leq \sqrt{1-c}||x_k||$. Since $\sqrt{1-c} + \frac{||B_{od}||}{||B_d||}||A_{\mathcal{G}}|| < 1$,

$$||x_{k+1}|| \leq (\sqrt{1-c} + \frac{||B_{od}||}{||B_d||}||A_{\mathcal{G}}||)||x_k|| \to 0 \tag{112}$$

as $k \to \infty$. ■

The ratio $\frac{||B_{od}||}{||B_d||}$ captures the interaction between the diagonal elements and the off diagonal elements of the B matrix. Smaller $\frac{||B_{od}||}{||B_d||}$ implies that the diagonal terms dominate the off diagonal ones which allows the system to stabilize with much more ease. Also note that $||A_{\mathcal{G}}||$ measures the instability of the system in a sense as the absolute value the matrix norm bounds the absolute value of its eigen values. So, larger $||A_{\mathcal{G}}||$ implies that it is harder to stabilize the system. All this information is encoded in the relation $(\sqrt{1-c} + \frac{||B_{od}||}{||B_d||}||A_{\mathcal{G}}||) < 1$.

Intuitively, we would expect the ratio $\frac{||B_{od}||}{||B_d||} < 1$ when the diagonal elements dominate the off diagonal term in some fashion. This notion is made more precise by the following definitions and lemma:

Definition 5.4.1 *A matrix M is said to be column diagonally dominant if $|M_{ii}| \geq \sum_{i \neq j} |M_{ji}|$ for all i.*

Definition 5.4.2 *A matrix M is row diagonally dominant if $|M_{ii}| \geq \sum_{i \neq j} |M_{ij}|$ for all i.*

The notion of diagonal dominance as defined above is usually a characterstic found in distributed power systems. The following lemma shows that if the B matrix is diagonally dominant, then the ratio $\frac{||B_{od}||}{||B_d||}$ is always lesser than 1.

Theorem 8 *Let B be a row and column diagonally dominant matrix. Then $\frac{||B_{od}||}{||B_d||} < 1$ where $B_{od} = B - B_d$ and $B_d = diag(B)$.*

Proof: The proof relies on the matrix norm inequality $||B_{od}|| \leq \sqrt{||B_{od}||_\infty ||B_{od}||_1}$. The $||B_d||_\infty$ is the largest absolute row sum and $||B_d||_1$ is the largest absolute column sum. They are both dominated by $||B_d||$ which is the largest absolute value of its diagonal elements. Therefore

$$||B_{od}|| \leq \sqrt{||B_d||^2} = ||B_d|| \implies \frac{||B_{od}||}{||B_d||} \leq 1 \tag{113}$$

■

Note that this is not enough to guarantee stability. It is necessary for the ratio $\frac{||B_{od}||}{||B_d||}$ to be small enough to counteract the natural instability of the $A_{\mathcal{G}}$ matrix. The larger the unstable eigenvalues of $A_{\mathcal{G}}$ are, smaller the ratio $\frac{||B_{od}||}{||B_d||}$ needs to be, in order to guarantee stability.

5.5 Simulations

In this section, we present simulations of the one-step MPC protocol described above on a small test system with diagonally dominant system matrices.

While we have provided a proof of stability for MPC protocol when the system model is given by

$$x(k+1) = A_{\mathcal{G}}x(k) + B_d u(k) \tag{114}$$

it is of interest to investigate the stability when we are considering the actual model given by

$$x(k+1) = A_{\mathcal{G}}x(k) + B_{\mathcal{G}} u(k) \tag{115}$$

The $A_{\mathcal{G}}$ and $B_{\mathcal{G}}$ system matrices which define the test system behaviour is given by

$$A_{\mathcal{G}} = \begin{bmatrix} -0.3937 & 0.2011 & 8.7127 \\ 0.0391 & -0.1823 & 0 \\ 0.3630 & 0 & -8.70521 \end{bmatrix} \tag{116}$$

$$B_{\mathcal{G}} = B_d + \alpha B_{od} \tag{117}$$

where $\alpha \in (0, 1]$, B_d and B_{od} are given as follows:

$$B_d = \begin{bmatrix} -0.3937 & 0 & 0 \\ 0 & -0.1823 & 0 \\ 0 & 0 & -8.70521 \end{bmatrix} \tag{118}$$

$$B_{od} = \begin{bmatrix} 0 & -0.04021 & 0.1269 \\ -0.0326 & 0 & 0 \\ -0.3025 & 0 & 0 \end{bmatrix} \tag{119}$$

