ABSTRACT

Title of dissertation: LEARNING IN ENGINEERED MULTI-AGENT SYSTEMS

Anup Menon, Doctor of Philosophy, 2014

Dissertation directed by: Professor John S. Baras
Department of Electrical and Computer Engineering

Consider the problem of maximizing the total power produced by a wind farm. Due to aerodynamic interactions between wind turbines, each turbine maximizing its individual power—as is the case in present-day wind farms—does not lead to optimal farm-level power capture. Further, there are no good models to capture the said aerodynamic interactions, rendering model based optimization techniques ineffective. Thus, model-free distributed algorithms are needed that help turbines adapt their power production on-line so as to maximize farm-level power capture.

Motivated by such problems, the main focus of this dissertation is a distributed model-free optimization problem in the context of multi-agent systems. The set-up comprises of a fixed number of agents, each of which can pick an action and observe the value of its individual utility function. An individual's utility function may depend on the collective action taken by all agents. The exact functional form (or model) of the agent utility functions, however, are unknown; an agent can only measure the numeric value of its utility. The objective of the multi-agent system is to optimize the welfare function (i.e. sum of the individual utility functions).

Such a collaborative task requires communications between agents and we allow for the possibility of such inter-agent communications. We also pay attention to the role played by the pattern of such information exchange on certain aspects of performance.

We develop two algorithms to solve this problem. The first one, *engineered Interactive Trial and Error Learning* (eITEL) algorithm, is based on a line of work in the Learning in Games literature and applies when agent actions are drawn from finite sets. While in a model-free setting, we introduce a novel qualitative graph-theoretic framework to encode known directed interactions of the form "which agents' action affect which others' payoff" (interaction graph). We encode explicit inter-agent communications in a directed graph (communication graph) and, under certain conditions, prove convergence of agent joint action (under eITEL) to the welfare optimizing set. The main condition requires that the union of interaction and communication graphs be strongly connected; thus the algorithm combines an implicit form of communication (via interactions through utility functions) with explicit inter-agent communications to achieve the given collaborative goal. This work has kinship with certain evolutionary computation techniques such as Simulated Annealing; the algorithm steps are carefully designed such that it describes an ergodic Markov chain with a stationary distribution that has support over states where agent joint actions optimize the welfare function. The main analysis tool is perturbed Markov chains and results of broader interest regarding these are derived as well.

The other algorithm, *Collaborative Extremum Seeking* (CES), uses techniques

from extremum seeking control to solve the problem when agent actions are drawn from the set of real numbers. In this case, under the assumption of existence of a local minimizer for the welfare function and a connected undirected communication graph between agents, a result regarding convergence of joint action to a small neighborhood of a local optimizer of the welfare function is proved. Since extremum seeking control uses a simultaneous gradient estimation-descent scheme, gradient information available in the continuous action space formulation is exploited by the CES algorithm to yield improved convergence speeds. The effectiveness of this algorithm for the wind farm power maximization problem is evaluated via simulations.

Lastly, we turn to a different question regarding role of the information exchange pattern on performance of distributed control systems by means of a case study for the vehicle platooning problem. In the vehicle platoon control problem, the objective is to design distributed control laws for individual vehicles in a platoon (or a road-train) that regulate inter-vehicle distances at a specified safe value while the entire platoon follows a leader-vehicle. While most of the literature on the problem deals with some inadequacy in control performance when the information exchange is of the nearest neighbor-type, we consider an arbitrary graph serving as information exchange pattern and derive a relationship between how a certain indicator of control performance is related to the information pattern. Such analysis helps in understanding qualitative features of the 'right' information pattern for this problem.

LEARNING IN ENGINEERED MULTI-AGENT SYSTEMS

by

Anup Menon

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2014

Advisory Committee:
Professor John S. Baras, Chair/Advisor
Professor P. S. Krishnaprasad
Professor Richard J. La
Professor Nuno C. Martins
Professor S. Raghavan

Dedication

To *Muttacha*, my life-long inspiration.

Acknowledgments

First and foremost I would like to express my sincere gratitude towards my advisor, Prof. John Baras. He encouraged me to broaden my interests and supported me while I pursued them. His emphasis on understanding real-world problems, focus on solving problems that matter, and tireless passion for research are some of the things I aspire to imbibe in my professional endeavors. I would like to thank Professors P. S. Krishnaprasad, Richard La, Nuno Martins and S. Raghavan for serving on my committee, and for their valuable feedback on the work. I have benefited greatly from the several courses I have taken from the exceptional faculty in the ECE department and am very grateful for their efforts and patience. I am especially grateful to Prof. Krishnaprasad for his helpful advise and encouragement on several occasions.

I would like to acknowledge the generous support offered by the US Air Force Office of Scientific Research (MURI grant FA9550-09-1-0538), the National Science Foundation (grant CNS-1035655), the National Institute of Standards and Technology (grant 70NANB11H148), and the Defense Advanced Research Projects Agency (DARPA) and the SRC FCRP grant SA00007007.

My office-mates over the years—Christoforos, Evripidis, Yuchen, Vassiliki—have made the room-with-a-window an enjoyable place to work and study. Special thanks are due to David Ward for lending a patient ear to my innumerable "ideas", and to Biswadip Dey and Ion Matei for the many 10-minute-math sessions. I am thankful to Kim Edwards and Alexis Jenkins for helping me navigate through administrative

matters. My time in grad-school has been especially fun thanks to my comrades Karla, Biswa, Arijit, Ranchu, Krishna, Arvind, Lakshmi and Sidharth.

I cannot summon the words that begin to describe my gratitude towards my parents. *Amma*'s selfless devotion to my wellbeing is, and always will be, the primary force behind whatever I achieve in life. *Achaa* has always stood by me, and his belief in my abilities has inspired me to reach higher. *Amumaa, Mausi* and Nidhi's unconditional love and encouragement have helped me stay on course in tough times. In keeping with tradition, I have saved the best for the last. While this dissertation bears only my name as the author, my beloved Sofia deserves equal credit for it. For since I started this journey, she has been my emotional support in moments of distress, my confidence in moments of self-doubt, my strength in moments of disappointment, and has filled this journey with that essential ingredient—happiness.

Table of Contents

List of Figures

List of Abbreviations

$\mathbb{N}$	Set of natural numbers
$\mathbb{R}$	Set of real numbers
$\mathbf{1}$	$[1, \ldots, 1]^T$
$\mathbf{0}$	$[0, \ldots, 0]^T$
$\delta(i, j)$	Kronecker delta function
MAS	Multi-agent System
NE	Nash equilibrium
eITEL	Engineered Interactive Trial and Error Learning
WE	Weak Ergodicity
SE	Strong Ergodicity
CES	Collaborative Extremum Seeking
HPF	High pass filter
LPF	Low pass filter

Notation

All Markov chains discussed in this thesis are discrete-time, finite state space Markov chains. A time-homogeneous Markov chain with Q as its 1-step transition probability matrix means that the i^{th} row and j^{th} column entry $Q_{i,j} = \mathbb{P}(\mathbf{X}_{t+1} = j|\mathbf{X}_t = i)$, where $\mathbf{X}_t$ denotes the state of the chain at time t. If the row vector η_t denotes the probability distribution of the states at time t, then $\eta_{t+1} = \eta_t\ Q$. More generally, if $Q(t)$ denotes the 1-step transition probability matrix of a time-nonhomogeneous Markov chain at time t, then for all $m > n$, $\mathbb{P}(\mathbf{X}_m = j|\mathbf{X}_n = i) = Q_{i,j}^{(n,m)}$, where the matrix $Q^{(n,m)} = Q(n) \cdot Q(n+1) \cdots Q(m-1)$. The time indices of all Markov chains take consecutive values from the set of natural numbers $\mathbb{N}$. For a multi-dimensional vector x, its i^{th} component is denoted by x_i; and that of x_t by $(x_t)_i$.

Given a matrix A,

1. its i^{th} row, j^{th} column element is denoted by $A[i, j]$.
2. the expression $A > (<)0$ denotes that it is a symmetric positive(negative) definite matrix.
3. the expression $A \geq (\leq)0$ denotes that it is a symmetric positive(negative) semi-definite matrix.
4. if it is symmetric, its largest and smallest eigenvalues are denoted by $\lambda_{max}(A)$ and $\lambda_{min}(A)$, respectively.
5. its rank is denoted by $rank(A)$.

Given a vector v, $diag(v)$ denotes the diagonal matrix with $diag(v)[i, i] = v(i)$.

Chapter 1: Introduction

Advances in communication and information technologies have fueled interest in numerous potential applications in other areas of engineering. A prominent idea is information sharing between previously independent decision making subsystems (within a larger system) in order to improve the overall system-wide performance. Such efforts call for appropriate theoretical and algorithmic advancements that can support and guide in achieving such goals, and development of such theory has taken the center stage in research in the Systems and Controls community. While several advances have been made, emerging applications continue to provide new challenges.

This thesis is devoted to the study of decision making in the context of engineered multi-agent systems (MAS)—a group of programmable individual decision making units (or agents), deployed with an intent of achieving a collaborative system-level objective. This is in contrast with other MAS settings considered in literature; for instance, in n-player games the focus is on studying competitive behavior between selfish decision makers.

The majority of discussion in this thesis is centered around synthesis (and analysis) of distributed algorithms for such MAS. These algorithms help solve a certain

collaborative problem that will be described in what follows. The salient feature of this problem is that it is model-free; and consequently the proposed algorithms share the trait of the agents exploring (in some sense) their respective action/input spaces in order to *learn*[1] the desired outcomes.

The usefulness of models and their shortcomings are very well understood in the field of control and optimization. Consequently, the literature is rich with a wide spectrum of techniques that deal with different problem formulations ranging from model-based to model-free. Table 1.1 makes a coarse categorization of some of these. The top two quadrants list widely known and successfully applied centralized techniques. The bottom two quadrants represent relatively recent advancements that extend these in a distributed setting or address formulations that only make sense in a MAS setting. Remarkable progress has been made in model-based distributed techniques in the last two decades—distributed optimization problems (see the seminal work of Tsitsilkis *et al.* [1]; for a more recent account, see Nedic *et al.* [2]), distributed H_2/H_∞ (see Rotkowitz *et al.* [3], Langbort *et al.* [4], and references therein), distributed LQR (see Borrelli *et al.* [5]), etc.

In contrast, progress in distributed model-free approaches has been relatively slow (see bottom right quadrant of Table 1.1. Most of the progress has been made by the Artificial Intelligence (AI) community where several MAS learning scenarios have been considered (see [6] and references therein). Primarily this literature addresses certain multi-agent formulations of Markov decision problems and solution techniques are inspired by appropriate extensions of ideas such as Q-learning, rein-

[1] And hence the title of the thesis.

forcement learning, regret minimization, etc. (see [7]). Often, ideas from Learning in Games have also proven useful. Learning in Games [8,9] refers to a body of work in the Game Theory literature that studies individual behavioral rules for players in an n-player game that lead to discovery of interesting equilibrium behavior. Another related development is the application of extremum seeking control techniques for discovery (or learning) of Nash equilibria in noncooperative n-player games [10, 11]. However, little has been done in the direction of collaborative operation of MAS for achieving a system-level performance objective other than Nash (or other similar) equilibrium. The model-free problem formulation considered in this thesis addresses this shortcoming.

Table 1.1: An Indicative Landscape of Control and Optimization Techniques. (SA stands for Stochastic Approximation, MCMC stands for Markov Chain Monte Carlo)

	Model-based	Model-free
Centralized	Nonlinear programming; Optimal control; LQG; H_∞; Nonlinear Control Techniques	Q-learning; Reinforcement Learning; Simulated Annealing; MCMC; Extremum-seeking; SA
Distributed	Distributed optimization [1, 2]; Distributed H_2/H_∞ [3, 4]; Distributed LQR [5]	Learning in MAS (AI) [6, 7]; Nash seeking [10, 11]; Learning in Games, ?

1.1 Learning in Games and Distributed Control of Engineered MAS

Let us briefly discuss yet another context for the algorithms developed in this thesis. Control of engineered MAS pose many of the same challenges that are encountered in multi-player games. Decisions have to be made locally by agents subject to constraints on the amount of non-local information and models available. An important direction of research in cooperative control leverages this connection to provide a hierarchical decomposition of the MAS control problem. This paradigm consists of: 1. designing individual utility functions for agents such that certain solution concepts (say Nash equilibrium (NE)) correspond to desirable system-wide outcomes; and 2. prescribing learning rules that allows agents to learn such equilibria [12, 13]. The idea being, once agents are programmed to follow the learning rule, they shall discover the NE which, by design, corresponds to a desirable system-wide outcome.

Indeed, the utilities and the learning rules must conform to the agents' information constraints. A popular choice is to design utilities such that the resulting game has a special structure so that the corresponding solution concepts are efficient w.r.t. system-wide objectives. In potential games, for instance, extremal values of the potential function correspond to NE of the game. Some specific examples of such utility design include distributed optimization [14], coverage problems in sensor networks [15] and distributed consensus problems [16]. The most attractive aspects of this paradigm are its scalability and resilience to failure.

The main advantage of designing utilities with special structure is that agents

can be prescribed well known learning algorithms from the vast literature on *learning in repeated games* that helps them learn NE [8, 9]. Many of these algorithms have the desirable feature of being payoff-based, i.e. an agent adjusts its play only on the basis of its past measured payoffs and actions, and does not require any knowledge of the underlying utility functions [17–19]. However, success is guaranteed for most of these learning procedures under an appropriate assumption on the game such as potential, weakly acyclic, congestion game, etc.

Thus, while effective, this paradigm has the following limitations:

- Since available algorithms are provably correct only for certain classes of games, there is a burden to design utilities that conform to such structure for each application.
- If a compromise is made in the utility design phase in order to obtain required game structure, the resulting equilibria may not reflect the desired system-wide outcome.
- If the system requirements prohibit design of utilities with special structure, equilibrating to NE may be inefficient w.r.t. desirable outcome (also, may be unnecessary in non-strategic situations).

While within this paradigm, our approach is complementary to that of utility design. Instead, we focus on algorithm design for welfare (i.e. the sum of individual utilities) optimization for arbitrary utility functions. Finally, in contrast to non-cooperative multi-player games, in engineered MAS, agents can be programmed to follow prescribed behavior without worrying about constraints related to individual

rationality, and explicit communications between the agents can be used to realize collaborative behavior.

1.2 Problem Formulation

Figure 1.1: Photograph of the Horns Rev wind farm with visual illustration of aerodynamic interactions between turbines (notice the turbines amidst the condensation; photo courtesy Christian Steiness).

Before stating the problem, we shall motivate it by discussing an application scenario. Wind farms make profits proportional to the amount of power they produce. Consequently, maximizing the total production of a wind farm is an important problem [20]. However, each turbine maximizing its individual power capture—as is the case in present day wind farms—does not lead to farm-level optimal power capture. This is due to aerodynamic interactions between turbines (see Figure 1.1); energy extraction by an upstream turbine creates a deficit in the wind speed observed by a downstream turbine and consequently a deficit in the latter's power

production. Hence, coordinated control of turbines can help improve total power capture [21–25]. What makes this problem challenging is that accurate models [26] for the said aerodynamic interactions (also called wake effects) are computationally expensive for use in control/optimization purposes and the simplified models are not accurate enough. Hence, model-free learning schemes, where turbines adapt their power set-points on-line in response to instantaneous measured value of their power production so that the sum of the total power produced is optimized, are needed.

This problem may be formulated as follows. Consider a MAS comprising of n agents indexed by i; agent i takes action $u_i \in U_i$, and receives/measures a private utility function $f_i(u)$ that can depend on the joint action $u = (u_1, ..., u_n),\ u \in U$, where $U = \prod_{i=1}^{n} U_i$. The exact functional form of the utility functions $f_i(\cdot)$s, however, is unknown. An agent merely receives or measures the realized values of its respective utility function; for instance, if the collective action at time t is $u(t)$, agent i receives the corresponding realized numerical value $f_i(u(t))$. The MAS has the collaborative objective to minimize the welfare function $W(u) = \sum_{i=1}^{n} f_i(u)$:

$$\min_{u \in U} W(u). \tag{WO}$$

Inter-agent communications can be realized in the context of engineered MAS, and are necessary to achieve such collaborative tasks. Hence, we shall allow for the possibility of such communications. Thus, *distributed* learning algorithms are sought that help each agent *learn* its (respective) action that corresponds to the optimizer of the welfare function. Note that so far we have not specified the permissible set of values U_i. This is deliberate as we shall consider two variants of this problem in

this thesis:

1. $U_i = A_i$ such that $|A_i| < \infty$ and

2. $U_i = \mathbb{R}$.

We shall refer to u_is interchangeably as "actions" or "inputs"; typically the former will be used in the case where U_i is finite and the latter when $U_i = \mathbb{R}$. Other application scenarios that can be modeled in this fashion include efficient coverage of source-fields by a group of robots [27], collaborative surveillance by a group of robots [15], etc.

1.2.1 Role of Information Exchange Network

Since we allow for the possibility of information exchange between the agents, the network over which the agents communicate quite naturally becomes a variable that can be tuned to achieve better performance of the algorithm. The aspect of performance of interest to us, in the context of the problem formulated above, is the speed of convergence of the algorithm. Time and again we shall return to such questions regarding the role of information exchange network on a certain performance aspect.

A broader theme of interest is understanding the role of the information exchange network on the performance of distributed control systems. While distributed control has been an active area of research since the 1970s (evident from the survey article [28]), the most prominent question that has been investigated is: 'Given the model of a decentralized plant and an information exchange structure

between its controllers, sensors and/or actuators, how can a decentralized controller be synthesized that achieves a control objective (like stabilization, LQG, H_∞, etc.)?' We seek answers to a different question; one of design. With the advent of modern communication technologies it is increasingly possible to *design* the network structure or the information exchange pattern between the controllers. This opens up the possibility of designing information patterns that lead to improved control performance.

The immediate roadblocks to such analysis are i) the distributed controller synthesis problem for an arbitrary information exchange pattern is unsolved[2] and ii) optimizing over the set of all information patterns is computationally expensive. Our approach, therefore, is to make context-relevant simplifying assumptions on the controller structure and to try and understand the right qualitative properties of the information pattern that yield good control performance. We discuss this approach in detail in Chapter 4 in the context of the problem of distributed control of vehicle platoons.

1.3 Contributions

1.3.1 Implicit and Explicit Communications for Collaboration: The eITEL Algorithm

The first major contribution of this thesis is the development and analysis of the *engineered Interactive Trial and Error Learning* (eITEL) algorithm that solves

[2]Solutions are known for special cases [3, 4, 29]; but in general the problem is intractable [30].

(WO) for the case when agent actions are drawn from finite sets. This algorithm is a successor to a line of work in the Learning in Games literature beginning with [19] where the *Interactive Trial and Error Learning* procedure was introduced and proved to converge to a (pure strategy) Nash equilibrium (NE) in n-player games. The desirable feature of this work is that the algorithm is model-free; the agents (or players in the Game Theory context) update their actions (drawn from a finite set) solely on the basis of measured values of their individual payoffs and do not know the exact functional form of the underlying utility functions. Indeed, success is guaranteed only when the underlying utility functions satisfy a certain property[3]. Under the same assumptions on the utility functions, this algorithm was further refined in [31] such that learning converged to the most efficient (welfare maximal) NE, and subsequently in [32] to converge to Welfare optimal joint actions.

The two shortcomings of these works that motivated our work are

1. requiring assumptions on utility functions renders the algorithm ineffective for several practical applications, and

2. the notion of convergence in these earlier works is inadequate.

Let us elaborate on the second concern. For a fixed value of a certain parameter $\epsilon > 0$, these algorithms describe an irreducible aperiodic Markov chain with $\mu(\epsilon)$ as its stationary distribution. It is proved that the limiting distribution $\lim_{\epsilon \to 0} \mu(\epsilon) =$

[3]This property is called *interdependence*; it essentially requires that the utility functions be such that each player can observe a change in her payoff as a consequence of a group of players (excluding the player in question) changing their actions. The significance of the result in [19], is that this restriction is far less restrictive than required by other uncoupled NE learning procedures.

$\mu(0)$ has support over states with the desired behavior (like NE, efficient NE, or welfare optimal). Hence, these algorithms converge in the following sense: 'If players implement the learning procedure with a sufficiently small ϵ, then in the limit, the joint action is a NE with high probability'. This is unsatisfactory for two reasons, i) there are no quantitative guidelines for choosing ϵ, and ii) there are results relevant to these algorithms [33] that suggest that smaller the ϵ (i.e. greater the accuracy) slower the convergence of the algorithm.