Note that the sparsity pattern of the matrices $A_{\mathcal{G}}$ and $B_{\mathcal{G}}$ matrices shows that the underlying graph $\mathcal{G}$ is a line graph with 3 nodes. Furthermore, the magnitude of the largest eigenvalue of the matrix $A_{\mathcal{G}}$ is 9.0697 implying that the system is unstable. The parameter α controls the magnitude of the off-diagonal entries and determines the diagonal dominance of the matrix $B_{\mathcal{G}}$. We are interested in the stability of the closed loop system given by

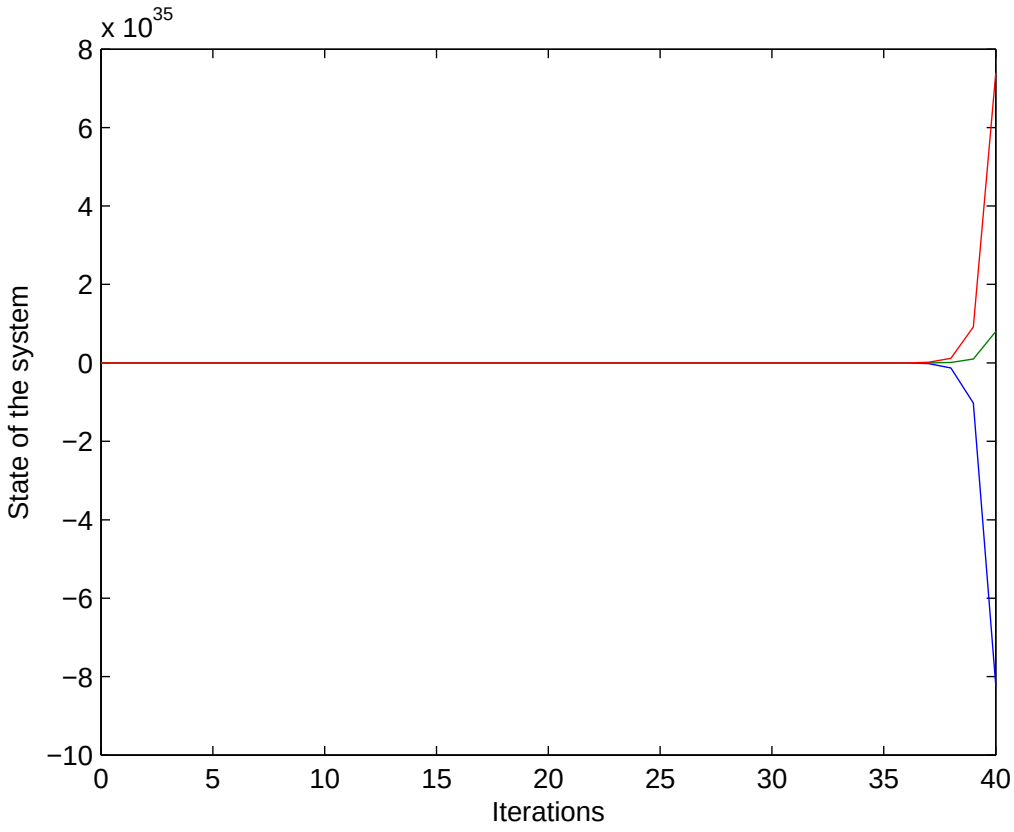


Figure 17: A plot of the state of the system when $\alpha = 1$

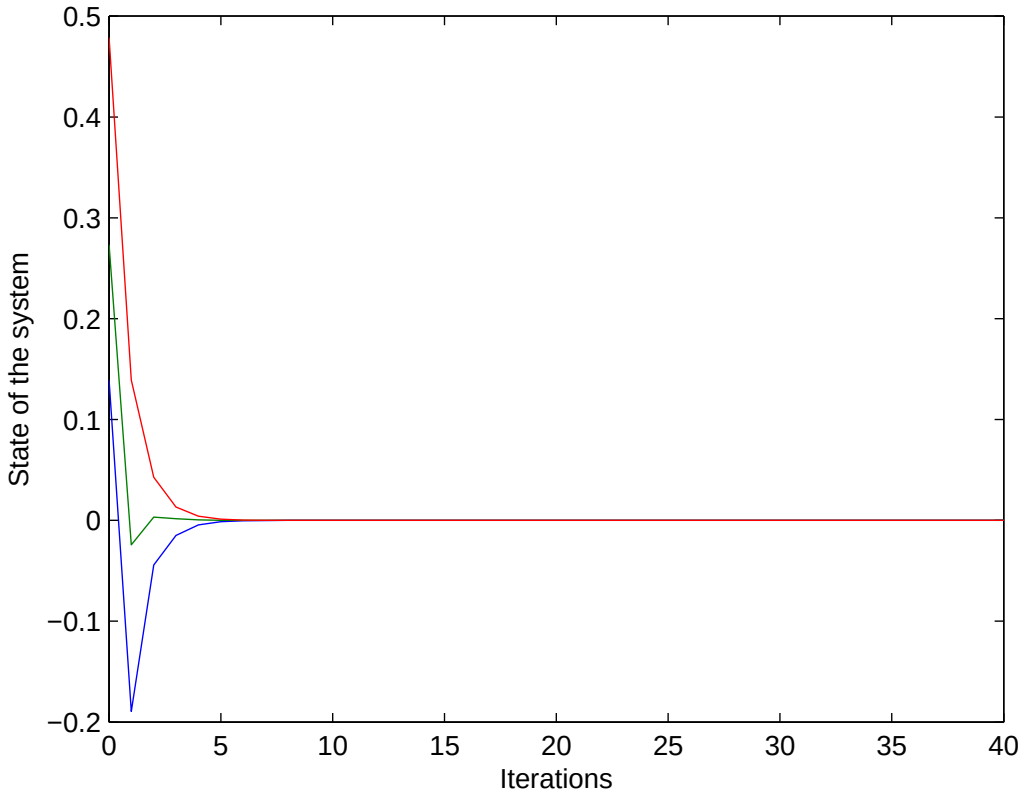


Figure 18: A plot of the state of the system when $\alpha = 0.05$

$$x(k+1) = A_{\mathcal{G}}x(k) - B_{\mathcal{G}}(R(k) + B_d^T B_d)^{-1} B_d^T A_{\mathcal{G}} x(k)$$

where $R(k)$ is computed as described in the previous section by approximating the $B_{\mathcal{G}}$ as a diagonal matrix by considering only its diagonal terms. In this case, the entries of the $R(k)$ are chosen by the individual agents such that

$$||x(k+1)||^2 - ||x(k)||^2 \leq 0.95||x(k)||^2.$$

We will now present simulations of the closed loop dynamics for different values of α and observe that the system remains stable if

$$(\sqrt{0.05} + \tfrac{\alpha||B_{od}||}{||B_d||}||A_{\mathcal{G}}||) < 1. \tag{120}$$