We introduce a novel graph theoretic framework to encode known qualitative features of the utility functions (while still not knowing the exact functional form of these), and interpret an observed change in an agents utility due to a change of action by another as an "implicit" message sent by the latter to the former (see Section 2.1.1.2). This is referred to as implicit since such communication occurs via the utility function (i.e. via the system itself). Similarly, we encode explicit inter-agent communications in a directed graph. By modifying the algorithm of [32] by appropriate "rewiring" and careful addition of explicit inter-agent communications, we arrive at the eITEL algorithm—a distributed learning procedure that allows agents in an engineered MAS to learn welfare optimal states *without any assumptions on the utility functions*. The key idea is to substitute requirements on utility structure with explicit communications. In fact, an interpretation of the convergence result Theorem 1, loosely speaking, is that 'if the utility structure does not permit enough implicit communications, then successful learning can be guaranteed by carefully adding explicit communications'.

Next, by reducing the parameter ϵ to zero along iterations of the algorithm

(analogous to temperature annealing in simulated annealing), we provide a precise statement regarding convergence of the algorithm in probability under conditions on the rate at which such "annealing" is performed (see Theorem 1). In the process we derive conditions for ergodicity for a nonhomogeneous variant of perturbed Markov chains [34] (a tool used in the analysis of the algorithm). Since there are other algorithms (like [19, 31, 32]) analyzed using the same tool, our proof technique and ergodicity results are expected to be useful in deriving analogous convergence criteria for these.

These results have been published in [35, 36], and can also be found in [37].

1.3.2 Gradient Estimation for Fast Convergence: The CES Algorithm

While performing simulation studies on the eITEL algorithm (discussed above), it was observed that the speed of convergence of eITEL was inadequate. While there are several variables that affect the convergence speed (and perhaps better parameter selection may mitigate the problem to a certain extent), the main reason is an artifact of the problem formulation itself. Since the actions are drawn from a finite set, essentially we have a combinatorial optimization problem at hand. However, our motivating examples consist of systems that can naturally be formulated with agent inputs being drawn from subsets of $\mathbb{R}$. Hence, intuitively, it must be possible to exploit gradient-type information in order to achieve better convergence speeds. Of course, such gradients must be estimated in some sense since ours is a model-free

setting.

Extremum seeking control is one such model-free adaptive control technique that performs a simultaneous gradient estimation-descent. The idea dates back to the 1920s and the technique has found several practical applications in scenarios where the plant is too difficult to model or models are inaccurate (see [38] and references therein). However, the formal proof appeared only in 2000 [39], and ever since the topic has attracted a lot of attention. In the MAS context, recent works [10, 11] have applied extremum seeking control to learn Nash equilibria in non-cooperative n-player games. A contribution of this thesis is the *Collaborative Extremum Seeking* (CES) algorithm—a solution to (WO) with $u_i = \mathbb{R}$ using ideas from extremum seeking control. Also provided is a rigorous justification for local convergence the CES algorithm (see Theorem 5).

Another important contribution is the demonstration of the CES algorithm on the wind farm power maximization problem via numerical simulations. While distributed model-free approaches have been suggested for this problem (see [23, 25]), they either suffer from slow convergence or do not provide formal guarantees of convergence. In contrast we observe promising convergence speeds in our simulations. Yet another contribution is a novel analysis of a dynamic consensus algorithm which, we believe, is an elegant treatment of the topic and is of independent interest (see Section 3.2).

These results are under review [40].

1.3.3 Effects of Information Sharing Network on Distributed Control of Vehicle Platoons

Vehicle platooning refers to grouping a number of vehicles together in a line formation and routing the whole "platoon" in required ways on highways. The idea is that, with intelligent planning and information technology enabled infrastructure, doing so will lead to higher highway throughputs and improved fuel efficiencies [41]. Vehicles traveling at high speeds with close spacings naturally call for automatic control; specifically distributed control laws have to be devised for each vehicle so that the platoon formation is maintained while maintaining prescribed inter-vehicle separation.

The problem has been studied extensively in the control literature [42–47] and the general nature of the results is that, as the number of vehicles increase, there is some kind of inadequacy in control performance when the information exchange between the vehicles is of the nearest-neighbor type (i.e. each vehicle exchanges information only with some of its nearest neighbors in the platoon). We apply the approach introduced in Section 1.2.1 to this problem: we make a simplifying assumption on the individual controllers and try to understand the effect of the information exchange pattern on the control performance in this problem. The main contribution is stated in Theorem 7 which relates a certain metric of control performance to a certain characteristic of the information exchange pattern. This is a generalization of previous results regarding such performance for specific information exchange patterns. Based on the findings, we argue that expander family of

graphs—mathematical objects that have found applications in coding theory and theoretical computer science—provide the right trade-off between acceptable control performance without excessive communication overhead.

These results have been published in [48].

1.4 Outline

The remainder of this thesis is organized as follows. In Chapter 2 we explain the qualitative modeling framework of interaction graphs (briefly touched upon earlier) and describe the eITEL algorithm that solves (WO). The main convergence guarantees are presented followed by detailed analysis. Results of the simulation studies performed are presented.

Chapter 3 addresses (WO) with $U_i = \mathbb{R}$. We begin by explaining how we propose to use Extremum Seeking control in solving (WO). Description of CES, the main convergence results, and the analysis are presented next. The wind farm model which is used to test the model-free CES algorithm is described, followed by the results of simulation study.

Chapter 4 presents the analysis of the distributed control of vehicle platoon problem with a general information exchange network between the vehicles. The main result is proved, expander families of graphs are introduced, and the performance benefits of using expanders in platooning problem are illustrated via numerical simulations.

General notation is explained in the beginning of the the thesis along with

abbreviations; chapter specific notation, if any, is explained at the end of the first section of the respective chapter. Proofs that use standard arguments have been placed in appendices to improve readability.

Chapter 2: Ideas from Learning in Games

The idea of the line of research presented in this chapter is to leverage existing work in learning in repeated games to design distributed algorithms that solve the model-free welfare optimization problem (WO) formulated in Section 1.2. Building on the work of Marden *et al.* [32], the *engineered Interactive Trial and Error Learning* algorithm is developed in this chapter that allows the agents to learn welfare optimal actions in a model-free setting without *any* assumptions on the underlying utility structure. Some salient aspects of the algorithm include:

1. Completely payoff-based implementation with no requirement of model information.

2. Changes in utility measured by an agent as a result of another agent changing its action is interpreted as an "implicit" communication to the former from the latter, and a graph-based framework is developed to capture such qualitative interactions between the agents.

3. Explicit communication between the agents is necessitated for the success of the proposed algorithm when the given utility structure does not afford enough implicit communications.

4. Conditions that guarantee asymptotic convergence to welfare optimal actions w.p. 1 are given.

The chapter is organized as follows. In section 2.1 we develop the analysis framework, present the algorithm and state the main convergence result. Section 2.2 introduces perturbed Markov chains and states relevant results. In Section 2.3, the results of Section 2.2 are used to prove Theorem 1. The chapter concludes with some numerical illustrations and discussions about future work.

2.1 Analysis Framework and Algorithm

In this section we develop a generic framework that several MAS can be cast into. Possible application scenarios and how they might be modeled in the proposed setup are discussed alongside.

2.1.1 A Coarse Modeling Framework

2.1.1.1 Agent Model

We consider n, possibly heterogeneous, agents indexed by $i \in \{1, ..., n\}$. The i^{th} agent can pick actions from a set $\mathcal{A}_i$, $1 < |\mathcal{A}_i| < \infty$; the joint action of the agents is an element of the set $\mathcal{A} = \prod_{i=1}^{n} \mathcal{A}_i$[1]. The action of the i^{th} agent in the joint action $a \in \mathcal{A}$ is denoted by a_i. Further, given the i^{th} individual's present action is $b \in \mathcal{A}_i$,

[1]The agent actions and action sets are denoted by a and $\mathcal{A}$ respectively in this chapter. This is intended to remind the reader of the finite action space setting and retain consistency with the Learning in Games literature.

the choice of its very next action is restricted to be from $\mathcal{A}_i(b) \subseteq \mathcal{A}_i$.

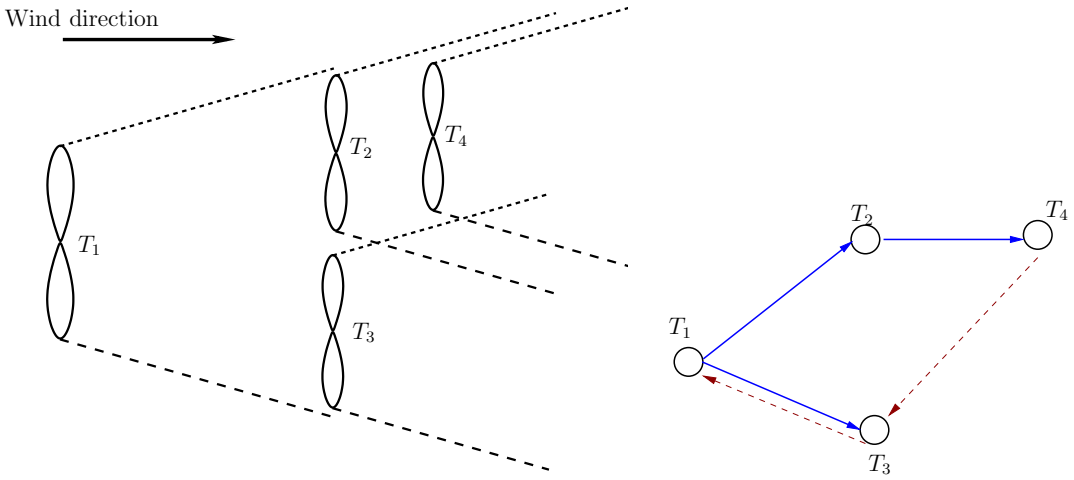
Assumption 1 (Reachability)**.** *For any $b \in \mathcal{A}_i$, $b \in \mathcal{A}_i(b)$; and for any $a_1, a_2 \in \mathcal{A}_i$, there exist a finite sequence $\{a_1 = b_1, b_2, ..., b_m = a_2\}$ where $b_{j+1} \in \mathcal{A}_i(b_j)$ for $j = 1, ..., (m-1)$.*

In particular this allows for the possibility of picking the same action in consecutive steps and for any element of $\mathcal{A}_i$ to be "reachable" from any other in finite number of steps. Specific instances of such agent models in literature include the discretized position and viewing-angle sets for mobile sensors in [15], discretized position of a robot in a finite lattice in [49, 50] and the discretization of the axial induction factor of a wind turbine in [25].

Each individual has its utility that can be an arbitrary time-invariant function of the action taken by the whole but is measured or accessed only by the individual. Agent i's utility is denoted by $f_i : \mathcal{A} \to \mathbb{R}^+$. Examples include artificial potentials used to encode information about desired formation geometry for collaborative control of autonomous robots in [49, 50] and the measured power output of an individual wind turbine in [25]. At any time t, agent i only measures or receives $(f_t^{mes})_i = f_i(a_t)$ since neither the joint action a_t nor the functional form of $f_i(\cdot)$ is known to the agent.

Let us set $W^* = \min_{a \in \mathcal{A}} W(a)$, and in accordance to the objective of the MAS stated in 1.2, we seek convergence of collective agent actions to the set

$$\mathcal{A}^* = \{\arg\min_{a \in \mathcal{A}} W(a)\}.$$



(a) A loop represents a wind turbine and the dotted lines its corresponding wake.

(b) Solid arrows represent edges in $\mathcal{G}_I$ and the dotted arrows edges in $\mathcal{G}_c$.

Figure 2.1: Wind farm modeled in proposed framework

2.1.1.2 Interaction Model

Interaction in a multi-agent setting can comprise of explicit communication between agents or can be implicit with actions of an agent reflecting on the payoff of another. The latter is an artifact of the given problem at hand and must be modeled appropriately. We present a general modeling framework that allows the designer to encode known inter-agent interactions in the system. While the communication network varies from one application to another, for the eITEL algorithm we only require a simple signaling network where a bit-valued variable is exchanged amongst the agents.

1. **Interaction Graph.** Consider a directed graph $\mathcal{G}_I(a)$ for every $a \in \mathcal{A}$ with a vertex assigned to each agent. Its edge set contains the directed edge (j, i) if

and only if $\exists\, b \in \mathcal{A}_j$ such that $f_i(a) \neq f_i(b, a_{-j})$.[2] Thus, for every action profile a, $\mathcal{G}_I(a)$ encodes the directional interaction between agents whose actions can (and must) affect the payoffs of other specific agents. We call this graph the *interaction graph*.

In the case of a wind farm, power production of a turbine downstream of another is affected by the actions of the latter (see Figure 2.1(a)). For the collaborative robotics problem, all robots that contribute to the artificial potential of a given robot constitute the latter's in-neighbors in the interaction graph. Essentially, the interaction graph is a way of encoding certain "coarse" information about the structure of the payoff functions even in the absence of explicit knowledge of their functional forms.

2. **Communication Graph.** The agents are assumed to have a mechanism to transmit a bit-valued message to other agents in its neighborhood (within a certain range, for instance). The mode of communication is broadcast and an agent need not know which other agents are receiving its message. For each $a \in \mathcal{A}$, we model this explicit information exchange by a directed graph $\mathcal{G}_c(a)$ over the set of agents called the *communication graph*. A directed edge (j, i) in $\mathcal{G}_c(a)$ represents that agent j can send a message to agent i when the joint action being played is a. Let $\mathcal{N}_i(a)$ denote the in-neighbors of agent i in the communication graph. Each transmission is assumed to be completed within

[2] We borrow notation from the game theory literature: a_J denotes the actions taken by the agents in subset J from the collective action a and the actions of the rest is denoted by a_{-J}.

the duration of the algorithm iterate.

The dependence of the above graphs on $a \in \mathcal{A}$ is required for accurate modeling. For instance, in the collaborative robotics example where $\mathcal{A}_i$ represents the set of discretized positions of robot i, which robots are within the interaction or communication range of specific others depends on the position of all robots ($\in \mathcal{A}$). On the contrary, in the wind farm example, these graphs can be considered constant for all $a \in \mathcal{A}$ (see Figure 2.1(b)). We stress that this framework is for modeling and analysis at the level of the system designer; the agents neither know the joint action nor the corresponding neighbors in $\mathcal{G}_I(\cdot)$ or $\mathcal{G}_c(\cdot)$. They simply go about measuring their utilities, broadcasting messages and receiving such messages from other agents.

2.1.2 The eITEL Algorithm

The algorithm requires each agent to synchronously update a certain individual state (which in part involves picking an action) on the basis of the last state and measured utility. Endow agent i with a state $x_i = [a_i, m_i]$, where $a_i \in \mathcal{A}_i$ corresponds to the action picked and m_i is a certain $\{0, 1\}$-valued 'mood' variable of agent i. When the mood variable equals 1 we call the agent "content" else "discontent". The collective state of all agents is denoted by $x = (a, m)$. Additionally, each agent maintains a variable $\overline{f}_i$, which records the payoff it received in the previous iterate. At $t = 0$, the agent i initializes $(m_0)_i = 0$, picks an arbitrary $(a_0)_i \in \mathcal{A}_i$ and records the received payoff $\overline{f}_i = (f_0^{mes})_i$. With slight abuse of notation, we let $\mathcal{G}_c(a_t) = \mathcal{G}_c(t)$ and $\mathcal{A}_i((a_t)_i) = \mathcal{A}_i(t)$.

For a certain pre-specified monotone decreasing sequence $\{\epsilon_t\}_{t\in\mathbb{N}}$, with $\epsilon_t \to 0$ as $t \to \infty$, and constants $c > W^*$, $\beta_C, \beta_I > 0$, agent i performs the following for each $t \in \{1, 2, ...\}$.

Step 1: Receive $(\mathbf{m}_{t-1})_j$ from all $j \in \mathcal{N}_i(t-1)$, i.e. the in-neighbors of i in $\mathcal{G}_c(t-1)$.

Compute temporary variable $\tilde{m}_i$ as follows.

1. If $(\mathbf{m}_{t-1})_i = 0$, set $\tilde{m}_i = 0$;
2. else, if $(\mathbf{m}_{t-1})_i = 1$ and $\prod_{j\in\mathcal{N}_i(t-1)} (\mathbf{m}_{t-1})_j = 1$, set $\tilde{m}_i = 1$;
3. else, if $(\mathbf{m}_{t-1})_i = 1$ and $\prod_{j\in\mathcal{N}_i(t-1)} (\mathbf{m}_{t-1})_j = 0$,
   set $\tilde{m}_i = \{0, 1\}$ w.p. $\{1 - \epsilon_t^{\beta_C}, \epsilon_t^{\beta_C}\}$.

Step 2: Pick $(\mathbf{a}_t)_i$ as follows.

1. If $\tilde{m}_i = 1$, pick $(\mathbf{a}_t)_i$ from $\mathcal{A}_i(t-1)$ according to the p.m.f.
$$p(b) = \begin{cases} 1 - \epsilon_t^c & \text{if } b = (\mathbf{a}_{t-1})_i \\ \frac{\epsilon_t^c}{|\mathcal{A}_i(t-1)|-1} & \text{otherwise.} \end{cases} \tag{2.1}$$
2. Else, if $\tilde{m}_i = 0$, pick $(\mathbf{a}_t)_i$ according to the uniform distribution
$$p(b) = \frac{1}{|\mathcal{A}_i(t-1)|} \text{ for all } b \in \mathcal{A}_i(t-1). \tag{2.2}$$

Step 3: Measure or receive payoff $(\mathbf{f}_t^{mes})_i (= f_i(\mathbf{a}_t))$.

Step 4: Update $(\mathbf{m}_t)_i$ as follows.

1. If $\tilde{m}_i = 1$ and $((\mathbf{a}_t)_i, (\mathbf{f}_t^{mes})_i) = ((\mathbf{a}_{t-1})_i, \overline{f}_i)$, then set $(\mathbf{m}_t)_i = 1$;
2. else, if $\tilde{m}_i = 1$ and $((\mathbf{a}_t)_i, (\mathbf{f}_t^{mes})_i) \neq ((\mathbf{a}_{t-1})_i, \overline{f}_i)$,
   set $(\mathbf{m}_t)_i = \{0, 1\}$ w.p. $\{1 - \epsilon_t^{\beta_I}, \epsilon_t^{\beta_I}\}$;

3. and if $\tilde{m}_i = 0$, set

$$(\mathbf{m}_t)_i = \begin{cases} 1 & \text{w.p.} \quad \epsilon_t^{(\mathbf{f}_t^{mes})_i} \\ 0 & \text{w.p.} \quad 1 - \epsilon_t^{(\mathbf{f}_t^{mes})_i}. \end{cases} \tag{2.3}$$

Update $\overline{f}_i \leftarrow (\mathbf{f}_t^{mes})_i$.

Step 5: Broadcast $(\mathbf{m}_t)_i$ to all out-neighbors in $\mathcal{G}_c(t)$. Go to Step 1 at $t+1$.

It is easy to see that the algorithm defines a nonhomogeneous Markov chain on the state space $S = \mathcal{A} \times \{0,1\}^n$.

2.1.2.1 Convergence Result

The following is the main convergence result for the eITEL algorithm.

Theorem 1 (eITEL Convergence)**.** *Let*

1. $\sum_{t=1}^{\infty} \epsilon_t^c = \infty$ *and*
2. *for every* $a \in \mathcal{A}$, $\mathcal{G}_c(a) \cup \mathcal{G}_I(a)$ *be strongly connected.*

Then, if $\mathbf{X}_t = [\mathbf{a}_t, \mathbf{m}_t]$ *denotes the collective state of the agents at time* t*,*

$$\lim_{t\to\infty} \mathbb{P}[\mathbf{a}_t \in \mathcal{A}^*] = 1.$$

The theorem is proved in Section 2.3. The idea of the proof is to first hold ϵ_t fixed at $\epsilon > 0$, resulting in the algorithm describing a homogeneous Markov chain. The stationary distribution of this Markov chain, in the limit as $\epsilon \to 0$, is shown to have support over states corresponding to agents playing efficient actions. Then, conditions on how fast the noise parameter may be reduced to zero along iterations of

the algorithm while maintaining ergodicity are derived. This is analogous to the rate conditions on the temperature variable in Simulated Annealing [51], [52]. The tool for the first step is the framework of *perturbed Markov chains*—introduced by Young in the seminal work [34]—while the second step utilizes Theorem 3 developed in Section 2.2 that gives conditions that ensure ergodicity of a certain nonhomogeneous variant of perturbed Markov chains.

From a practical view-point, the result provides the system-designer with guidelines on how to guarantee convergence of eITEL algorithm. The first assumption translates to a constraint on the sequence $\{\epsilon_t\}_{t\in\mathbb{N}}$ on how fast it may approach zero (analogous to conditions on the annealing schedule in simulated annealing).

The second provides flexibility to design a 'minimal' communication network by utilizing information about the payoff structure (such a choice is made in Figure 2.1(b)). For instance, if the designer has no information about the structure of the payoff function ($\mathcal{G}_I(\cdot) = \emptyset$), a communication network where $\mathcal{G}_c(a)$ is strongly connected for all $a \in \mathcal{A}$ can be installed to guarantee convergence. The other extreme case is for all $a \in \mathcal{A}$, $\mathcal{G}_I(a)$ is strongly connected; then eITEL converges even in the absence of any explicit communication.

2.2 Perturbed Markov Chains

In this section we introduce perturbed Markov chains, the main tool used in the analysis of the eITEL algorithm. Consider a homogeneous Markov chain with possibly several stationary distributions. Perturb the transition matrix of this

chain by adding appropriate functions of a certain noise parameter ϵ to obtain an ergodic chain with a unique stationary distribution. These functions are such that, as $\epsilon \to 0$, the transition matrix of the perturbed chain converges to that of the 'unperturbed chain' with the $x-$row, $y-$column element converging to its limit with an asymptotic rate $r(x, y)$ (i.e. as $\Theta(\epsilon^{r(x,y)})$). What is interesting is that, under appropriate conditions, as $\epsilon \to 0$, the stationary distribution of the perturbed chain converges to a certain stationary distribution of the unperturbed chain and the support of the latter can be characterized in terms of the rates $r(\cdot, \cdot)$. Thus, in effect, one can "choose" amongst the possibly several stationary distributions of the unperturbed chain. This section describes this theory in detail and also how to reduce ϵ over time while evolving according to the perturbed chain (rendering the chain nonhomogeneous) while retaining ergodicity.

While results similar to Theorem 3 have appeared in the Game Theory literature (see [53], [54]), we have not found a version needed for our specific purposes.

2.2.1 Stochastically Stable States

Let $P(0)$ be the 1-step transition probability matrix of a Markov chain on a finite state space S. We refer to this chain as the *unperturbed chain*.

Definition 2.2.1. *A regular perturbation of $P(0)$ consists of a stochastic matrix valued function $P(\epsilon)$ on some non-degenerate interval $(0, a]$ that satisfies, for all $x, y \in S$,*

1. *$P(\epsilon)$ is irreducible and aperiodic for each $\epsilon \in (0, a]$,*

2. $\lim_{\epsilon \to 0} P_{x,y}(\epsilon) = P_{x,y}(0)$ *and*

3. *if* $P_{x,y}(\epsilon) > 0$ *for some* ϵ, *then* $\exists\ r(x,y) \geq 0$ *s.t.* $0 < \lim_{\epsilon \to 0} \epsilon^{-r(x,y)} P_{x,y}(\epsilon) < \infty$.

An immediate consequence of the first requirement is that there exists a unique stationary distribution $\mu(\epsilon)$ satisfying $\mu(\epsilon)P(\epsilon) = \mu(\epsilon)$ for each $\epsilon \in (0, a]$. The other two requirements dictate the way the perturbed chain converges to the unperturbed one as $\epsilon \to 0$.

It follows that for a sufficiently small ϵ^*, $\exists\ 0 < \underline{\alpha}(x,y) < \overline{\alpha}(x,y) < \infty$, such that

$$\underline{\alpha}(x,y) < \epsilon^{-r(x,y)} P_{x,y}(\epsilon) < \overline{\alpha}(x,y),\ \forall\ \epsilon < \epsilon^*.$$

By denoting $\min_{x,y \in S} \underline{\alpha}(x,y) = \underline{\alpha}$ and $\max_{x,y \in S} \overline{\alpha}(x,y) = \overline{\alpha}$, we have, for any $x, y \in S$,

$$\underline{\alpha}\ \epsilon^{r(x,y)} < P_{x,y}(\epsilon) < \overline{\alpha}\ \epsilon^{r(x,y)},\ \forall\ \epsilon < \epsilon^*. \tag{2.4}$$

Let $\mathfrak{L} = \{f \in \mathcal{C}^\infty |\ f(\epsilon) \geq 0, f(\epsilon) = \sum_{i=1}^{L} a_i \epsilon^{b_i}$ for some $a_i \in \mathbb{R}, b_i \geq 0\}$ for some large enough but fixed $L \in \mathbb{N}$, where $\mathcal{C}^\infty$ is the space of smooth real valued functions. The following assumption will be invoked later.

Assumption 2. *For all* $x, y \in S$, $P_{x,y}(\epsilon) \in \mathfrak{L}$.

We develop some notation that will help state the main result regarding perturbed Markov chains. The parameter $r(x,y)$ in the definition of regular perturbation is called the *1-step transition resistance* from state x to y. Notice that $r(x,y) = 0$ only for the one step transitions $x \to y$ allowed under $P(0)$. A *path* $h(a \to b)$ from a state $a \in S$ to $b \in S$ is an ordered set $\{a = x_1, x_2, \ldots, x_n = b\} \subseteq S$

such that every transition $x_k \to x_{k+1}$ in the sequence has positive 1-step probability according to $P(\epsilon)$. The resistance of such a path is given by $r(h) = \sum_{k=1}^{n-1} r(x_k, x_{k+1})$.

Definition 2.2.2 (Resistance)**.** *The resistance from x to y is given by*

$$\rho(x, y) = \min\{r(h) |\ h(x \to y) \text{ is a path}\}.$$

Definition 2.2.3 (Co-radius)**.** *Given a subset $A \subset S$, its co-radius is given by*

$$CR(A) = \max_{x \in S \setminus A} \min_{y \in A} \rho(x, y).$$

Thus, $\rho(x, y)$ is the resistance of the least resistive path from x to y, and the co-radius of a set specifies the maximum resistance that must be overcome to enter it from outside[3]. We will extend the definition of resistance to include resistance between two subsets $S_1, S_2 \subset S$:

$$\rho(S_1, S_2) = \min_{x \in S_1, y \in S_2} \rho(x, y).$$

Since $P(\epsilon)$ is irreducible for $\epsilon > 0$, $\rho(S_1, S_2) < \infty$ for all $S_1, S_2 \subset S$.

Definition 2.2.4 (Recurrence classes)**.** *A recurrence or communication class of a Markov chain is a non-empty subset of states $E \subseteq S$ such that for any $x, y \in E$, $\exists\ h(x \to y)$ and for any $x \in E$ and $y \in S \setminus E$, $\nexists\ h(x \to y)$.*

Let us denote the recurrence classes of the unperturbed chain $P(0)$ as $E_1, ..., E_M$. Consider a complete directed graph $\mathcal{G}_{RC}$ on the vertex set $\{1, ..., M\}$ with each vertex corresponding to a recurrence class. Let a *j-tree* be a spanning subtree in $\mathcal{G}_{RC}$ that contains a unique directed path from each vertex in $\{1, ..., M\} \setminus \{j\}$ to j and denote the set of all j-trees in $\mathcal{G}_{RC}$ by $\mathcal{T}_{RC}^j$.

[3]These definitions are adopted from relevant literature [53], [54].

Definition 2.2.5. *The stochastic potential of a recurrence class E_i is*

$$\gamma(E_i) = \min_{T \in \mathcal{T}^i_{RC}} \sum_{(j,k)\in T} \rho(E_j, E_k).$$

Let

$$\gamma^* = \min_{E_i} \gamma(E_i).$$

We are now ready to state the main result regarding perturbed Markov chains.

Theorem 2 (Stochastic Stability; see [34], Theorem 4)**.** *Let $E_1, ..., E_M$ denote the recurrence classes of the Markov chain $P(0)$ on a finite state space S. Let $P(\epsilon)$ be a regular perturbation of $P(0)$ and let $\mu(\epsilon)$ denote its unique stationary distribution. Then,*

1. *as $\epsilon \to 0$, $\mu(\epsilon) \to \mu(0)$, where $\mu(0)$ is a stationary distribution of $P(0)$ and*

2. *a state is stochastically stable i.e. $\mu_x(0) > 0 \Leftrightarrow x \in E_i$ such that $\gamma(E_i) = \gamma^*$.*

2.2.2 Ergodicity with Time-Vanishing Perturbations

Let us begin with recalling the definition of ergodicity for a nonhomogeneous Markov chain on a finite state space S, with $Q(t)$ being the 1-step transition probability matrix at time t.

Definition 2.2.6 (Ergodicity)**.** *The chain is*

- *weakly ergodic (WE) if for all $t' \in \mathbb{N}$ and all $x, y, z \in S$,*

$$\lim_{t\to\infty} |Q^{(t',t)}_{x,z} - Q^{(t',t)}_{y,z}| = 0.$$

- *strongly ergodic (SE) if there exists a probability distribution π on S such that for any initial distribution $\eta(0)$ on S and any $t' \in \mathbb{N}$,*

$$\lim_{t\to\infty} \eta(0)Q^{(t',t)} = \pi.$$

We call π the limiting distribution of the chain.

Both definitions of ergodicity capture a certain notion of forgetfulness in that the chain forgets where it started after sufficiently large time steps. It is also clear that *SE* implies *WE*. Now, consider the nonhomogeneous Markov chain resulting from picking the ϵ along the evolution of $P(\epsilon)$ at time instant t as the corresponding element ϵ_t of the sequence $\{\epsilon_t\}_{t\in\mathbb{N}}$, where $\epsilon_t \to 0$ as $t \to \infty$. We henceforth refer to this sequence as the annealing schedule and the resulting Markov chain as the nonhomogeneous perturbed chain. Theorem 3 provides conditions on the annealing schedule that guarantee ergodicity of the nonhomogeneous perturbed chain with $\mu(0)$ (as in Theorem 2) being the limiting distribution. We denote the time-varying transition matrix of the nonhomogeneous perturbed chain by the bold-font $\mathbf{P}$, i.e. $\mathbf{P}(t) = P(\epsilon_t)$.

Define

$$\kappa = \min_{E\in\{E_i\}} CR(E). \tag{2.5}$$

Theorem 3 (Ergodicity Criterion)**.** *Let the recurrence classes of the unperturbed chain $P(0)$ be aperiodic and the parameter ϵ in the perturbed chain be scheduled according to the monotone decreasing sequence $\{\epsilon_t\}_{t\in\mathbb{N}}$, with $\epsilon_t \to 0$ as $t \to \infty$, as described above. Then, a sufficient condition for weak ergodicity of the resulting*

nonhomogeneous Markov chain $\mathbf{P}(t)$ *is*

$$\sum_{t\in\mathbb{N}} \epsilon_t^\kappa = \infty.$$

Furthermore, if the chain is weakly ergodic and Assumption 2 holds, then it is strongly ergodic with the limiting distribution being $\mu(0)$ *as described in Theorem 2.*

(See Appendix A.1 for proof.)

2.3 Analysis of the eITEL Algorithm

The objective of this Section is to prove Theorem 1. We will first consider eITEL with the parameter ϵ_t held constant at $\epsilon > 0$. It then describes a Markov chain on the finite state space $S = \mathcal{A} \times \{0,1\}^n$ and we denote its 1-step transition matrix as $P(\epsilon)$. The reason for choosing the same notation here as for the general perturbed Markov chain discussed in Section 2.2 is that we wish to view the Markov chain induced by eITEL as a perturbed chain and analyze it using results from Section 2.2. Similarly, $\mathbf{P}(t)$ denotes the 1-step transition probability matrix for the duration $(t, t+1)$ of the nonhomogeneous Markov chain induced by eITEL, i.e. with time varying ϵ_t. Henceforth, the components of any $x \in S$ will be identified with a superscript i.e. $x = [a^x, m^x]$.

Lemma 2.3.1. *The Markov chain* $P(\epsilon)$ *is irreducible and aperiodic.*

(See Appendix A.2 for proof.)

Lemma 2.3.1 implies that $P(\epsilon)$ has a unique stationary distribution which we denote, as in the previous section, by $\mu(\epsilon)$. It is also clear that $P(\epsilon)$ is a regular

perturbation of $P(0)$ (the latter obtained by setting $\epsilon_t \equiv 0$ in eITEL). Thus, by Theorem 2, $\mu(\epsilon) \to \mu(0)$ as $\epsilon \to 0$ where $\mu(0)$ is a stationary distribution of $P(0)$.

2.3.1 Stochastically Stable States - Support of $\mu(0)$

Consider the following subsets of the state space.

Definition 2.3.1. *Let*

$$C^0 = \{x \in S | m^x = \mathbf{1}\} \text{ and}$$

$$D^0 = \{x \in S | m^x = \mathbf{0}\}.$$

Lemma 2.3.2 (Recurrence Classes)**.** *If for every $a \in \mathcal{A}$, $\mathcal{G}_c(a) \cup \mathcal{G}_I(a)$ is strongly connected, the recurrence classes of the unperturbed chain $P(0)$ are D^0 and the singletons $z \in C^0$.*[4]

Proof. Consider transitions defined by eITEL with $\epsilon = 0$. Consequently, in Step 1, $\tilde{m}_i = (m_{t-1})_i \cdot \prod_{j \in \mathcal{N}_i(a_{t-1})} (m_{t-1})_j$. Every $z \in C^0$ satisfies $\mathbb{P}(\mathbf{X}_{t+1} = z | \mathbf{X}_t = z) = 1$ since by (2.1) the same joint action a^z is picked w.p. 1 resulting in the same payoff which in turn results in execution of step 4.1. A state $y \in D^0$ is also constrained to evolve only in D^0 since (2.3) with $\epsilon = 0$ does not permit a transition to $m_i = 1$ for any i. Also, by Assumption 1 and transition rule (2.2), there is a positive probability of transitioning from any joint action profile in $\mathcal{A}$ to any other. Thus D^0 and each $z \in C^0$ are recurrence classes of $P(0)$.

[4]Lemma 2.3.2 (and consequently Theorem 1) holds under weaker but more cumbersome to state condition that is analogous to *interdependence* in [32].

Now consider a state $x \in S \setminus \{C^0 \cup D^0\}$. Let $J^x = \{i | m_i^x = 0\} \subset \{1, ..., n\}$ be the non-empty subset of discontent agents. Since $\mathcal{G}_c(a^x) \cup \mathcal{G}_I(a^x)$ is strongly connected, there must exist an outward edge in this graph from at least one vertex $i' \in J^x$ to a vertex $i \notin J^x$. Two cases arise.

1. If (i', i) belongs to $\mathcal{G}_I(a^x)$, then $\exists\, b_{i'} \in \mathcal{A}_{i'}$ that agent i' can pick with positive probability according to (2.2) and due to Assumption 1, such that $f_i(a^x) \neq f_i(a^x_{-i'}, b_{i'})$. This changes the mood variable of agent i from 1 to 0 in step 4.2.

2. If (i', i) belongs to $\mathcal{G}_c(a^x)$, then agent i receives a 0 from its in-neighbor i' in step 4.1. Agent i sets $\tilde{m}_i = 0$ and consequently m_i is set to 0 in Step 4.3.

Thus x transitions to x' such that $|J^x| < |J^{x'}|$ i.e. at least one more agent becomes discontent with positive probability. Since there are finite number of agents and because of the strong connectivity assumption, repeating this argument for x' yields that there is a positive probability of transitioning from x to D^0; all agents eventually become discontent. Hence no state in $S \setminus \{C^0 \cup D^0\}$ is in a recurrence class. Since all these transitions are according to $P(0)$, we also have for any $y \in D^0$,

$$\rho(x, y) = 0,\ \forall\ x \in S \setminus C^0. \tag{2.6}$$

□

Guided by Theorem 2, we now proceed to calculate the stochastic potential of the recurrence classes of $P(0)$. But first we organize some calculations in the following lemma.

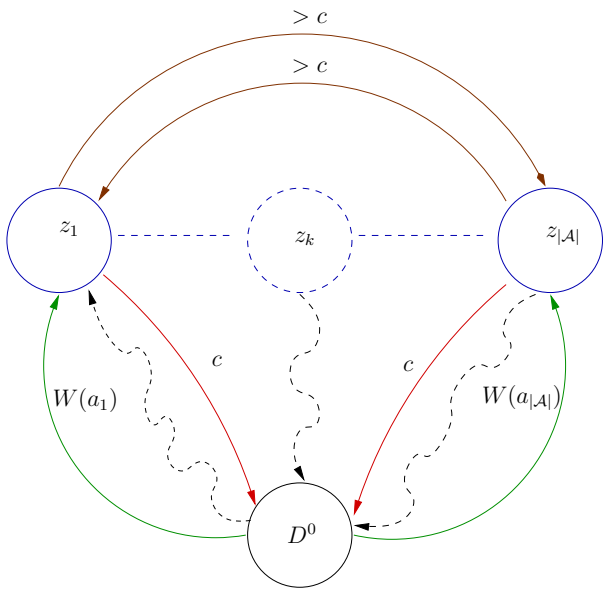


Figure 2.2: The circles represent recurrence classes of $P(0)$ and weights on the arrows the corresponding $\rho(\cdot,\cdot)$s. If $W(a^{z_1}) = W^*$, the zig-zag lines represent edges in the minimum resistance tree rooted at z_1.

Lemma 2.3.3. *Under the same assumption as Lemma 2.3.2, for any $y \in D^0$ and $z \in C^0$,*

$$\rho(x, y) = c, \ \forall\ x \in C^0, \tag{2.7}$$

$$\rho(y, z) = W(a^z), \tag{2.8}$$

$$\rho(x, z) \leq W(a^z), \ \forall\ x \in S \setminus C^0, \tag{2.9}$$

$$\text{and } \rho(z', z) > c, \ \forall\ z' \in C^0, \ z' \neq z. \tag{2.10}$$

Proof. Consider $x \in C^0$. For any i, a change in m_i^x from 1 to 0 must involve some agent picking a different action. From (2.1), such a change by an agent has resistance c. Therefore $\rho(x, y) \geq c$. Once such an action is picked by a content agent, its mood can change to 0 with a zero resistance transition in step 4.2. From

(2.6), this intermediate state can now move to $y \in D^0$ with zero resistance. Thus (2.7) is proved.

For any $y \in D^0$, any $h(y \to z)$ must undergo n discontent to content transitions according to (2.3) (because $m_i^y = 0 \Rightarrow \tilde{m}_i = 0$ in the ensuing iterate). Thus $\rho(y, z) \geq \sum_{i=1}^{n} f_i(a^z) = W(a^z)$. Since $\tilde{m}_i = 0$ for all i, from (2.2), all agents can collectively pick a^z via a zero resistance transition and become content with resistance $W(a^z)$. Thus there exists an $h(x \to z)$ such that $r(h) = W(a^z)$. Hence $\rho(x, z) = W(a^z)$ establishing (2.8). Then (2.9) follows in view of (2.6): consider $h(x \to y)$ followed by $h(y \to z)$.

For $z' \in C^0$, $z' \neq z$, there exists at least one agent playing different actions in the two states. Thus any $h(z' \to z)$, must involve this content agent picking a different action with resistance c (from (2.1)) and becoming content with resistance β_2 or $f_j(a^z)$ by step 4.2 or 4.3 respectively. Hence (2.10) is established. □

From Lemma 2.3.2, there are exactly $|\mathcal{A}| + 1$ recurrence classes of $P(0)$; $|\mathcal{A}|$ corresponding to each $a \in \mathcal{A}$ (i.e. each element of C^0) and one for the set D^0. Let $\{z_1, ..., z_{|\mathcal{A}|}\}$ be an enumeration for C^0.

Lemma 2.3.4. *Under the same assumption as Lemma 2.3.2, the stochastically stable set is* $\{z_i \in C^0 | W(a^{z_i}) = W^*\}$.

Proof. We will show that the minimum potential z-trees in $\mathcal{G}_{RC}$ are rooted at $\{z_i \in C^0 | W(a^{z_i}) = W^*\}$. The claim then follows as a consequence of Theorem 2. Consider Figure 2.2 which depicts edges of the $\mathcal{G}_{RC}$ corresponding to eITEL. The resistances between the recurrence classes are as calculated in Lemma 2.3.3. For $z_i \in C^0$,

consider any z_i-tree in this graph. Any such tree must have one outward edge from each of the $(|\mathcal{A}|-1)$ states in C^0 and an outward edge from D^0. The former contribute a resistance of at least $(|\mathcal{A}|-1)c$ and the latter $W(a)$ for some $a \in \mathcal{A}$, hence the least possible stochastic potential for a tree rooted at a state in C^0 is $(|\mathcal{A}|-1)c + W^*$. It is possible to construct such a tree for any state $z_i \in C^0$ with $W(a^{z_i}) = W^*$ as denoted by the zig-zag lines in Figure 2.2. By a similar argument, the stochastic potential of D^0 is $|\mathcal{A}| \cdot c$. Since $c > W^*$, any state $z_i \in C^0$ with $W(a^{z_i}) = W^*$ corresponds to the least stochastic potential state.