We present simulation results for $\alpha = 1$ and $\alpha = 0.05$ for fixed initial conditions. Note that for $\alpha = 1$, the matrix $B_{\mathcal{G}}$ is not diagonally dominant and $\frac{\alpha||B_{od}||}{||B_d||} > 1$. Also the $||A_{\mathcal{G}}|| = 12.32$. As such the condition given in (120) is not met. So, the stability of the system is guaranteed and Figure 17 shows that the system is unstable. For $\alpha = 0.05$, we have $(\sqrt{0.05} + \frac{\alpha||B_{od}||}{||B_d||}||A_{\mathcal{G}}||) = 0.7810 \leq 1$ which guarantees the stability of the system. This can be seen in Figure 18.

5.6 Summary

In this chapter, the problem of stabilizing a linear system with sparse system matrices was considered. It is shown by considering an approximate model of the actual system, a single step Model Predictive Controller can be computed which relies only local state measurements to stabilize the system. We also provide conditions under which such a controller will stabilize the actual underlyng system.

CHAPTER 6

SUMMARY AND FUTURE WORK

6.1 Summary and Future Work

The research presented in this thesis considers specific problems that occur in networked cyber-physical systems. The focus has been on understanding the problems that arise due to the networked nature of such systems. The work presented can be used as a stepping stone towards the development of general framework which can address problems arising in networked cyber-physical systems. In the subsequent paragraphs, we explicitly identify directions along which the work presented in this dissertation can be expanded.