All that is left to prove is that any state $z_i \in C^0$ with $W(a^{z_i}) > W^*$ has stochastic potential greater than $(|\mathcal{A}|-1)c+W^*$. Again, consider any z_i-tree. If the outgoing edge from D^0 is incident on a state $z_k \in C^0$, $z_k \neq z_i$ with $W(a^{z_k}) = W^*$, there must exist at least one edge from a state in C^0 to z_i and $(|\mathcal{A}|-2)$ outward edges from the rest of elements of C^0 to complete the tree. Such a tree has resistance strictly greater than $(|\mathcal{A}|-1)c + W^*$ because of the link between two states in C^0. Else, if the outgoing edge from D^0 is to a state $z_k \in C^0$ with $W(a^{z_k}) > W^*$, the outward edges from the $(|\mathcal{A}|-1)$ states in C^0 result in a resistance at least greater than $(|\mathcal{A}|-1)c + W(a^{z_k})$. □

2.3.2 Ensuring Ergodicity with $\mu(0)$ as Limiting Distribution

Having established the required properties for the nonhomogeneous chain, we now turn to the analysis of the nonhomogeneous Markov chain, $\mathbf{P}$, induced by eITEL with the annealing schedule $\{\epsilon_t\}_{t\in\mathbb{N}}$. The proof relies on noting that if the annealing

schedule satisfies $\sum_{t=1}^{\infty} {\epsilon_t}^c = \infty$, $\mathbf{P}$ is strongly ergodic with the limiting distribution having support over states with efficient actions as described by Lemma 2.3.4.

Lemma 2.3.5. *Under the same assumption as Lemma 2.3.2, for the nonhomogeneous Markov chain defined on S by eITEL, κ as defined in* (2.5) *equals c.*

Proof. From Lemma 2.3.2 and (2.5),

$\kappa = \min\{\{CR(z)\}_{z\in C^0}, CR(D^0)\}$. For any $z \in C^0$, from (2.9) and (2.10), $CR(z) > c$. From (2.6) and (2.7), $CR(D^0) = c$. Hence $\kappa = c$. □

Proof of Theorem 1. The Assumption of Lemma 2.3.2 is included in the statement of the Theorem. All transition probabilities in eITEL belong to $\mathfrak{L}$; thus Assumption 2 holds. For any $y \in D^0$ and $z \in C^0$, $P_{y,y}(0) > 0$ and $P_{z,z}(0) > 0$. Hence the recurrence classes of the unperturbed Markov chain are aperiodic and, from Theorem 3 and Lemma 2.3.5, the chain is strongly ergodic if

$$\sum_{t=1}^{\infty} {\epsilon_t}^c \;=\; \infty. \tag{2.11}$$

Next, for any initial distribution η_0 on S and any subset $\tilde{S} \subset S$, $\mathbb{P}(\mathbf{X}_t \in \tilde{S}) = \sum_{j\in\tilde{S}} (\eta_0 \mathbf{P}^{(1,t)})_j$. Since (2.11) implies *SE* with limiting distribution $\mu(0)$ as in Theorem 2 and from the definition of *SE*, $\lim_{t\to\infty} \mathbb{P}(\mathbf{X}_t \in \tilde{S}) = \sum_{j\in\tilde{S}} \mu_j(0)$. Let $\tilde{S} = \{x \in S | W(a^x) = W^*, m^x = \mathbf{1}\}$, then in view of Lemma 2.3.4,

$$\lim_{t\to\infty} \mathbb{P}[\mathbf{a}_t \in \mathcal{A}^*] = 1.$$

□

2.4 Simulations

To illustrate the setup and the execution of the proposed algorithm, we consider the payoff structure in Table 2.1. As explained earlier, the agents do not know this structure, they can only pick actions simultaneously from $\mathcal{A}_i = \{l, h\}$ for $i = 1, 2, 3$ and measure the resulting payoffs. Let the agents implement eITEL of Section 2.1.2 to learn the welfare minimizing state. In this example, it is clear (at the level of the system designer) that agent 3's payoff depends only on its own actions and are unaffected by actions of agents 1 and 2. Thus, the interaction graph $\mathcal{G}_c(a)$ is not strongly connected for any a since there is no incident edge on node/agent 3. The plot in Figure 2.3 shows MATLAB simulation runs for eITEL. The plot on the top is for when $\mathcal{G}_c$ is empty (thereby violating the hypothesis of Theorem 1) and the one below for when $\mathcal{G}_c(a)$ consists of the directed edge $(1, 3)$ for all $a \in \mathcal{A}$ (thereby satisfying the hypothesis of Theorem 1). Observe that the instances where the product of the mood variables equals 1 can be interpreted as "conclusion of learning".

In the first case, since agent 3 cannot be influenced, it seems to learn to play h which offers it an individually rational lower payoff of $\frac{1}{10}$ as opposed to playing l with a payoff $\frac{1}{4}$. Quite intuitively, agents 1 and 2 seem to learn to play (h, h): the welfare minimizing action of the 'sub-game' where 3 chooses h with welfare $= \frac{2}{5}$ which is suboptimal to the global welfare minimal of $\frac{9}{20}$ achieved with (l, l, l). In the second case, it is observed that when the link $(1, 3)$ is added to $\mathcal{G}_c$, the agents lean to play the welfare minimal (l, l, l). An interesting question is how does the performance

Table 2.1: Payoff structure of a three agent system

Agent 3 →	l	l	h	h
Agent 2 →	l	h	l	h
Agent 1				
l	$(\frac{1}{10}, \frac{1}{10}, \frac{1}{4})$	$(\frac{1}{2}, 1, \frac{1}{4})$	$(\frac{3}{4}, \frac{3}{4}, \frac{1}{10})$	$(1, \frac{1}{2}, \frac{1}{10})$
h	$(1, \frac{1}{2}, \frac{1}{4})$	$(\frac{3}{4}, \frac{3}{4}, \frac{1}{4})$	$(\frac{1}{2}, 1, \frac{1}{10})$	$(\frac{1}{4}, \frac{1}{4}, \frac{1}{10})$

of eITEL depend on $\mathcal{G}_I$ and $\mathcal{G}_c$. We present results of some numerical experiments to motivate such questions. First of all, we quantify performance as the percentage of times the welfare minimal actions are played in a fixed duration. To analyze the effect of $\mathcal{G}_I$, consider n identical agents with $\mathcal{A}_i = \{0.1, 1\}$. Let us endow agent i with utility function $u_i(a) = \frac{1}{1+2q}\sum_{j=i-q}^{i+q} a_j$, where the operations in the limits of the summation are mod n. The welfare function $W(a) = \sum_{i=1}^{n} a_i,\ \forall\ q = 1, ..., n/2$, with a unique minimum at $(0.1, ..., 0.1)$. Notice that, for each q, $\mathcal{G}_I^q(a)$ is the same for all $a \in \mathcal{A}$ and can be varied by varying $q = 1, ..., n/2$. In Table 2.2 we report the performance for the case $n = 10$, $c = 1.1$, $\beta_C = \beta_I = 0.5$, $\epsilon_t = \frac{1}{\sqrt[c]{t}}$ and eITEL is allowed to run while $\epsilon_t > 10^{-4}$. The algorithm is implemented on MATLAB and the reported numbers are averaged over 100 runs for each value of q and the standard deviation is reported as well. Since a greater value of q can be interpreted as more complex interaction, the result seems to agree with the intuitive notion that the speed of convergence reduces with increased interaction complexity. To study the effect of $\mathcal{G}_c$, we use the same set up with $u_i(a) = a_{i-1}$ for $i = 2, ..., n$ and $u_1(a) = a_n$.

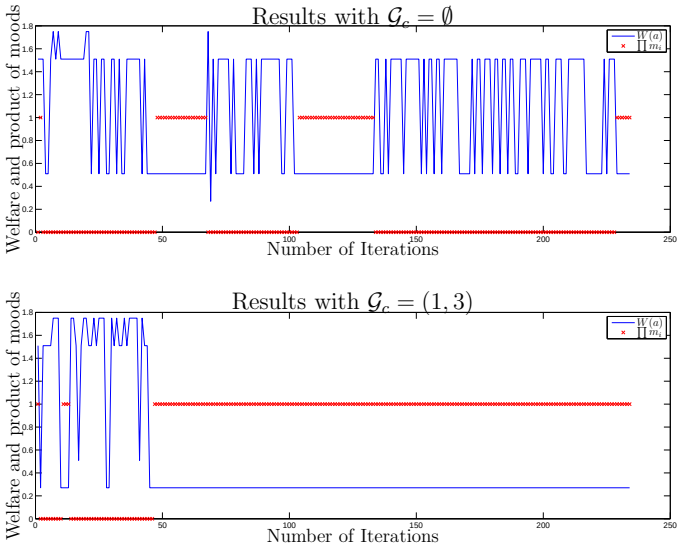


Figure 2.3: Simulation resuts for the three agent experiment; welfare plotted in blue solid lines and product of moods plotted with red crosses.

Thus $\mathcal{G}_I(a)$ is a directed ring for all $a \in \mathcal{A}$ (see Figure 2.4(a)). Let directed edges $(i, i-q)$ (where subtraction is mod n) for all i constitute $\mathcal{G}_c^q(a)$ for all $a \in \mathcal{A}$. Let $G(q) = \mathcal{G}_c^q \cup \mathcal{G}_I$. The same experiment as before is carried out with different values of q; the performance measure, the length of the longest shortest-path (SP) in $G(q)$ and length of the shortest cycle in $G(q)$ are plotted for values of q in $\{0, .., (n-1)\}$ in Figure 2.4(b). The results suggest a heuristic: To improve performance, pick $\mathcal{G}_c(a)$ to comprise of edges exactly opposite of $\mathcal{G}_I(a)$ and thereby reducing the cycle lengths.

2.5 Discussion

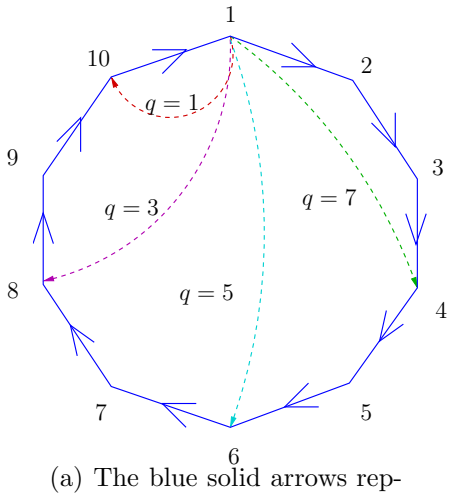
The modeling framework of interaction graph and communication graph presented in this chapter formalizes the intuitive notion of implicit communication via

Table 2.2: Effects of varying $\mathcal{G}_I^q$

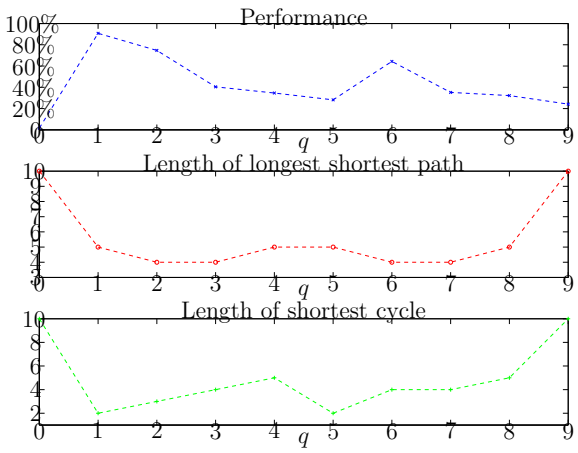
q	Performance	Std. Deviation
1	93.78%	2.92%
2	62.21%	7.84%
3	48.15%	9.71%
4	45.35%	11.11%
5	44.31%	11.79%

the system (individual utility functions in this setup) and explicit communication. What is noteworthy is that the first requirement of Theorem 1 resonates with the intuition that, in general, to perform a collaborative (global) task, each individual in the group must have access to some (non-local) information - implicitly or explicitly. While we do not make a formal claim here, it is a reasonable conjecture that if an agent can neither measure a change in its utility as a result of the others' actions nor can it receive an explicit communication, simple utility structures can be conceived where *any* decentralized algorithm would fail to achieve the stated global objective (recall the payoff structure in Table 2.1). In this sense, heuristically speaking, eI-TEL appears to require certain 'minimal' explicit communications when there isn't enough implicit communication available.

Before concluding this chapter we wish to point out that $\beta_C, \beta_I > 0$ are free parameters that can be tuned to possibly get improved performance. These parameters can also be interpreted, in some sense, as weights on the communication and interaction graphs as larger values of β_C and β_I correspond to agents being



(a) The blue solid arrows represent $\mathcal{G}_I$ and the dotted arrows denote the edge $(1, 1-q)$ in the corresponding $\mathcal{G}_c^q$.



(b) Plot of performance, longest SP and cycle length w.r.t. different values of q.

Figure 2.4: Effects of varying $\mathcal{G}_c$ for $n = 10$.

more sensitive to the information from the communication and interaction graphs respectively (see steps 1.3 and 4.2 of the algorithm).

An important open question is determining the rate of convergence of eITEL. One way to answer this question is to calculate the rate of convergence of $||\eta_t - \mu(0)||$ as $t \to \infty$, where η_t is the density of $\mathbf{X}_t = [\mathbf{a}_t, \mathbf{m}_t]$. This is difficult since the Markov chain is nonhomogeneous and the best results we know in such situations are for the simulated annealing algorithm [55]. It should be pointed out that the rate condition on the annealing schedule $\{\epsilon_t\}$ in Theorem 1 tells little about the rate of convergence of eITEL or its finite-time behavior (see [52], Chapter 5 for a discussion on the analogous question of annealing schedules for fast convergence in simulated annealing).

Chapter 3: Ideas from Extremum Seeking Control

3.1 Main Idea

The main contribution of this chapter is a solution to problem (WO) using ideas from extremum seeking control. The motivation for this approach is to exploit the gradient information available in the continuous formulation of problem (WO) of Section 1.2 in order to get improved convergence speed. The algorithm developed is called the *Collaborative Extremum Seeking* (CES) algorithm, and a rigorous justification for local convergence of the agent actions $u(t)$ (when following CES) to an arbitrarily small neighborhood of a minimizer of $W(\cdot)$ as $t \to \infty$ (see Theorem 5) is provided. Extremum seeking control is a model-free adaptive control technique that performs a simultaneous gradient estimation-descent (see [38] for an overview and [39] for a detailed analysis). Loosely speaking, our solution is based on the agents running a dynamic average consensus algorithm with their payoffs as inputs so that the consensus output "tracks" $W(u)$. Next, this arrangement is interfaced suitably with an extremum seeking loop (see Figure 3.1) that leads agent i's action to evolve according to

$$\dot{u}_i \approx -\frac{\partial W(u)}{\partial u_i}.$$

In this sense, this work can be considered as an application of extremum seeking control and dynamic consensus for solving (WO). Noteworthy in this context are recent works [10,11] which apply extremum seeking control to learn Nash equilibria in non-cooperative $n-$player games.

The chapter is organized as follows. Section 3.2 discusses a dynamic average consensus algorithm. We provides a novel analysis for this consensus algorithm which, while necessary for the following discussion, we believe is an elegant treatment of the topic. The proposed solution to (WO) and related results are discussed in Section 3.3. Numerical simulations on the wind farm problem are presented in Section 3.4.

3.2 Dynamic Average Consensus Revisited

Consider a MAS comprising n agents indexed by i. Agent i has access to a reference signal r_i, and our goal is to devise a dynamic average consensus algorithm that helps every agent track the average of the references $\frac{1}{n}\sum_{i=1}^{n} r_i$. This algorithm has appeared earlier in literature as 'Proportional-Integral Dynamic Consensus' in [56]; however, our analysis is quite different from this earlier work and is inspired by [57] where distributed optimization is treated along similar lines.

We begin by stating the communication requirements between the agents. It is assumed that the agents exchange information over a connected, undirected communication graph $\mathcal{G}_c = (V, E)$, where $V = \{1, ..., n\}$, and $(i, j) \in E (\Leftrightarrow (j, i) \in E)$ means that agents i and j can exchange information. An adjacency matrix A of $\mathcal{G}_c$

satisfies $(i,j) \notin E \Rightarrow A[i,j] = 0$; and the corresponding Laplacian matrix L is given by $L = diag(A\mathbf{1}) - A$[1]. It is clear that $L\mathbf{1} = 0$; however, when $rank(L) = (n-1)$, then $Lx = 0 \Leftrightarrow x = \alpha\mathbf{1}$ for some $\alpha \in \mathbb{R}$. A rank $(n-1)$ Laplacian can always be constructed for a connected graph by picking elements of the adjacency matrix such that $(i,j) \in E \Rightarrow A[i,j] > 0$ (this follows from the Perron-Frobenious Theorem [58]).

Now, consider the following optimization problem for the scalar variable $\hat{x}$:

$$\min_{\hat{x}} \frac{1}{2}\sum_{i=1}^{n}(\hat{x} - r_i)^2. \tag{P0}$$

Elementary calculations yield the optimizer to be $\frac{1}{n}\sum_{i=1}^{n} r_i$. Next, consider the following equivalent restatements of (P0):

$$\begin{aligned} &\min \frac{1}{2}\sum_{i=1}^{n}(x_i - r_i)^2 \text{ subject to } \; x_i = x_j, \; \forall \; i,j; \\ &\min \frac{1}{2}x^T x - r^T x + \frac{1}{2}r^T r \text{ subject to } L_I x = 0, \end{aligned} \tag{P1}$$

where L_I is a Laplacian matrix of $\mathcal{G}_c$ with $rank(L_I) = (n-1)$ (as noted above, this essentially enforces $x_i = x_j, \; \forall \; i,j$). Finally, consider the following equivalent restatement of (P1):

$$\min \frac{1}{2}x^T x - r^T x + \frac{\rho}{2}x^T L_P x \text{ subject to } L_I x = 0, \tag{P2}$$

where L_P is a Laplacian matrix of $\mathcal{G}_c$. The last equivalence follows by noting that the constraint $L_I x = 0 \Rightarrow x = \alpha\mathbf{1} \Rightarrow x^T L_P x = 0$, so that (P2) is exactly the same as (P1). Addition of $x^T L_P x$ to the objective (with an appropriate choice of L_P)

[1]Recall, for a vector v, $diag(v)$ denotes the diagonal matrix with $diag(v)[i,i] = v(i)$, and the bold-font $\mathbf{1}$ stands for the vector $[1,...,1]^T$.

amounts to adding a penalty term for constraint violation that can help improve the speed of numerical algorithms for finding the solution.

Denote the Lagrangian corresponding to (P2) by $\mathcal{L}(x,\lambda) = \frac{1}{2}x^T(I+\rho L_P)x - r^T x + \lambda^T L_I x$, where λ is the dual variable. It is a well known fact from optimization theory that the solution x^* to (P2) corresponds to a saddle point (x^*,λ^*) of $\mathcal{L}(x,\lambda)$,

$$\mathcal{L}(x^*,\lambda) \leq \mathcal{L}(x^*,\lambda^*) \leq \mathcal{L}(x,\lambda^*).$$

Thus, consider the saddle-seeking system

$$\begin{aligned}\begin{bmatrix} \dot{x} \\ \dot{\lambda} \end{bmatrix} &= \begin{bmatrix} -\nabla_x \mathcal{L}(x,\lambda) \\ \nabla_\lambda \mathcal{L}(x,\lambda) \end{bmatrix} \\ &= \begin{bmatrix} -I-\rho L_P & -L_I^T \\ L_I & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} + \begin{bmatrix} I \\ 0 \end{bmatrix} r. \end{aligned} \tag{3.1}$$

While r is a fixed vector in the context of the optimization problems (P0), (P1) and (P2), in what follows we will also treat the system in (3.1) as an LTI system driven by the input r. Notice that the update rules for $\dot{x}_i$ and $\dot{\lambda}_i$ in (3.1), depend only on the values of x_j, λ_j and $L_I[j,i]$ such that $(i,j) \in E$ (apart from the i^{th} rows of L_I and L_P, and r_i which agent i already knows). Thus, (3.1) can be implemented in a distributed manner by means of exchange of appropriate variables and data over $\mathcal{G}_c$. So far the introduction of L_I and L_P only made sense due to their algebraic properties. The reason for requiring them to be Laplacians of $\mathcal{G}_c$ is to obtain such distributed implementation of (3.1).