Development of a general framework : While, we have explored a variety of problems that can occur due to the networked nature of cyber-physical system, we have not provided a general framework which can be used to pharse these problems. This is one of the main directions in which the work presented in this dissertation can be expanded.

Design of Controllable networks : The work presented in Chapter 4 can be further developed and can be used to identify topological substructures which are common among controllable networks. This can be used for designing power networks which would remain controllable even when multiple nodes are experiencing communication failure. This work can also be further expanded by looking at much more general systems where the system matrices have a sparsity structure which reflects the networked nature of the system.

Communication Delays and optimization algorithms : The work presented in Chapter 5 explored the relationship between the communication delays and computation of an optimal control signal when two different optimization algorithms are used. The update law for computing the optimal control signal was linear due to the quadratic nature of the cost and our choice of algorithms. It is not clear how to extend this framework when that is not the case and this merits further investigation.

Another avenue of research involves the exploration of the connection between the stepsize and how it affects the rate at which the spectral radius of aggregate system dynamics presented in Chapter 5. Larger stepsizes leads to instabilty while too small a step size leads to slower convergence. It is of interest to optimize this quantity in order to achieve convergence without sacrificing speed. This has not been explored in this work and is another avenue of research.

In conclusion, we have a explored a diverse subset of problems occuring networked cyber-physical systems. Our initial efforts (Chapter 1 and Chaper 2) focussed on exploring the relationship between information requirements and algorithmic choices. The work presented in Chapter 1 solves the power allocation probelm in a distributed fashion and identifies the connection between information requirements and performance. Chapter 2 addresses the problem of stabilizing a linear system without communication with only sensor measurements. Our later efforts focussed on addressing more diverse problems arising due communication faiulre and delays. In Chapter 3, we explored the relationship between controllability and communication failure and provided topological results. These results can be used to ascertain the controllability of the system by looking at its topology when certain nodes are experiencing communication failure. In Chapter 4, we looked at the relationship between communication delays and computation of optimal control laws using gradient descent and Nesterov's accelerated gradient descent. We provided conditions which can be used to determine the number of iterations that is required to compute a stabilizing control signal in the presence of communication delays.

REFERENCES

[1] V. Skendzic and A. Guzma, "Enhancing power system automation through the use of real-time ethernet," in *Power Systems Conference: Advanced Metering, Protection, Control, Communication, and Distributed Resources, 2006. PS'06*, pp. 480–495, IEEE, 2006.

[2] I. 61850-5, "Communication networks and systems in substations–part 5: Communication requirements for functions and device models," 2003.

[3] W. Wang, Y. Xu, and M. Khanna, "A survey on the communication architectures in smart grid," *Computer Networks*, vol. 55, no. 15, pp. 3604 – 3629, 2011.

[4] C. Grigg, P. Wong, P. Albrecht, R. Allan, M. Bhavaraju, R. Billinton, Q. Chen, C. Fong, S. Haddad, S. Kuruganty, *et al.*, "The ieee reliability test system-1996. a report prepared by the reliability test system task force of the application of probability methods subcommittee," *Power Systems, IEEE Transactions on*, vol. 14, no. 3, pp. 1010–1020, 1999.

[5] P. Tabuada, *Verification and control of hybrid systems: a symbolic approach*. Springer Science & Business Media, 2009.

[6] T. A. Henzinger, *The theory of hybrid automata*. Springer, 2000.

[7] A. Ipakchi and F. Albuyeh, "Grid of the future," *Power and Energy Magazine, IEEE*, vol. 7, no. 2, pp. 52–62, 2009.

[8] H. L. Willis, *Distributed power generation: planning and evaluation*. CRC Press, 2000.

[9] S. Grijalva, M. Costley, and N. Ainsworth, "Prosumer-based control architecture for the future electricity grid," in *Control Applications (CCA), 2011 IEEE International Conference on*, pp. 43–48, IEEE, 2011.

[10] S. Grijalva and M. Tariq, "Prosumer-based smart grid architecture enables a flat, sustainable electricity industry," in *Innovative Smart Grid Technologies (ISGT), 2011 IEEE PES*, pp. 1–6, IEEE, 2011.