By now the reader would have guessed that (3.1) is the proposed dynamic consensus algorithm, with the individual x_is expected to converge to $\frac{1}{n}\sum_{i=1}^n r_i$ (the

solution of (P0)), for fixed values of r_is. The remainder of this section deals with formalizing this statement. Let μ be such that $\mu^T L_I = 0$, and let $U = [\tilde{U}\ \mu]$ be an orthonormal matrix. Replace λ in (3.1) with $\lambda = U \begin{bmatrix} \tilde{\lambda} \\ \lambda_\mu \end{bmatrix}$, where $\tilde{\lambda} \in \mathbb{R}^{n-1}$ and $\lambda_\mu \in \mathbb{R}$, to obtain

$$\begin{bmatrix} \dot{x} \\ \dot{\tilde{\lambda}} \\ \dot{\lambda}_\mu \end{bmatrix} = \left[\begin{array}{cc|c} -I - \rho L_P & -L_I^T \tilde{U} & 0 \\ \tilde{U}^T L_I & 0 & 0 \\ \hline 0 & 0 & 0 \end{array}\right] \begin{bmatrix} x \\ \tilde{\lambda} \\ \lambda_\mu \end{bmatrix} + \left[\begin{array}{c} I \\ 0 \\ \hline 0 \end{array}\right] r. \tag{3.2}$$

Since λ_μ does not interact with the rest of the state and inputs (and vise versa) in (3.2), one can eliminate it to obtain the reduced-order system

$$\begin{bmatrix} \dot{x} \\ \dot{\tilde{\lambda}} \end{bmatrix} = \begin{bmatrix} -I - \rho L_P & -L_I^T \tilde{U} \\ \tilde{U}^T L_I & 0 \end{bmatrix} \begin{bmatrix} x \\ \tilde{\lambda} \end{bmatrix} + \begin{bmatrix} I \\ 0 \end{bmatrix} r. \tag{3.3}$$

The following lemma is easily verified.

Lemma 3.2.1. *Let $y = x$ be the output of (3.1), and $y_{ro} = x$ be that of (3.3). Let the same input $r : [0, T] \to \mathbb{R}^n$ be applied to both systems (3.1) and (3.3), with the same initial conditions for the variable x, and*

- *for a specified initial condition $\lambda(0) \in \mathbb{R}^n$, pick $\tilde{\lambda}(0) = \tilde{U}^T \lambda(0)$; or*
- *for a specified initial condition $\tilde{\lambda}(0) \in \mathbb{R}^{n-1}$, pick $\lambda(0) = \tilde{U}\tilde{\lambda}(0)$.*

Then

$$y(t) = y_{ro}(t),\ \forall\, t \in [0, T].$$

The reason for introducing the reduced order system (3.3) is twofold. First, the properties of the equilibrium point of (3.3) are helpful in proving Theorem 4.

Second, it turns out that the global asymptotic stability of the LTI system (3.3) (as opposed mere stability of (3.1)) will be helpful in the analysis of the next section. We will now prove the main convergence result for the dynamic average consensus algorithm (3.1). We will need the following technical lemma.

Lemma 3.2.2. *Let $Q \in \mathbb{R}^{n\times n}$ be such that $Q^T + Q > 0$, and $R \in \mathbb{R}^{n\times m}$ $(m < n)$ be full rank. Then the matrix*

$$M = \begin{bmatrix} -Q & -R \\ R^T & 0 \end{bmatrix}$$

is Hurwitz.

(See Appendix B.1 for proof.)

Theorem 4 (Dynamic Consensus Convergence Criterion)**.** *Let $\mathcal{G}_c$ be a connected, undirected graph, L_I and L_P be Laplacians of $\mathcal{G}_c$ such that $rank(L_I) = (n-1)$, and ρ be such that $\frac{1}{2}\rho\lambda_{min}(L_P + L_P^T) < 1$. Then, for any $w \in \mathbb{R}^n$ and $r(t) = w \;\forall\; t \geq 0$,*

1. *the LTI system (3.3) has a unique globally exponentially stable equilibrium point $(x^{eq}(w), \tilde{\lambda}^{eq}(w))$, where $x_w = \frac{1}{n}(\mathbf{1}^T w)\mathbf{1}$, and*
2. *the state of the LTI system (3.1), with arbitrary initial conditions $x(0), \lambda(0) \in \mathbb{R}^n$, remains bounded and $x(t)$ converges exponentially to $\frac{1}{n}(\mathbf{1}^T w)\mathbf{1}$ as $t \to \infty$.*

Proof. Notice that if $\frac{1}{2}\rho\lambda_{min}(L_P + L_P^T) < 1$, then $2I + \rho(L_P + L_P^T) > 0$. Since $rank(L_I) = (n-1)$ and U is orthonormal, $U^T L_I = \begin{bmatrix} \tilde{U}^T L_I \\ 0 \end{bmatrix} \Rightarrow L_I^T \tilde{U} \in \mathbb{R}^{n\times n-1}$ is

full rank. Thus, by Lemma 3.2.2, the matrix

$$A_{ro} = \begin{bmatrix} -I - \rho L_P & -L_I^T \tilde{U} \\ \tilde{U}^T L_I & 0 \end{bmatrix}$$

is Hurwitz.

Next, notice that (P2) is a convex optimization problem (in particular, a quadratic program). Since $L_I x = 0 \Leftrightarrow \tilde{U}^T L x = 0$, the equality constraint in (P2) can be replaced with $\tilde{U}^T L x = 0$. The corresponding Lagrangian, with $\tilde{\lambda}$ as the Lagrange multiplier, is given by $\mathcal{L}(x, \tilde{\lambda}) = \frac{1}{2} x^T (I + \rho L_P) x - r^T x + \tilde{\lambda}^T \tilde{U}^T L_I x$. With $r = w$, the KKT conditions for this convex problem state that $x^{\text{eq}}(w)$ is optimal *if and only if* $\exists\, \tilde{\lambda}^{\text{eq}}(w)$ such that

$$\begin{bmatrix} -I - \rho L_P & -L_I^T \tilde{U} \\ \tilde{U}^T L_I & 0 \end{bmatrix} \begin{bmatrix} x^{\text{eq}}(w) \\ \tilde{\lambda}^{\text{eq}}(w) \end{bmatrix} + \begin{bmatrix} w \\ 0 \end{bmatrix} = 0. \tag{3.4}$$

Since A_{ro} is Hurwitz, there exists a unique solution $(x^{\text{eq}}(w), \tilde{\lambda}^{\text{eq}}(w))$ to (3.4). Further, subtracting (3.4) from (3.3) (with $r(t)$ set to w), we note the error variables $\begin{bmatrix} x - x^{\text{eq}}(w) \\ \tilde{\lambda} - \lambda^{\text{eq}}(w) \end{bmatrix}$ converge to zero exponentially since A_{ro} is Hurwitz. Finally, since (P2) is a restatement of (P1), which in turn is a restatement of (P0), we have $x^{\text{eq}}(w) = \frac{1}{n}(\mathbf{1}^T w)\mathbf{1}$. This concludes the proof for the first claim.

The second claim regarding boundedness of the state of (3.1) follows by noting that by performing a similarity transformation of the state in (3.1), a mode with zero eigenvalue (λ_μ) has been isolated in (3.2) and shown to be unaffected by the input, while the rest of the state is Input-to-State-Stable as A_{ro} is Hurwitz. The exponential convergence $x(t) \to \frac{1}{n}(\mathbf{1}^T w)\mathbf{1}$ is due to Lemma 3.2.1 and the first claim

of this theorem. □

Hence, we have argued the correctness of the dynamic consensus algorithm (3.1) by associating it to an optimization problem and avoided the tedious calculations needed in the proof of [56, Theorem 5].

3.3 The CES Algorithm

In this section we discuss in detail the proposed Collaborative Extremum Seeking (CES) algorithm. We assume the agents can exchange information over a communication graph $\mathcal{G}_c$ as described in Section 3.2. Consider the schematic representation shown in Figure 3.1. Agent i applies its input u_i and receives the corresponding payoff or utility value $f_i(u)$. The agent uses this observed/measured value as the reference command r_i in the dynamic consensus algorithm. Finally, the 'consensus output' x_i is used as the feedback to the extremum seeking block which, in turn, generates u_i, thus closing the loop. Observe that variables in agent i's 'loop' (highlighted in Figure 3.1) comprise of only variables measured, controlled or communicated to it over $\mathcal{G}_c$, leading to the required distributed implementation. Within the extremum seeking blocks, agents use their respective sinusoidal 'dither' signal $\nu_i(t) = \sin(\omega_i t + \phi)$ and certain parameters which satisfy

$$\begin{aligned} \omega_i &= \omega_c \overline{\omega}_i, \\ a_i &= a_c \overline{a}_i, \end{aligned} \tag{3.5}$$

where, for all i, ω_c, a_c and $a_i \in \mathbb{R}$ are positive, and ω_i is a positive rational number. The constant $k_i = \omega_c K_i$, for some $K_i > 0$.

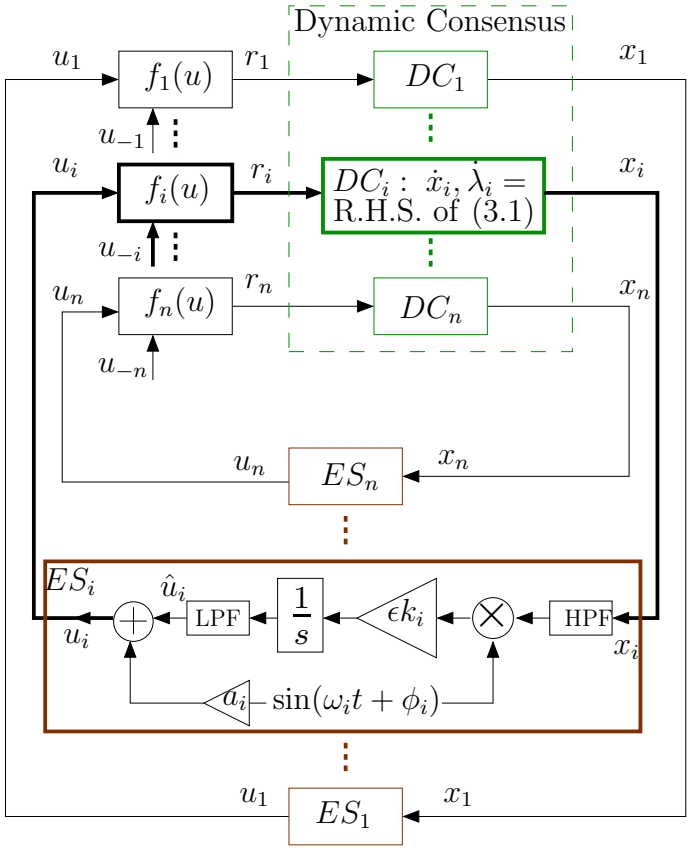


Figure 3.1: A schematic representation of the CES algorithm. DC_i refers to part of the dynamic consensus algorithm (3.1) implemented by agent i, ES_i refers to the extremum seeking law implemented by agent i, and u_{-i} refers to the elements of the vector u other than u_i.

The system depicted in Figure (3.1) is given by[2]

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{\lambda} \end{bmatrix} &= \begin{bmatrix} -I - \rho L_P & -L_I^T \\ L_I & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} + \begin{bmatrix} f(u) \\ 0 \end{bmatrix}, \\ \dot{\hat{u}}_i &= -\epsilon k_i \nu_i x_i, \ \forall \ i, \\ u &= \hat{u} + a\nu(t); \end{aligned} \tag{3.6}$$

[2]We ignore the low-pass filters (LPF) and the high-pass filters (HPF) shown in Figure 3.1 since these are not needed for the convergence result of this section and including them makes the discussion cumbersome due to additional states. We will use these in the simulations of Section 3.4.

where $f(\cdot) = [f_1(\cdot), ..., f_n(\cdot)]^T$ and, by slight abuse of notation, $a\nu(t) = [a_1\nu_1(t), ..., a_n\nu_n(t)]^T$.

Assumption 3 (Existence of Local Minimizer)**.** *The function $f_i(\cdot)$ is smooth for each i, and there exists $u^* \in \mathbb{R}^n$ such that $\frac{\partial W(u^*)}{\partial u} = 0$ and $\frac{\partial^2 W(u^*)}{\partial u^2} > 0$.*

This assumption ensures existence of a strict local minimizer for W. We are now ready to state the main result of this section which provides conditions that guarantee local convergence of the variable u to a neighborhood of u^*.

Theorem 5 (CES Local Convergence)**.** *Let the hypothesis of Theorem 4 and Assumption 3 hold. Let $\omega_i \neq \omega_j$, $2\omega_i \neq \omega_j$ and $\omega_i \neq \omega_j + \omega_k$, for all distinct $i, j, k \in \{1, ..., n\}$[3]. Then there exists $\omega_c^* > 0$ such that for any $\omega_c \in (0, \omega_c^*)$, there exist $\epsilon^* > 0$ and $a_c^* > 0$, such that for the chosen ω_c, and any $\epsilon \in (0, \epsilon^*)$ and $a_c \in (0, a_c^*)$, the solution to (3.6) with initial conditions $\hat{u}(0)$ sufficiently close to u^* (and $x(0), \lambda(0)$ arbitrary) is such that $u(t)$ converges exponentially to an $O(\omega_c + \epsilon + a_c)$ neighborhood of u^*.*

The proof can be found in Appendix B.2. It uses averaging and singular perturbation arguments that are standard in proving local convergence of extremum seeking schemes [39]. While arguments similar to that in [10] are used, certain subtleties prevent direct use of results from the extremum seeking control literature. It is worth mentioning that the 'exponential stability' of the reduced order system (3.3) (proved in Section 3.2) plays a key role in the analysis of (3.6) by enabling the use of relevant singular perturbation results (which require such exponential stability).

[3]For $n = 2$, drop conditions that refer to ω_k.

3.4 Simulations: Wind Farm Power Maximization

In this section we explore the applicability of the CES algorithm to the problem of maximizing the total power production of a wind farm via numerical simulations. The approach is to build a simulation model of a wind farm based on a model for computing wake affected wind speeds and a model for power production of turbines. Next, we run simulations of our model-free CES algorithm (3.6) on this simulation model and evaluate its performance (as though the model were the "ground truth"). Since the details of the models are not directly related to the main theme of the thesis, we will only briefly describe the models used and will provide appropriate references for further details. Note that application of extremum seeking control for wind energy has appeared elsewhere in literature; [24] investigates a centralized controller for farm-level power maximization and [59] for turbine-level power maximization.

3.4.1 Wind Farm Model [25]

The action variable u_i of a turbine is its *axial induction factor* which takes values in $[0, 1/2]$. The power produced is given by the model

$$f_i(u) = \frac{1}{2}\rho_{\text{air}} A_i C_p(u_i) V_i(u)^3,$$

where $C_p(u_i)$ is the power efficiency coefficient and is modeled as $C_p(u_i) = u_i(1-u_i)^2$, $V_i(u)$ is the wind speed at turbine i in m/s, ρ_{air} is the density of air and is fixed at the constant value $1.225 kg/m^3$, and A_i is the area swept by the blades of turbine i

in m^2. We assume all turbines in the farm to be identical with diameters equal to $77m$. It can be seen from the power capture expression that the influence of other turbine's actions on turbine i's power production (or utility) is due to the $V_i(u)$ term. This term captures the effect of aerodynamic interaction between turbines and we will use analytical models to describe it.

Consider a wind farm with a given layout and a given wind direction. Let us suitably enumerate the turbines so that the coordinates of the turbines $\{(x_1, y_1), ..., (x_n, y_n)\}$ are such that $x_j < x_i$ implies turbine i is downstream from turbine j. Then the wind speed at turbine i is given by

$$V_i(u) = V_\infty \left(1 - \sqrt{\sum_{j \in \{1,...,n\}: x_j < x_i} (u_j C[j,i])^2}\right),$$

where V_∞ is the free stream wind speed in m/s, and the matrix $C \in \mathbb{R}^{n \times n}$ depends on the farm layout. The model used for computation of the C matrix is the popular low-fidelity wake model called the *Park Model* that can be found in references [21, 23–25]. In the simulation results reported in this section, we set $V_\infty = 10m/s$, $n = 3$, consider the three turbines located at coordinates $\{(0,0), (0,5D), (0,10D)\}$ (where D is the turbine diameter), and consider the wind blowing along the positive horizontal axis. Specifically, with regard to reference [25], the roughness coefficient 'k' used in computation of the C matrix is set to 0.04.

3.4.2 Simulation Results

A Simulink model is developed for implementing the CES algorithm in Figure 3.1 with the functions $f_i(\cdot)$ modeled as described above. The parameters for the

extremum seeking loop are chosen as $\omega = [1, 1.3\bar{3}, 1.90476]^T$, $\omega_c = 1$, $a_c = 5 \cdot 10^{-2}$, $a = [1, 1, 1]^T$, $\epsilon = 5 \cdot 10^{-1}$, and ϕ is chosen at random. It is typical for each turbine to be set to maximize its power production in present-day wind farm operations, and this corresponds to $u_i = \frac{1}{3}$. We shall think of this setting as the 'baseline' and compare the performance of our solution against it. Also, unless stated otherwise, reported simulations are run with initial conditions $\hat{u}(0) = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]^T$, and with $x(0), \lambda(0)$ set to zero. The value of gains K_i are set to the inverse of the baseline power of turbine i, $(f_i(\frac{1}{3}))^{-1}$. The filters depicted in Figure 3.1 are implemented using simple first order filters

$$\text{LPF: } \frac{1}{s + \omega_l}, \quad \text{HPF: } \frac{s}{s + \omega_h},$$

where $\omega_l = \min\{\omega\}$ and $\omega_h = 1.5 \cdot \max\{\omega\}$. Finally, we choose $L_p = 0$ and

$$L_I = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}.$$

Figure 3.2 shows time traces of relevant variables when simulations are run with the aforementioned parameters. It can be seen that the input variables $u(t)$ and total farm power $W(u(t))$ continue to oscillate around constant values. Such oscillations are to be expected in any extremum seeking scheme since the dither $a_i\nu_i(t)$ is injected additively in $u_i(t)$. The performance of CES is better understood by plotting the "learning variable" $\hat{u}$. In Figure 3.3 we plot the evolution of $(\hat{u}_1, \hat{u}_2)$ starting with initial condition $\hat{u}(0) = [0, 0, 0]^T$ over contours of $W(\cdot)$ (the variable $\hat{u}_3$, excluded from the plot, converges to $\frac{1}{3}$; this is optimal $(= u_3^*)$ for the most downstream

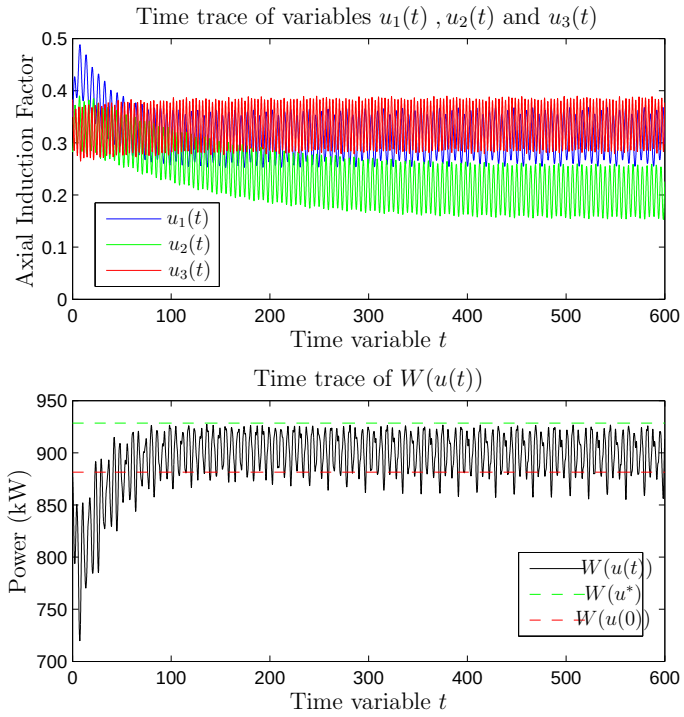


Figure 3.2: A typical run of CES on a three turbine wind farm model

turbine which only has to maximize its individual power regardless of what the other turbines do). CES appears to perform very well, and $(\hat{u}_1, \hat{u}_2)$ converge to a small neighborhood of the optimal values (u_1^*, u_2^*) (computed by using standard optimization software on the model).