[11] M. Egerstedt, "From algorithms to architectures in cyber-physical networks," *Cyber-Physical Systems*, pp. 1–9, 2015.

[12] D. Zelazo, A. Rahmani, and M. Mesbahi, "Agreement via the edge laplacian," in *Decision and Control, 2007 46th IEEE Conference on*, pp. 2309–2314, IEEE, 2007.

[13] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *Automatic Control, IEEE Transactions on*, vol. 48, no. 6, pp. 988–1001, 2003.

[14] A. Chapman, "Advection on graphs," in *Semi-Autonomous Networks*, pp. 3–16, Springer, 2015.

[15] R. Olfati-Saber, A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[16] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, pp. 1520–1533, 2004.

[17] H. G. Tanner, "On the controllability of nearest neighbor interconnections," in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, vol. 3, pp. 2467–2472, IEEE, 2004.

[18] A. Rahmani and M. Mesbahi, "On the controlled agreement problem," in *American Control Conference, 2006*, pp. 6–pp, IEEE, 2006.

[19] M. Ji and M. Egerstedt, "A graph-theoretic characterization of controllability for multi-agent systems," in *American Control Conference, 2007. ACC'07*, pp. 4588–4593, IEEE, 2007.

[20] S. Martini, M. Egerstedt, and A. Bicchi, "Controllability decompositions of networked systems through quotient graphs," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pp. 5244–5249, IEEE, 2008.

[21] A. Rahmani, M. Ji, M. Mesbahi, and M. Egerstedt, "Controllability of multi-agent systems from a graph-theoretic perspective," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 162–186, 2009.

[22] A. Y. Yazicioglu, W. Abbas, and M. Egerstedt, "A tight lower bound on the controllability of networks with multiple leaders," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pp. 1978–1983, IEEE, 2012.

[23] S. Zhang, M. K. Camlibel, and M. Cao, "Controllability of diffusively-coupled multi-agent systems with general and distance regular coupling topologies," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 759–764, IEEE, 2011.

[24] A. Y. Yazicioglu and M. Egerstedt, "Leader selection and network assembly for controllability of leader-follower networks," in *American Control Conference (ACC), 2013*, pp. 3802–3807, IEEE, 2013.

[25] W. Abbas and M. Egerstedt, "Hierarchical assembly of leader-asymmetric, single-leader networks," in *Proceedings of the 2011 American Control Conference*.

[26] J. Wang, Y. Tan, and I. Mareels, "Robustness analysis of leader-follower consensus," *Journal of systems science and complexity*, vol. 22, no. 2, pp. 186–206, 2009.

[27] G. F. Young, L. Scardovi, and N. E. Leonard, "Robustness of noisy consensus dynamics with directed communication," in *American Control Conference (ACC), 2010*, pp. 6312–6317, IEEE, 2010.

[28] G. F. Young, L. Scardovi, and N. E. Leonard, "Rearranging trees for robust consensus," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pp. 1000–1005, IEEE, 2011.

[29] W. Abbas and M. Egerstedt, "Robust graph topologies for networked systems," in *Estimation and Control of Networked Systems*, vol. 3, pp. 85–90, 2012.

[30] J. R. Lawton, R. W. Beard, and B. J. Young, "A decentralized approach to formation maneuvers," *Robotics and Automation, IEEE Transactions on*, vol. 19, no. 6, pp. 933–941, 2003.

[31] S. G. Lee and M. Egerstedt, "Controlled coverage using time-varying density functions," in *Estimation and Control of Networked Systems*, vol. 4, pp. 220–226, 2013.

[32] A. Howard, M. J. Matarić, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Distributed Autonomous Robotic Systems 5*, pp. 299–308, Springer, 2002.

[33] M. H. Nazari, Z. Costello, M. J. Feizollahi, S. Grijalva, and M. Egerstedt, "Distributed frequency control of prosumer-based electric energy systems," *Power Systems, IEEE Transactions on*, vol. 29, no. 6, pp. 2934–2942, 2014.

[34] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE/ACM Transactions on Networking (TON)*, vol. 14, no. SI, pp. 2508–2530, 2006.

[35] M. Rotkowitz and S. Lall, "Decentralized control information structures preserved under feedback," in *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, vol. 1, pp. 569–575, IEEE, 2002.

[36] L. Lessard and S. Lall, "Internal quadratic invariance and decentralized control," in *American Control Conference (ACC), 2010*, pp. 5596–5601, IEEE, 2010.

[37] L. Lessard and S. Lall, "An algebraic framework for quadratic invariance," in *49th IEEE Conference on Decision and Control (CDC)*.

[38] L. Lessard and S. Lall, "Quadratic invariance is necessary and sufficient for convexity," in *American Control Conference (ACC), 2011*, pp. 5360–5362, IEEE, 2011.

[39] M. Rotkowitz and S. Lall, "On computation of optimal controllers subject to quadratically invariant sparsity constraints," in *American Control Conference, 2004. Proceedings of the 2004*, vol. 6, pp. 5659–5664, IEEE, 2004.

[40] J. M. O'Kane and S. M. LaValle, "Localization with limited sensing," *Robotics, IEEE Transactions on*, vol. 23, no. 4, pp. 704–716, 2007.

[41] L. H. Erickson, J. Knuth, J. M. Kane, and S. M. LaValle, "Probabilistic localization with a blind robot," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pp. 1821–1827, IEEE, 2008.

[42] J. Yu, S. M. LaValle, and D. Liberzon, "Rendezvous without coordinates," *Automatic Control, IEEE Transactions on*, vol. 57, no. 2, pp. 421–434, 2012.

[43] T. Ramachandran, Z. Costello, P. Kingston, M. Egerstedt, and S. Grijalva, "Distributed power allocation in prosumer networks," in *Estimation and Control of Networked Systems*, vol. 3, pp. 156–161, 2012.

[44] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton Univ Pr, 2010.

[45] T. Ramachandran, M. Nazari, and M. Egerstedt, "Controllability of prosumer-based networks in the presence of communication failures," in *54th IEEE Conference on Decision and Control, CDC 2015, Osaka, Japan, December 15-18, 2015*, 2015.

[46] T. Ramachandran, M. H. Nazari, S. Grijalva, and M. Egerstedt, "Overcoming communication delays in distributed frequency regulation," *IEEE Transactions on Power Systems*, vol. PP, no. 99, pp. 1–9, 2015.

[47] M. Ilic, X. Liu, B. Eidson, C. Vialas, and M. Athans, "A structure-based modeling and control of electric power systems," *Automatica*, vol. 33, no. 4, pp. 515 – 531, 1997.

[48] M. H. Nazari, Z. Costello, M. Feizollahi, S. Grijalva, and M. Egerstedt, "Distributed frequency control of prosumer-based electric energy systems," *IEEE Transactions on Power Systems*, vol. 29, pp. 2934–2942, Nov 2014.

[49] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–124, 2011.

[50] M. J. Feizollahi, M. Costley, S. Ahmed, and S. Grijalva, "Large-scale decentralized unit commitment," *International Journal of Electrical Power & Energy Systems*, vol. 73, pp. 97–106, 2015.

[51] H. F. Illian, "Frequency control performance measurement and requirements," *Lawrence Berkeley National Laboratory*, 2011.

[52] J. L. Jerez, P. J. Goulart, S. Richter, G. A. Constantinides, E. C. Kerrigan, and M. Morari, "Embedded online optimization for model predictive control at megahertz rates," *Automatic Control, IEEE Transactions on*, vol. 59, no. 12, pp. 3238–3251, 2014.

[53] M. H. Nazari, *Electrical Networks of the Azores Archipelago, in Chapter 3 of Engineering IT-Enabled Sustainable Electricity Services*. Springer, 2013.

[54] M. H. Nazari, *Small-Signal Stability Analysis of Electric Power Systems on the Azores Archipelago, in Chapter 17 of Engineering IT-Enabled Sustainable Electricity Services*. Springer, 2013.

[55] M. H. Nazari, *Making the Most out of Distributed Generation without Endangering Normal Operation: A Model-Based Technical-Policy Approach*. Phd thesis, Carnegie Mellon University, 2012.

[56] T. Ramachandran and M. Egerstedt, "Model predictive control design for stability of linear networked systems with communication," in *IEEE Transactions on Control Systems Technology. (Submitted. Under review.)*.

[57] J. Löfberg, "Linear model predictive control: Stability and robustness," 2001.