3.4.3 Consensus Time Scale vs. Learning Time Scale

The choice of the parameters made so far are somewhat arbitrary. Indeed, an accurate choice of the parameters in the dynamic consensus stage require additional hardware specific information. To put things in perspective, we repeat the simulation in Section 3.4.2 for different values of time-scale separation between the learning dynamics and the consensus dynamics. This is done by multiplying the

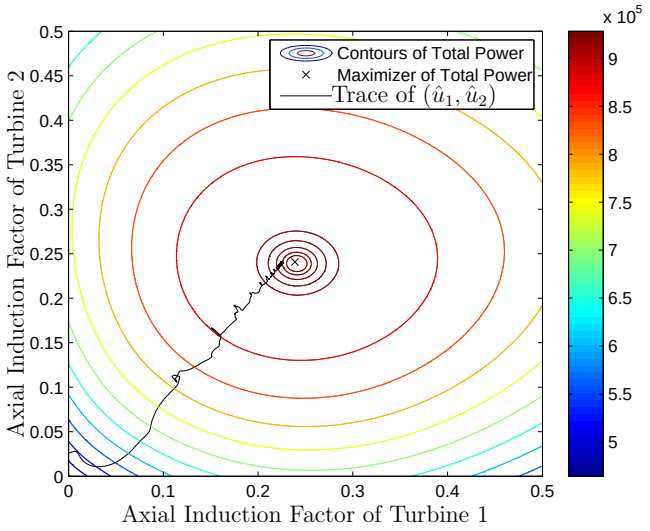


Figure 3.3: Trace of $(\hat{u}_1, \hat{u}_2)$ plotted over contours of total power $W(\cdot)$ (in Watts)

R.H.S. of the consensus part of dynamics in (3.6) by a positive constant α_{TS}. This effectively translates to the consensus dynamics (3.1) running at a time scale $\alpha_{\text{TS}} \cdot t$ when compared to the learning dynamics of $\hat{u}$ that run at time scale t. In Figure 3.4 we plot the performance of the algorithm with varying values of α_{TS} (all other parameters are kept fixed during these runs). Results show that so long as the consensus algorithm runs at least an order of magnitude faster than the learning dynamics, learning is successful. The degraded performance observed for small values of α_{TS} is expected since in the absence of information exchange, each turbine tries to maximize its power production leading to convergence to a neighborhood of the 'Nash equilibrium' $[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$; reminiscent of the Nash-seeking result in [10].

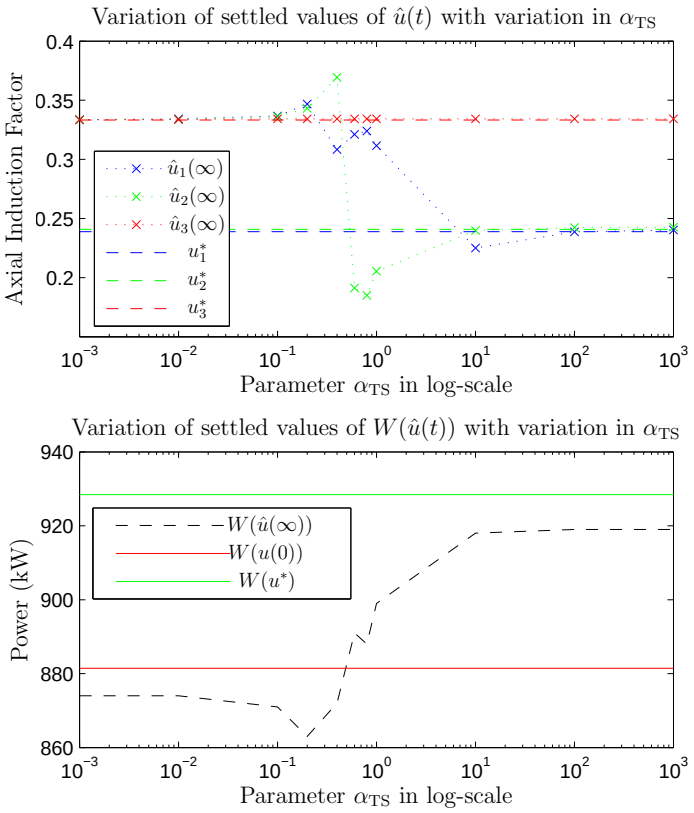


Figure 3.4: Performance across varying learning and consensus time scales

3.5 Discussion

In view of the observed non-local convergence properties of CES in Figure 3.3, a natural extension of the local convergence result Theorem 5 is a semi-global practical stability analysis for the CES algorithm. Such analysis is well known in extremum seeking control (see [60]). Also, extending the convergence results to the more realistic case with agents modeled by nonlinear dynamics is forthcoming. From an implementation standpoint, an important next-step is to obtain a discrete-time counterpart to CES, and results from discrete time extremum seeking control are expected to be helpful in this regard. Effects of actuator saturation (or a setting

where u_is are constrained to bounded intervals) on the convergence properties must also be investigated.

An interesting direction of research would be to understand the effects of the structure of the communication graph on the convergence speed of CES. The effects of a time varying communication graph must also be addressed. Other dynamic average consensus algorithms in literature can also be potentially used in the proposed scheme and similar analysis can be performed to justify convergence.

Other application scenarios, especially in the area of collaborative robotics will be investigated in future work. In the reported simulation study of Section 3.4, the delay between the change in a turbine's action and the measurement of its consequences by a downstream turbine are ignored (see [23, 24][4]). The proposed algorithm will be tested on higher fidelity wind farm models in future work.

[4]Such a delay improves the accuracy of the wind farm model since in a real wind farm the wake travels at the speed of wind which takes some non-zero time to reach from one turbine to the other.

Chapter 4: Effects of Information Exchange Pattern: Vehicle Platooning Case

This chapter deals with a different problem than that in the last two chapters. In Section 1.2.1 we briefly described a broad question regarding the effects of the information exchange network (between different agents in a MAS) on the performance of the MAS distributed control/decision/optimization task. While we discussed this question in the previous two chapters in the context of the eITEL and CES algorithms, in this chapter we explore this question in the context of a specific MAS (not related to (WO)) and show a concrete example of the approach described in Section 1.2.1.

Distributed control of large vehicle platoons has been an active area of research in the control community. In particular, the one dimensional variant of the problem has received special interest due to potential applications in increasing throughput of Intelligent Vehicle/Highway Systems(IVHS) (see [41]). The idea is that some lead vehicles in the platoon are given desired trajectory information by a supervisory layer in the IVHS and the whole platoon follows this desired trajectory while maintaining prescribed safe inter vehicle spacing. High speeds and close inter-vehicle spacings naturally call for automatic control. The challenge is to design distributed control

laws for vehicles in the platoon that regulate the inter-vehicle spacings while each vehicle (agent) acts only on the information available to it.

Several researchers have analyzed this scenario and we discuss only some of the results here. A double integrator model for individual vehicle dynamics is a common abstraction. Using identical double integrator dynamics for all vehicles in the platoon, [42] and [43] analyze the problem in the frequency domain. They consider the cases where the local information available to a vehicle is limited to the (sensed) relative distance from its predecessor and from both predecessor and follower. For both cases they conclude that the H_∞ norm of the transfer function from disturbances acting on the vehicles to the spacing errors and of that between the spacing errors grow without bounds in the size of the platoon. In a similar setting, it is shown in [44] that this is the case with a general LTI model for the individual dynamics and local information restricted to a fixed neighborhood of the individual.

A state space formulation of the problem was studied in [45]. A more general notion of information pattern is introduced with each vehicle receiving information from its neighbors on a D-dimensional lattice. By defining the stability margin in terms of the least damped eigenvalue of the appropriate closed loop matrix, it is shown that the stability margin decays to zero as $O(1/n^{2/D})$, where n is the number of vehicles in the platoon. For the case where only a few vehicles (independent of size of platoon) are provided desired trajectory information, this translates to the nearest neighbor type information pattern with D being 1 yielding a decay of $O(1/n^2)$. This result assumed what has come to be known as *symmetric control*,

where an agent weighs the information from all its neighbors on the information graph equally. In a more recent result [46], the authors argue that by introducing asymmetry in weighing the information between the predecessor and the follower, the stability margin can be bounded away from zero uniformly in n.

In [47], the authors analyze the infinite dimensional limiting case of the problem making use of tools from infinite dimensional systems theory. They study the 'rigidity of the formation' or coherence of the whole platoon by defining it in terms of the variance of spacing errors when each vehicle is subjected to additive white Gaussian disturbances. They conclude that in the one dimensional setting nearest neighbor type information patterns inevitably lead to loss of coherence.

Thus, the general trend of the results is a conclusion about a certain inadequacy in control performance when the information available to the individual is constrained to be of the nearest neighbor type. While some ([44], [45]) have mentioned the possibility of using more general information patterns, it does not seem that much progress has been made in this direction. With technological advances, precise and dynamic measurements of positions of vehicles and their communication over inter-vehicle communication networks (see [61]) is practically possible. This gives the control engineer the flexibility to choose from a wider class of information patterns. However, such a choice must be made being mindful of the demand imposed on the communication network.

This chapter is organized as follows. In Section 4.1 we develop a state space formulation of the vehicle platooning problem and describe the performance index w.r.t. which the control performance will be measured. The main result is presented

in Section 4.2 which brings forth the importance of the information pattern in the performance of the distributed control system. Section 4.3 describes the specific class of information patterns that are especially suited for the specific problem studied and in Section 4.4 we present results of numerical simulations. Section 4.5 discusses directions for future work.

Notation

Given a graph $G = (V, E)$, $\mathcal{N}(i)$ denotes the adjacency list of vertex i, $deg(i) = |\mathcal{N}(i)|$ is the degree of the i^{th} vertex, $A_G \in \mathbb{R}^{|V|\times|V|}$ denotes its (normalized) adjacency matrix given by

$$A_G(i,j) = \begin{cases} \frac{1}{deg(i)} & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

and the (normalized) Laplacian $L_G = I - A_G$. A *family of graphs* is an infinite sequence of graphs $\{G_n\}_{n\in\mathbb{N}}$, $G_n = (V_n, E_n)$, with increasing number of vertices such that $n \to \infty \Rightarrow |V_n| \to \infty$.

4.1 Vehicle Platooning—Problem Formulation

Our goal is to understand the effect of the underlying information pattern on the performance of the symmetric distributed control algorithm for the 1-D vehicle platoon. To this end, we assume identical vehicle dynamics and identical controller parameters for every vehicle. Next, we quantify what we mean by control performance by defining an appropriate stability margin. This section closely follows the

set-up in [45].

4.1.1 State Space Formulation

We consider n vehicles indexed by $i \in \{1, \ldots, n\}$, each governed by double-integrator dynamics $\ddot{x}_i = u_i$, where x_i and u_i are real valued functions of time (argument suppressed) denoting the position and control input of the i^{th} vehicle. The first vehicle in the formation, $i = 1$, is given the desired/reference trajectory information $x_{1,d}$. Reference inter-vehicle distances $\Delta_{i,j}$ between vehicle i and its nearest neighbors $j \in \{i-1, i+1\}$ are specified for $i = 2, \ldots, n-1$. Since the formation is one dimensional, such reference spacing between nearest neighbors in effect specifies inter vehicle distance between any two vehicles i and j, denoted by $\Delta_{i,j}$. Further, specifying $x_{1,d}$ specifies the reference trajectory $x_{i,d} = x_{1,d} + \Delta_{1,j}$ for all vehicles $i = 2, \ldots, n$. Note that vehicle 1 alone is given the reference trajectory input $x_{1,d}$. For example, in the case of an IVHS, the supervisory traffic control system can command the reference trajectory to the platoon leader.

Each vehicle is assumed to have sensing capabilities and can measure relative distances and velocities to its nearest neighbors i.e. can measure $(x_i - x_j)$ and $(\dot{x}_i - \dot{x}_j)$ for $j \in \{i-1, i+1\}$ (the first and the last vehicle can measure relative distance and velocity only from vehicles 2 and $n-1$ respectively). Apart from sensing, each vehicle is also capable of inter-vehicle communication and can send and receive relative position and velocity information instantaneously to and from other vehicles. We construct an abstraction of all such inter-vehicle sensing and

communication and represent it by an undirected graph. Each vehicle is represented by a vertex and an edge is drawn between two vertices if the corresponding vehicles have access to their relative spacing information. We call this graph the *information graph* and denote the adjacency list of vertex i by $\mathcal{N}(i)$ (i.e. the set of neighbors of node i in the information graph).

The control objective is to enable tracking of $x_{1,d}$ by vehicle 1 while the others follow maintaining the prescribed $\Delta_{i,j}$ spacings. In particular, we want to achieve this by means of a local feedback law for every vehicle based on the local information available to it. Consider the following feedback law

$$\begin{aligned} u_i &= \frac{1}{deg(i)} \sum_{j \in \mathcal{N}(i)} [-k(x_i - x_j - \Delta_{i,j}) - b(\dot{x}_i - \dot{x}_j)] \\ &+ \delta(1,i)[-k(x_1 - x_{1,d}) - b(\dot{x}_1 - \dot{x}_{1,d})]. \end{aligned} \tag{4.1}$$

Noting that the relative measurements can be written as $x_i - x_j - \Delta_{i,j} = (x_i - x_{i,d}) - (x_j - x_{j,d})$, we define $z = [x_1^T - x_{1,d}^T, \dot{x}_1^T - \dot{x}_{1,d}^T, \ldots, x_n^T - x_{n,d}^T, \dot{x}_n^T - \dot{x}_{n,d}^T]^T$. Substituting u_i from (4.1) to the double-integrator $\ddot{x}_i = u_i$ and assuming $\ddot{x}_{1,d} \equiv 0$, we obtain the following closed loop system

$$\begin{aligned} \dot{z} &= (I_n \otimes A_1 + (L + D_{ext}) \otimes A_2) z \\ \text{where } A_1 &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 \\ -k & -b \end{bmatrix}, \end{aligned} \tag{4.2}$$

where $L \in \mathbb{R}^{n \times n}$ is the Laplacian of the information graph, I_n is the $n \times n$ identity matrix, and D_{ext} is the diagonal matrix with $D_{ext}(i,i) = \delta(1,i)$. We denote the closed loop matrix by $A_{cl} = I_n \otimes A_1 + (L + D_{ext}) \otimes A_2$.

4.1.2 Performance Metric: Stability Margin

We consider the real part of the least damped eigenvalue of A_{cl} as a measure of stability of (4.2). The following was proved in [45].

Theorem 6. *The spectrum of A_{cl} is*

$$\begin{aligned}\sigma(A_{cl}) &= \bigcup_{\gamma\in\sigma(L+D_{ext})} \{\sigma(A_1+\gamma A_2)\} \\ &= \bigcup_{\gamma\in\sigma(L+D_{ext})} \left\{-\frac{\gamma b}{2}\left(1\pm\sqrt{1-\frac{4k}{\gamma b^2}}\right)\right\}. \end{aligned} \tag{4.3}$$

The proof relies on Schur's triangulation theorem and proceeds along the lines of using the unitary matrix U that puts $U^{-1}(L+D_{ext})U$ into an upper triangular matrix, in the similarity transformation $(U^{-1}\otimes I_2)A(U\otimes I_2)$ which yields an upper block diagonal matrix with $(A_1+\gamma A_2)$ on the block diagonal for every $\gamma\in\sigma(L+D_{ext})$.

As a consequence of the Perron Frobenius theorem (see [58]), it is known that the Laplacian of a connected graph has a simple eigenvalue at zero and the corresponding eigenvector is $\mathbf{1}$. Further, all other eigenvalues of the Laplacian of a connected undirected graph are strictly positive and hence $L\geq 0$.

Lemma 4.1.1. *For a connected undirected graph with Laplacian L, $L+D_{ext}>0$ and hence $0\notin\sigma(L+D_{ext})$.*

Proof. $L+D_{ext}=L+e_1e_1^T$ is positive semidefinite. For any $x\in\mathbb{R}^n$,

$$x^T[L+e_1e_1^T]x=0\Leftrightarrow Lx=0 \text{ and } x(1)=0.$$

Since the graph is connected, $Lx = 0 \Leftrightarrow x \in col(\mathbf{1})$, where $col(\mathbf{1})$ denotes column span of $\mathbf{1}$. Now $x^T Lx = 0 \Leftrightarrow Lx = 0 \Leftrightarrow x \in col(\mathbf{1})$ which contradicts $x(1) = 0$ unless $x = 0$. We have $x^T(L + D_{ext})x = 0 \Rightarrow x = 0$ implying $L + D_{ext} > 0$ and $0 \notin \sigma(L + D_{ext})$. □

From Theorem 6, the eigenvalues of A_{cl} are the roots of the polynomial $\prod_{\gamma \in \sigma(L+D_{ext})} (s^2 + b\gamma s + k\gamma) = 0$. Since $k, b > 0$ and from Lemma 4.1.1 all the elements in $\sigma(L+D_{ext})$ are positive, the real parts of all eigenvalues of A_{cl} are strictly negative (i.e. A_{cl} is Hurwitz) and can be explicitly written as (4.3). Define

$$\gamma_{min} = \min \sigma(L + D_{ext}). \tag{4.4}$$

We will be concerned about the asymptotic behavior of γ_{min} within a family of information graphs. Observe that as $n \to \infty$, the only possibility for an eigenvalue of A_{cl} to converge to zero is if $\gamma_{min} \to 0$ (since $\gamma_{min} = 0 \Leftrightarrow 0 \in \sigma(L + D_{ext}) \Leftrightarrow 0 \in \sigma(A_{cl})$). If $\gamma_{min} \to 0$, we can pick a sufficiently large n_0 such that $1 - \frac{4k}{\gamma_{min} b^2} < 0$ for all $n > n_0$ and the real part of the corresponding eigenvalue of A_{cl} is given by $\gamma_{min} b/2$. And if $\gamma_{min} \to 0$, the real parts of the least damped eigenvalues of A_{cl} approach zero in the limit. With this interpretation, we call γ_{min} the *stability margin* of (4.2).

4.2 Effects of Information Exchange Pattern on Stability Margin of Vehicle Platoons

In order to understand the effect of the information pattern on the stability margin we try to relate the spectral properties of the information graph to the latter. In particular, we try to relate γ_{min} and $\lambda_{min} = \min\{\sigma(L) \setminus \{0\}\}$. What we have at hand is essentially the problem of relating an eigenvalue of the sum of two matrices to the eigenvalues of the summands which is known to be difficult. It should be noted that we need something tighter than some known inequalities regarding addition of rank one positive semidefinite matrices to another semi-definite matrix; for instance Theorem 4.3.4, pp. 182, [58] in this case gives $0 \leq \gamma_{min} \leq \lambda_{min}$. We use the special structure of the matrices involved in proving Theorem 7 which provides bounds for the stability margin γ_{min} in terms of λ_{min}. First a technical lemma.

Lemma 4.2.1. *Let $p(s)$ be a monic polynomial of degree $n > 1$ with real roots $\alpha_1 \geq \alpha_2 \cdots \geq \alpha_n > 0$. Then $p(s)$ is strictly convex for n even and strictly concave for n odd in the interval $(-\infty, \alpha_n)$.*

Proof. For any polynomial, the roots of its derivative polynomial lie in the convex

hull of its roots. Thus $\frac{d}{ds}p(s) \neq 0$ and $\frac{d^2}{ds^2}p(s) \neq 0$ for $s \in (-\infty, \alpha_n)$. Now,

$$\begin{aligned} p(s) &= \prod_{i=1}^{n}(s-\alpha_i) \Rightarrow \frac{d}{ds}p(s) = \prod_{i=1}^{n}(s-\alpha_i) \cdot \sum_{j=1}^{n} \frac{-\alpha_j}{s-\alpha_j} \\ \Rightarrow \frac{d^2}{ds^2}p(s) &= \prod_{i=1}^{n}(s-\alpha_i) \cdot \left[\sum_{j=1}^{n} \frac{-\alpha_j}{s-\alpha_j}\right]^2 + \prod_{i=1}^{n}(s-\alpha_i) \cdot \sum_{j=1}^{n} \frac{\alpha_j}{(s-\alpha_j)^2} \\ &= p(s) \cdot \left\{ \left[\sum_{j=1}^{n} \frac{-\alpha_j}{s-\alpha_j}\right]^2 + \sum_{j=1}^{n} \frac{\alpha_j}{(s-\alpha_j)^2} \right\}. \end{aligned}$$

The term in the curly braces in the expression for $\frac{d^2}{ds^2}p(s)$ is strictly positive and well defined in the interval $(-\infty, \alpha_n)$. Noting that $p(s) > 0$ for n even and $p(s) < 0$ for n odd in the interval $(-\infty, \alpha_n)$ leads to the sufficiency conditions on the second derivative for convexity and concavity respectively. □

Theorem 7 (Relation between Control Performance and Information Pattern [48])**.** *Let L be the normalized Laplacian of a connected undirected graph $G = (V, E)$ and D_{ext} be a diagonal matrix with $D_{ext}(i,i) = \delta(1,i)$. Then γ_{min} defined in (4.4), and the second smallest eigenvalue of L, λ_{min}, satisfy*

$$\frac{\lambda_{min}}{4n} < \gamma_{min}, \tag{4.5}$$

where $n = |V|$.

Proof. The idea is to perform some similarity transformations on $L + D_{ext}$ to get it in a form where its characteristic polynomial can be expressed in terms of the eigenvalues of L. Then we use a first order Taylor approximation of the characteristic polynomial at zero to obtain the desired lower bound.

From the spectral theorem, there exists an orthogonal matrix P such that $\Lambda \doteq P^T L P = diag([\lambda_1, \lambda_2, ..., \lambda_{min}, 0])$, where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{n-1} = \lambda_{min} > 0$ are

the eigenvalues of L. Let the set of vectors $\{e_i\}_{i=1}^n$ denote the natural basis of $\mathbb{R}^n$. Let the i^{th} column of P be denoted by p_i and that of P^T by r_i. Since $P^TP = PP^T = I$, we have $p_i^T p_j = r_i^T r_j = \delta(i,j)$. Thus $P^TLPe_n = 0 \Rightarrow Lp_n = 0 \Leftrightarrow p_n = \frac{1}{\sqrt{n}}\mathbf{1}$.

We prove a more general result by allowing D_{ext} to be a rank one matrix with any one of the diagonal elements being one and the rest of the entries being zero. We can write D_{ext} as $e_k e_k^T$ for some k; $k = 1$ corresponds to the D_{ext} in the statement of the theorem. Next, we perform the similarity transformation $P^T(L + D_{ext})P = \Lambda + r_k r_k^T$,

$$P^T(L + D_{ext})P = \Lambda + \begin{bmatrix} | & | & & | \\ r_k(1)r_k & r_k(2)r_k & \cdots & r_k(n)r_k \\ | & | & & | \end{bmatrix}.$$

Since $p_n = \frac{1}{\sqrt{n}}\mathbf{1}$, $r_k(n) = \frac{1}{\sqrt{n}}$ for any k. One can perform elementary column operations $j^{th}\ column \rightarrow j^{th}\ column - \frac{r_k(j)}{r_k(n)}\ n^{th}\ column$ for all $j \in \{1, \cdots, n-1\}$. Such an elementary operation can be expressed as right multiplication by a matrix $\tilde{T}$ given by

$$\tilde{T} = \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & \\ -\frac{r_k(1)}{r_k(n)} & \cdots & -\frac{r_k(n-1)}{r_k(n)} & 1 \end{bmatrix}.$$

Applying these column operations on $P^T(L + D_{ext})P$,

$$P^T(L + D_{ext})P\tilde{T} = \begin{bmatrix} | & & | & | \\ \lambda_1 e_1 & \cdots & \lambda_{min} e_{n-1} & r_k(n)r_k \\ | & & | & | \end{bmatrix}.$$

Let $\tilde{\Lambda}$ be the $(n-1)\times(n-1)$ matrix obtained by removing the last row and last column of Λ and $\tilde{r}_k$ be the $(n-1)$ dimensional column vector obtained by removing the last element of r_k. Then

$$\tilde{T} = \begin{bmatrix} I_{n-1} & 0 \\ -\frac{1}{r_k(n)}\tilde{r}_k^T & 1 \end{bmatrix}, \quad \tilde{T}^{-1} = \begin{bmatrix} I_{n-1} & 0 \\ \frac{1}{r_k(n)}\tilde{r}_k^T & 1 \end{bmatrix}$$

and we can rewrite

$$P^T(L+D_{ext})P\tilde{T} = \begin{bmatrix} \tilde{\Lambda} & r_k(n)\tilde{r}_k \\ 0 & r_k^2(n) \end{bmatrix}.$$

Left multiplying by $\tilde{T}^{-1}$,

$$\tilde{T}^{-1}P^T(L+D_{ext})P\tilde{T} = \begin{bmatrix} \tilde{\Lambda} & r_k(n)\tilde{r}_k \\ \frac{1}{r_k(n)}\tilde{r}_k^T\tilde{\Lambda} & 1 \end{bmatrix} \doteq \hat{L}.$$

Since $\hat{L} = (P\tilde{T})^{-1}(L+D_{ext})(P\tilde{T})$, the spectrum of $\hat{L}$ is the same as the spectrum of $L+D_{ext}$.

We now find the characteristic polynomial of $L+D_{ext}$ by finding that of $\hat{L}$. For some $\hat{x}\in\mathbb{R}^n$, $\hat{x}\neq 0$, $(sI-\hat{L})\hat{x}=0 \Rightarrow$

$$(s-\lambda_j)\hat{x}(j) = r_k(n)r_k(j)\hat{x}(n) \text{ for all } j\in\{1,\cdots,n-1\}$$
$$\text{and } (s-1)\hat{x}(n) - \sum_{j=1}^{n-1}\frac{r_k(j)}{r_k(n)}\lambda_j\hat{x}(j) = 0.$$

Substituting for $\hat{x}(j)$ for all $j\in\{1,\cdots,n-1\}$ in terms of $\hat{x}(n)(\neq 0)$, we obtain the characteristic polynomial for $L+D_{ext}$

$$\chi(s) \;=\; \left[\prod_{j=1}^{n-1}(s-\lambda_j)\right]\left[(s-1) - \sum_{i=1}^{n-1}\frac{r_k^2(j)}{s-\lambda_j}\right].$$

Since $L+D_{ext}$ is symmetric positive definite, the roots of $\chi(s)=0$ are real and positive; the smallest being γ_{min}. From Lemma 4.2.1, for even (odd) $n>1$, $\chi(s)$

is strictly convex (concave) in the interval $(-\infty, \gamma_{min})$. For n even, $\chi(s) > 0$ for $s \in (-\infty, \gamma_{min})$ and the tangent at $s = 0$ is below the curve due to strict convexity implying that in a plot of s vs. $\chi(s)$, the intercept of tangent with the horizontal axis is a strict lower bound for γ_{min}. A similar argument can be made for the case where n is odd. The derivative of $\chi(s)$ at $s = 0$ is

$$\frac{d}{ds}\chi(s)\bigg|_{s=0} = (-1)^{n-1}\left(\prod_{j=1}^{n-1}\lambda_j\right)\left[\frac{1}{n}\sum_{j=1}^{n-1}\frac{1}{\lambda_j} + 1 + \sum_{j=1}^{n-1}\frac{r_k^2(j)}{\lambda_j}\right]$$

and the value of $\chi(s)$ at $s = 0$ is $\chi(0) = \frac{1}{n}(-1)^n\left(\prod_{j=1}^{n-1}\lambda_j\right)$ (we have used $\sum_{j=1}^{n-1} r_k^2(j) + r_k^2(n) = 1$ and $r_k(n) = \frac{1}{\sqrt{n}}$ for simplifications). The equation of the tangent to $\chi(s)$ at $s = 0$ is given by $\chi_T(s) = \chi(0) + s\chi'(0)$. If n is even (odd), the intercept with the vertical axis, $\chi(0)$, is positive (negative) and the slope $\chi'(0)$ is negative (positive) resulting in a positive horizontal axis intercept. Solving for $\chi_T(s^*) = 0$ gives s^*, a lower bound to γ_{min}.

$$\begin{aligned} 0 < s^* &= \frac{\chi(0)}{\chi'(0)} < \gamma_{min} \\ \Rightarrow \frac{1}{\gamma_{min}} &< n\left[\frac{1}{n}\sum_{j=1}^{n-1}\frac{1}{\lambda_j} + 1 + \sum_{j=1}^{n-1}\frac{r_k^2(j)}{\lambda_j}\right] \\ &\leq \frac{n}{\lambda_{min}} + n + \frac{n}{\lambda_{min}}\sum_{j=1}^{n-1} r_k^2(j) \\ &\leq \frac{n}{\lambda_{min}}(2 + \lambda_{min}). \end{aligned} \tag{4.6}$$

The lower bound in (4.5) follows by noting that $\lambda_i \leq 2$ for all i. □

Remark. The lower bound in (4.5) holds even with $m > 1$ (independent of n) 'lead' vehicles each given reference trajectory information externally which translates to D_{ext} having m ones on its diagonal and rest of the elements zero. This

is a direct consequence of splitting D_{ext} as a sum of a rank one matrix $e_k e_k^T$ and a rank $(m-1)$ matrix $D_{ext} - e_k e_k^T$ for some k such that $D_{ext}(k,k) = 1$ and applying Weyl's inequality to the sum $(L + e_k e_k^T) + (D_{ext} - e_k e_k^T)$ where the smallest eigenvalue of $L + e_k e_k^T$ satisfies the bounds in (4.5).

For a nearest neighbor type information graph, it has been proved in [45] that with a control law similar to (4.1), the stability margin γ_{min} decays to zero as $O(1/n^2)$. We would like to improve upon the asymptotic rate by changing the information graph. It is clear from the bounds in (4.5) that the asymptotic behavior of γ_{min} is closely related to the asymptotic behavior of the second smallest eigenvalue of the Laplacian λ_{min}.

In terms of asymptotic behavior, λ_{min} can either be bounded away from zero or can approach zero with some rate. If λ_{min} is bounded away from zero for arbitrarily large n for a family of graphs, we can conclude from (4.5), that $\gamma_{min} > O(1/n)$ with such a family chosen as the information graph. Asymptotically, a $O(1/n)$ decay is strictly slower than the $O(1/n^2)$ resulting from the nearest neighbor type information graphs and hence is better as the stability margin approaches zero slower. A question arises regarding which families of graphs satisfy the property that λ_{min} is bounded away from zero as $n \to \infty$?

4.3 Expander Families as Information Patterns

We now define an expander family of graphs and briefly present some results from graph theory relevant to the analysis of expander families. We then discuss the

use of members of expander families as information graphs in the vehicle platoon problem. For a detailed introduction to expander families, we refer the interested reader to the survey article [62].

Definition 4.3.1. *The* edge expansion *of a d-regular undirected graph* $G = (V, E)$ *is given by*

$$h(G) = \min_{S \subset V, 0 < |S| \leq |V|/2} \frac{|E(S, V \setminus S)|}{d\,|S|}$$

where $E(S, V \setminus S)$ *is the set of edges which are incident on a vertex in* S *and* $V \setminus S$.

Thus edge expansion is the minimum across all nontrivial cuts $(S, V \setminus S)$ of the ratio of the number of edges across a cut and the number of vertices in the smaller set in the cut. Intuitively, a graph with 'large' edge expansion can be interpreted as a 'better connected' graph as one has to remove a 'large' number of edges to disconnect a 'large' enough component.

Definition 4.3.2. *A family of d-regular graphs* $\{G_n\}_{n \in \mathbb{N}}$ *is an* expander family *if for some* $\epsilon > 0$, $h(G_n) > \epsilon$ *for all* n.

Let us denote the adjacency matrix of G_n by A_n, $|V_n| = n$ and the eigenvalues of A_n by $1 = \alpha_1 \geq \alpha_2 \cdots \geq \alpha_n$. The following inequality is attributed to Cheeger (see pp. 454, Theorem 2.4 [62])

$$\frac{1 - \alpha_2}{2} \leq h(G) \leq \sqrt{2 \cdot (1 - \alpha_2)}. \tag{4.7}$$

The eigenvalues of the corresponding Laplacian L_n and A_n get related as $\lambda_{n-i+1} = 1 - \alpha_i$ with $\alpha_1 = 1$ corresponding to the zero eigenvalue of L_n and $\lambda_{min} = 1 - \alpha_2$.

Substituting in (4.7) we have

$$\frac{\lambda_{min}}{2} \leq h(G) \leq \sqrt{2 \cdot \lambda_{min}}.$$

This leads to an equivalent characterization of expander families as a family of graphs such that for some $\epsilon > 0$, $\lambda_{min} > \epsilon$ for all n. Note that in the discussion at the end of the previous section, this was the sought after property for outperforming nearest neighbor type information patterns.

Explicit constructions of expander families based on results from different areas of mathematics have been known since 1970s. It is known that a random d-regular graph in the limit is a good expander with high probability. A technique called *zig-zag product*, which is related to the replacement product of two graphs, has also been discovered which can be applied to construct expander families (see [62] and [63]). Finding the appropriate expander family to serve as the information graph in the platoon problem will require further work and will involve several implementation related considerations such as physical or hop distance in the communication network between two neighbors on the information graphs etc. We do not attempt to answer these questions in detail here. Instead, we present an example applicable when the number of vehicles in the platoon is a prime.

Fact 4.3.1. *(pp. 453, §2.2, [62]) Let $\{p_i\}_{i\in\mathbb{N}}$ be an infinite sequence of increasing primes. The 3-regular family of graphs $\{S_{p_i}\}_{i\in\mathbb{N}}$, $S_{p_i} = (V_i, E_i)$, with $V_i = \mathbb{Z}_{p_i}$ and for every $a \in V_i$, $(a, a+1), (a, a-1)$ and $(a, a^{-1}) \in E$ is an expander family (all operations are mod p_i).*

The motivation for picking this family of expanders is that the edges $(a, a-1)$

and $(a, a+1)$ have a physical meaning as the sensed distance between a vehicle and its immediate successor and predecessor. In Figure 4.1 we plot the sparsity pattern of a member of this family using the MATLAB function $spy(\cdot)$ where the off diagonal elements are seen to be non-zero.

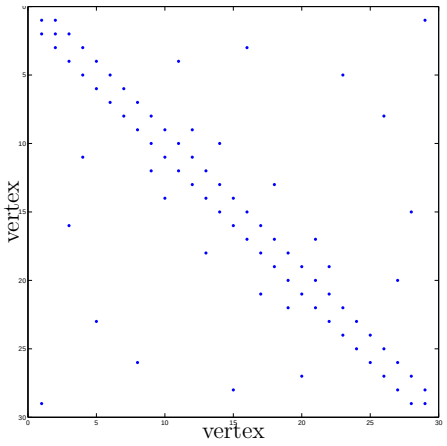


Figure 4.1: Sparsity pattern of S_{29}

Several known explicit constructions of expander families yield multi-graphs i.e. graphs with multiple self loops and multiple edges. It is worth mentioning at this point that all the analysis of section 3 holds for information graphs that are possibly multi-graphs. One has to account for these by modifying the normalized adjacency matrix entries as $A(i,j) = \frac{|j \in \mathcal{N}(i)|}{deg(i)}$, where $|j \in \mathcal{N}(i)|$ is the multiplicity of j in the adjacency list of i.

We would like to make another feature of expander families explicit. While expander families have good connectivity properties, due to the restriction of d-regularity in the definition the number of edges are of order $O(n)$. In contrast, a complete graph has very good connectivity but at the 'cost' of having a larger

number of edges of the order $O(n^2)$. From an implementation point of view, not only does using expanders as information graphs improve control performance in terms of stability margins, it also limits the amount of inter vehicle communication required. For instance if a multi-hop wireless network were used for such communication, the number of messages at any given point of time would be at most $O(n)$ giving flexibility in designing such protocols.

This brings us to another issue in the choice of the expander family. After making a choice for the information graph based on such spectral considerations, a question arises as to which vertex of the graph be assigned to which vehicle. Since any assignment does not change the spectral properties of the information graph one can try to do this assignment so as to reduce the inter-vehicle communication. If the communication is over a multi-hop network it would be prudent to reduce the hop count of the longest edge on the graph. This means finding the permutation of the vertex assignment which results in the least bandwidth adjacency matrix. This is the graph bandwidth minimization problem which is known to be NP hard. Several approximation algorithms are known and we show the results of the Cuthill-McKee algorithm implemented by the MATLAB function $symrcm(\cdot)$ used to reduce the bandwidth of S_{541} in Figure 4.2.

4.4 Simulations

We begin with some numerical illustrations. Figure 4.3 shows an experimental verification of the bound in (4.5) with the plot on log scale along the vertical axis

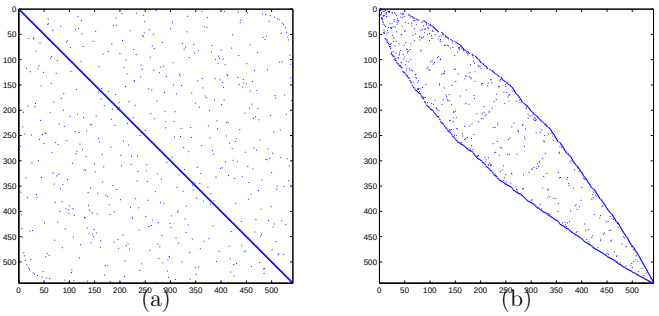


Figure 4.2: Sparsity pattern before, and after, bandwidth reduction

Sparsity plots of (a) S_{541} and (b) bandwidth reduced S_{541}

of γ_{min} being strictly above λ_{min}/n against the first 150 primes as the number of vehicles along the horizontal axis with the expander family S_n as the information pattern. The nontrivial gap between the curves suggests that the lower-bound is not tight and that our conclusion of an at most $O(1/n)$ decay rate for the stability margin may be pessimistic i.e. the decay rate may be slower. In Figure 4.4 we compare the stability margin with expander as the information pattern against the nearest neighbor type information pattern along the same axis. Notice that the decay rate in the case of the former is slower than the latter, providing validation to our argument of using an expander as the information pattern.

4.5 Discussion

Returning to the main theme of this chapter, the main goal of Sections 4.1, 4.2 and 4.3 is to bring forth the possibility of employing more general information patterns in the vehicle platoon problem and present an analysis where different kind

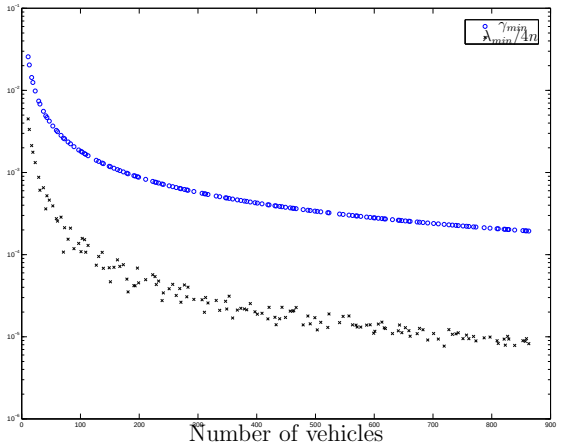


Figure 4.3: Plot of γ_{min} and $\lambda_{min}/4n$ on log scale vs. number of vehicles

of patterns can be compared. We have also tried to address the issue of reducing the load on the communication network while improving control performance. It is clear that optimizing a cost for communication over the set of all possible information patterns is NP hard; instead we argue that expander families, due to their sparsity, lower the demand on the communication network while their spectral properties help improve control performance. The practical gains from such an approach is once we identify "good" information patterns by using simplistic controllers, one can then fix the information pattern and consider improving the performance by tuning the controller. Thus, while not optimal, the approach leads to a useful design paradigm.

The problem formulation considered, consisting of individual double integrators, can be viewed as the consensus problem with double integrator dynamics. Consensus dynamics with single integrators have been widely studied and the relationship between information pattern and their convergence properties is well understood in terms of the spectral gap of the Laplacian [64]. In this sense, our result

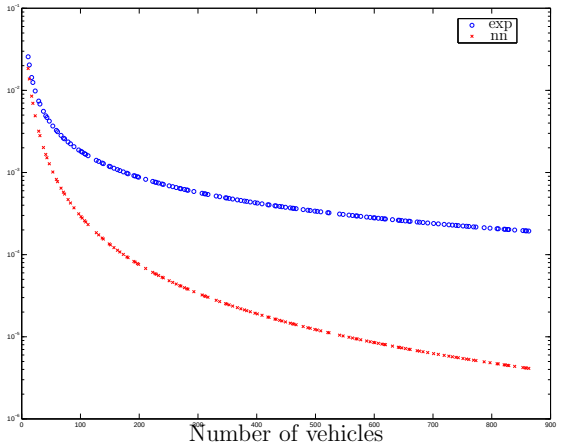


Figure 4.4: Plot of stability margin on log scale with expander (exp) and nearest neighbor (nn) information pattern vs. number of vehicles

is an extension of such work to the case of double integrator dynamics. One possible extension can be understanding the effects of the information pattern when the agent dynamics are more general. Similar questions were studied in [65].

Finally, it should point out that while we have ignored the issue of implementing non-nearest neighbor type information patterns, such as the $\{S_{p_i}\}$ family, current vehicle-to-vehicle wireless communication technologies make the implementation of longer inter-vehicle communication implementation possible.

Chapter 5: Conclusion

An important model-free distributed optimization problem (see Section 1.2. (WO)) is formulated in this thesis. Besides the several applications that can be modeled in this framework, an underlying motivation for this work is to show that in engineered multi-agent systems (MAS), inter-agent communications and collaboration can be leveraged to achieve behaviors beyond Nash equilibria. The desired emergent property of the MAS is indeed application specific, and this thesis adds the option of a welfare optimal outcome to the literature.

Two algorithms, the eITEL algorithm and the CES algorithm, are developed to solve the problem. The eITEL algorithm applies in a setting where agent actions are drawn from a finite action-set, while CES applies when the agents actions are real numbers. The local convergence of CES is guaranteed under existence of a local minimizer for the welfare function and a connected communication graph between the agents, and the algorithm essentially performs a gradient estimation followed by descent. This gives CES an edge over eITEL in terms of speed of convergence as the latter does not exploit any gradient-type information in its search for the extrema. The convergence result of eITEL, on the other hand, is a more refined one than that of CES. Instead of a connected communication graph, only some carefully placed

edges in the communication graph are required by eITEL which exploits implicit communications within the MAS to achieve the collaborative objective.

A possible direction for future work consists of developing model-free counter parts of the existing distributed model-based optimization procedures by incorporation of appropriate-order derivative estimators using extremum seeking type ideas. Such extensions have been performed for extremum seeking control (see, for instance, [66]). Another direction is the development of a continuous space analog for the eITEL algorithm that exploits gradient estimates for fast convergence while requiring minimal explicit inter-agent communications. A noteworthy work in this context is [67] where inspiration is drawn from bacterial chemotaxis for designing decentralized algorithms that converges to prescribed system-wide behavior. A hybrid stochastic differential equation reminiscent of the mood variable based dynamics of eITEL is used for the agent dynamics in that work.

On the question of determining appropriate information exchange networks for distributed control of MAS, we provide a case study of the vehicle platooning problem. By deriving a relationship between the control performance (of this problem) for an arbitrary information exchange pattern we are able to point out the necessary qualitative features for the right information pattern. In the case of the platooning problem this qualitative feature turns out to be: 'the first positive eigenvalue of the communication graph Laplacian matrix being bounded away from zero leads to desirable control performance'. At the intersection of this property and that of a 'small' number of edges in the graph lie expander families of graphs. Thus, we argue expanders are an example of a right trade-off between good control performance at

tolerable communication overhead for the vehicle platooning problem.

Another interesting case study for answering such a question is in collaborative robotics, scenarios where a group of autonomous robots perform a task. These could be mobile UAVs and ground robots performing surveillance or tasked with lifting an object, etc. Consider a group of robots with individual dynamics given by $\dot{x}_i = f_i(x_i, u_i)$. The task is often planned ahead of time by posing it as an optimal control problem with appropriate penalty between the state and control variables of the robots: $\sum_{I \subset \{1,\ldots,n\}} \int_0^T J_I(x_I, u_I)dt$. This often leads to an open-loop optimal state-control trajectory (x_i^*, u_i^*). When implementing, however, an appropriate feedback mechanism must be implemented to ensure the performance is met. Let $u_i^*(t)+\delta u_i(t)$ be the control applied to the i^{th} robot that results in the state trajectory $x_i^* + \delta x_i$. Substituting these in the state equations and performance functional, for small δu_i and δx_i, we have state equations $\delta \dot{x}_i = \frac{\partial f_i}{\partial x_i}|_{x_i^*(t)} \delta x_i(t) + \frac{\partial f_i}{\partial u_i}|_{u_i^*(t)} \delta u_i(t)$ and performance functional $\sum_{I \subset \{1,\ldots,n\}} \int_0^T [\delta x_I^T, \delta u_I^T] \frac{\partial^2 J_I}{\partial^2 (x_I, u_I)}|_{(x_I^*, u_I^*)} [\delta x_I^T, \delta u_I^T]^T dt$. Notice the state equations are time-varying linear systems and the objective is a time varying quadratic penalty. This motivates the investigation of the simpler time-invariant decentralized LQR problem: robot i's dynamics are given by $\dot{x}_i = A_i x_i + B_i u_i$, its local measurement $y_i = c_i x_i$ and the task is for the agents to jointly minimize the cost $\int_0^T [x'Qx + u'Ru]\ dt$. The matrices Q and R capture the coupling between the robots and correspond to the notion of which robot "works" with which other. This lends to an interpretation of these matrices as manifestations of a *collaboration graph* between the robots. Let the robots be allowed to exchange state-information over a static communication graph with a sparsity structure given by matrix $S \in \{0, 1\}^{n \times n}$.

Since such communication is energy intensive, a natural question that emerges is which communication graph is appropriate for a given collaboration graph? One approach to tackle the problem is to work directly with the associated differential (or algebraic) Riccati equation and understanding its interplay with the sparsity constraints. Alternately, as mentioned earlier, one can fix simple suboptimal feedback rules for the agents and analyze the relationship between the information pattern S and the required task Q. Such relationships, we hope, will be useful guidelines for co-design of the information pattern and the distributed control system.

Chapter A: Ergodicity of Nonhomogeneous Perturbed Markov Chains

A.1 Proof of Theorem 3

We review results regarding ergodicity of nonhomogeneous Markov chains before proceeding to the proof of Theorem 3.

Definition A.1.1 (Ergodic Coefficient)**.** *Given a row stochastic matrix* $Q \in \mathbb{R}^{|S|\times|S|}$*, its ergodic coefficient is given by*

$$\delta(Q) = 1 - \min_{x,y\in S} \sum_{z\in S} \min\{Q_{x,z}, Q_{y,z}\}.$$

The following result due to Doeblin provides a characterization for *WE* based on the ergodic coefficient.

Theorem 8 (Weak Ergodicity, see [68], Theorem 8.2)**.** *The chain is weakly ergodic if and only if there exists a strictly increasing sequence of positive integers* $\{t_n\}_{n\in\mathbb{N}}$ *such that*

$$\sum_{n\in\mathbb{N}} (1 - \delta(Q^{(t_n,t_{n+1})})) = \infty. \tag{A.1}$$

The next Theorem provides a sufficiency condition for *SE*.

Theorem 9 (Strong Ergodicity, see [68], Theorem 8.3)**.** *Suppose the chain is weakly*

ergodic, there exists π_t *such that* $\pi_t Q(t) = \pi_t$ *for all* t*, and*

$$\sum_{t\in\mathbb{N}} \|\pi_{t+1} - \pi_t\| < \infty, \tag{A.2}$$

then the chain is strongly ergodic. Furthermore, the limiting distribution π *as in the definition of SE is the same as the limit of the sequence* $\{\pi_t\}_{t\in\mathbb{N}}$*.*

We will need the following technical Lemma.

Lemma A.1.1. *Let* $\sum_{n\in\mathbb{N}} a(n) = \infty$ *and* $a(n) \geq a(n+1)\ \forall\ n$*. Then for any* $n', l \in \mathbb{N}$*,* $\sum_{n\in\mathbb{N}} a(n' + l + n) = \infty$*.*

Proof. The case for $l = 1$ is trivially true. If $l > 1$, $\forall\ n$,

$$\begin{aligned} &a(n' + ln) \geq a(n' + ln + m),\ \forall\ m = 1, ..., l-1 \\ &\Rightarrow l \cdot a(n' + ln) \geq \sum_{m=0}^{l-1} a(n' + ln + m). \end{aligned}$$

Thus

$$l \cdot \sum_{n\in\mathbb{N}} a(n' + ln) \geq \sum_{n\in\mathbb{N}} \sum_{m=0}^{l-1} a(n' + ln + m) = \sum_{n\in\mathbb{N}} a(n' + n) = \infty.$$

□

Proof of Theorem 3. Weak Ergodicity: Let E^* be a recurrent class such that $CR(E^*) = \kappa$. Since E^* is aperiodic according to $P(0)$, there exists an $l_1 \in \mathbb{N}$ such that for all $m \geq l_1$ and $x, y \in E^*$, $P^m_{x,y}(0) > 0$ (see [68], Theorem 4.3, pp. 75). Since any path under $P(0)$ has zero resistance, once the chain enters a state in E^*, it can remain there with zero resistance via a path of length greater than l_1.

Let $e^* \in E^*$ be such that $\exists\ x' \in S \backslash E^*$ such that $\rho(x', e^*) = \kappa$ i.e. the transition $x' \to e^*$ has the most resistance among all $x \to e^*$, $x \in S$. For all $x \in S$, consider the

shortest paths $h(x \to e^*)$ such that $r(h(x \to e^*)) = \rho(x, e^*)$ and denote the length of such paths by $l(x, e^*)$. Let $l_2 = \max_{x \in S} l(x, e^*)$. So by waiting for l_2 transitions, there is a path to E^* from all states $x \in S$ with resistance $\rho(x, e^*)$. Thus by allowing more than $l = l_1 + l_2$ transitions, we have for any $x \in S$ and a sufficiently small ϵ^*,

$$P^m_{x,e^*}(\epsilon) > \underline{\alpha}^m \epsilon^\kappa, \quad \forall\, \epsilon < \epsilon^*,\ m \geq l.$$

From (2.4), since $\epsilon_t \to 0$, for sufficiently large t,

$$\underline{\alpha}\ \epsilon_t^{r(x,y)} < \mathbf{P}_{x,y}(t) < \overline{\alpha}\ \epsilon_t^{r(x,y)}.$$

Consequently, by choosing a subsequence such that $t_{n+1} - t_n = l$, for sufficiently large n,

$$\mathbf{P}^{(t_n, t_{n+1})}_{x,e^*} > \underline{\alpha}^l \epsilon^\kappa_{t_{n+1}},\ \forall\, x \in S.$$

Then, for sufficiently large n, we can bound

$$\sum_{z \in S} \min\{\mathbf{P}^{(t_n,t_{n+1})}_{x,z}, \mathbf{P}^{(t_n,t_{n+1})}_{y,z}\} \geq \min\{\mathbf{P}^{(t_n,t_{n+1})}_{x,e^*}, \mathbf{P}^{(t_n,t_{n+1})}_{y,e^*}\} > \underline{\alpha}^l \epsilon^\kappa_{t_{n+1}},\ \forall\, x, y \in S.$$

Taking minimum over x, y, for sufficiently large n,

$$\min_{x,y \in S} \sum_{z \in S} \min\{\mathbf{P}^{(t_n,t_{n+1})}_{x,z}, \mathbf{P}^{(t_n,t_{n+1})}_{y,z}\} > \underline{\alpha}^l \epsilon^\kappa_{t_{n+1}}. \tag{A.3}$$

Since $\{t_n\}_{n \in \mathbb{N}}$ is an equally spaced subsequence, from Lemma A.1.1 and the hypothesis of the theorem, $\sum_{n \in \mathbb{N}} \epsilon^\kappa_{t_{n+1}} = \infty$. In view of this and (A.3), *WE* follows by noting that (A.1) is verified with $Q = \mathbf{P}$.

Strong Ergodicity: Recall the homogeneous perturbed Markov chain $P(\epsilon)$. Consider a graph $\mathcal{G} = (S, \mathcal{E})$ with the state space S as the vertex set and a directed

edge $(x, y) \in \mathcal{E}$ if and only if $P_{x,y}(\epsilon) > 0$ for some ϵ. For any vertex $z \in S$, a *z-tree* is a subset of $\mathcal{E}$ that forms a spanning tree in $\mathcal{G}$ such that for every vertex $x \neq z$, there exists a unique directed path from x to z. Let $\mathcal{T}_z$ be the set of all z-trees in $\mathcal{G}$. Then it is known (see [34]) that the stationary distribution $\mu(\epsilon)$ is given by

$$\begin{aligned} \mu_z(\epsilon) &= \frac{q_z(\epsilon)}{\sum_{x \in S} q_x(\epsilon)} \\ \text{where } q_z(\epsilon) &= \sum_{T \in \mathcal{T}_z} \prod_{(x,y) \in T} P_{x,y}(\epsilon). \end{aligned} \tag{A.4}$$

Under Assumption 2, both the numerator and denominator of the R.H.S of (A.4) belong to $\mathfrak{L}$ for a sufficiently large L. Denoting the derivative w.r.t. ϵ by primes and suppressing the argument, $\mu_z' = (1/(\sum_{x \in S} q_x)^2) \cdot (q_z' \sum_{x \in S} q_x - q_z \sum_{x \in S} q_x')$. Thus, after multiplying and dividing with an appropriate power of ϵ, the numerator of μ_z' also belongs to $\mathfrak{L}$ for a sufficiently large L. For a sufficiently small $\epsilon_z > 0$, μ_z' will be dominated by the term with the least exponent of ϵ for all $\epsilon < \epsilon_z$. Thus, the sign of μ_z' will be either non-positive or positive for all $\epsilon < \epsilon_z$. Let $\epsilon^* = \min_{z \in S} \epsilon_z$, $S^- \subset S$ be such that $z \in S^- \Leftrightarrow \mu_z' \leq 0 \ \forall \ \epsilon < \epsilon^*$ and $S^+ = S \setminus S^-$. Let t^* be such that $\epsilon_t < \epsilon^*, \ \forall \ t > t^*$. Then,

$$\begin{aligned} &\sum_{t=1}^{\infty} \|\mu(\epsilon_t) - \mu(\epsilon_{t+1})\|_1 = \\ &\quad \sum_{t=1}^{t^*} \|\mu(\epsilon_t) - \mu(\epsilon_{t+1})\|_1 + \sum_{t=t^*+1}^{\infty} \left[\sum_{z \in S^-} (\mu_z(\epsilon_t) - \mu_z(\epsilon_{t+1})) + \sum_{z \in S^+} (\mu_z(\epsilon_{t+1}) - \mu_z(\epsilon_t)) \right] \\ &= M + \sum_{z \in S^-} \sum_{t=t^*+1}^{\infty} (\mu_z(\epsilon_t) - \mu_z(\epsilon_{t+1})) + \sum_{z \in S^+} \sum_{t=t^*+1}^{\infty} (\mu_z(\epsilon_{t+1}) - \mu_z(\epsilon_t)) \\ &< \infty \end{aligned}$$

since $M = \sum_{t=1}^{t^*} \|\mu(\epsilon_t) - \mu(\epsilon_{t+1})\|_1$ is a finite sum of finite terms and successive terms

cancel within both infinite sums. Since (A.2) is satisfied with $\pi(t) = \mu(\epsilon_t)$, as shown above, and the chain is *WE*, *SE* follows from Theorem 9. The limiting distribution, in view of Theorem 2, is $\mu(0)$. □

A.2 Proof of Lemma 2.3.1

Proof of Lemma 2.3.1. Let us consider the transition probability from state $y \in S$ to $z \in S$. For $\epsilon > 0$, irrespective of the values of respective $\tilde{m}_i$, the transition probabilities (2.1) and (2.2) let the agents pick a joint action $a' \in \mathcal{A}$ such that $a'_i \neq a^y_i$ for any i with positive probability. Then, again irrespective of the values of $\tilde{m}_i$, by step 4.2 or 4.3, the state can transition from y to $[a', \mathbf{0}]$ with positive probability. Next, starting from state $[a', \mathbf{0}]$, by Assumption 1 and transition probability (2.2), agent i can pick the action a^z_i with positive probability in a finite number of steps and can keep playing a^z_i with positive probability for any arbitrary finite duration thereafter while maintaining $m_i = 0$ all the while. Thus there is a positive probability for all agents to pick actions that correspond to a', i.e. transition from state $[a', \mathbf{0}]$ to $[a^z, \mathbf{0}]$. Finally, in the very next time instant, agent i can repeat its action with positive probability and update its mood variable to m^z_i with positive probability (2.3). Hence the transition y to z occurs with positive probability.

Aperiodicity follows by noting that the $P_{x,x}(\epsilon) > 0$ for any $x \in S$: the same action can be picked by the agents in consecutive time steps with positive probability and (2.3) permits picking the same mood variable again with positive probability.

□

Chapter B: Convergence Analysis of Collaborative Extremum Seeking

B.1 Proof of Lemma 3.2.2

Proof of Lemma 3.2.2. It is known that M is Hurwitz if and only if the origin is the asymptotically stable equilibrium point for the LTI system $\dot{x} = Mx$. Let $Q_{sym} = Q+Q^T$, according to hypothesis $Q_{sym} > 0$. Consider a candidate Lyapunov function $V(x) = x^T P_\delta x$ where

$$P_\delta = \begin{bmatrix} -Q_{sym} & R \\ R^T & 0 \end{bmatrix} + \delta I.$$

Then,

$$\dot{V} = x^T(M^T P_\delta + P_\delta M)x = x^T \begin{bmatrix} S - \delta Q_{sym} & QR \\ R^T Q^T & -2R^T R \end{bmatrix} x$$

$$= x^T \left\{ \begin{bmatrix} -I & 0 \\ 0 & -R^T R \end{bmatrix} + \begin{bmatrix} -QQ^T & QR \\ (QR)^T & -R^T R \end{bmatrix} + \begin{bmatrix} S + QQ^T + I - \delta Q_{sym} & 0 \\ 0 & 0 \end{bmatrix} \right\} x, \tag{B.1}$$

where $S = S^T = Q^T Q_{sym} + Q_{sym} Q + 2RR^T$. Now, pick $\delta > \max\{\lambda_{max}(Q_{sym}), \lambda_{max}(S + QQ^T + I)/\lambda_{min}(Q_{sym})\}$, so that $P_\delta > 0$ and $S + QQ^T + I - \delta Q_{sym} < 0$. Notice that the first matrix in the curly brackets in (B.1) is negative definite as

$rank(R) = m \Rightarrow R^TR > 0$, and the third matrix in the curly brackets is negative semi-definite due to the choice of δ made above. Thus, if the second matrix in the curly brackets in (B.1) is negative semi-definite, then we have $V(x) > 0$ and $\dot{V} < 0,\ \forall\ x \neq 0$, which concludes the proof by Lyapunov's Theorem. Observe that this is indeed the case: Let x_1 and x_2 be vectors of appropriate dimensions, then

$$\begin{aligned}
&[x_1^T x_2^T] \begin{bmatrix} -QQ^T & QR \\ (QR)^T & -R^TR \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\
&= -\|Q^Tx_1\|_2^2 + 2(Q^Tx_1)\cdot(Rx_2) - \|Rx_2\|_2^2 \\
&\leq -\|Q^Tx_1\|_2^2 + 2\|Q^Tx_1\|\|(Rx_2)\| - \|Rx_2\|_2^2 \leq 0;
\end{aligned}$$

where the first inequality follows from the Cauchy-Schwarz inequality. □

B.2 Proof of Theorem 5

First note that, due to Lemma 3.2.1, the dynamics of $\hat{u}$ and x variables are unaffected by replacing the $2n$ equations for $\dot{x}$ and $\dot{\lambda}$ by the $(2n-1)$ equations (3.3) with an appropriate initial condition for $\tilde{\lambda}(0)$ (given in Lemma 3.2.1). Thus, for the purpose of analysis, we will henceforth assume that such a replacement is made in (3.6), and shall rewrite it in the time scale $\tau = \omega_c t$ as

$$\begin{aligned}
\omega_c \frac{dz}{d\tau} &= g(z, \hat{u} + a\nu(\tau)) \\
\frac{d\hat{u}_i}{d\tau} &= -\epsilon K_i \nu_i(\tau)(e_i^T z),\ \forall\ i,
\end{aligned} \tag{B.2}$$

where $z = [x^T, \tilde{\lambda}^T]^T,\ e_i \in \mathbb{R}^{2n-1}$ is the vector with all entries 0 except for a 1 in the i^{th} row, $\nu_i(\tau) = \sin(\overline{\omega}_i\tau)\ \forall\ i$ (recall (3.5)), and $g([x^T, \tilde{\lambda}^T]^T, u)$ is identical to the

R.H.S. of (3.3) with $r = f(u)$. Thus, an immediate consequence of Theorem 4 is that for a fixed u, $\dot{z} = g(z, u)$ has a globally exponentially stable equilibrium $z^{\text{eq}}(u)$ and, $\forall\ i$, $e_i^T z^{\text{eq}}(u) = \frac{1}{n}\sum_{i=1}^{n} f_i(u) = \frac{1}{n}W(u)$.

Next, observe that for a small ω_c, the system of equations in (B.2) is in the standard singular perturbation form [69, Chapter 11]. The structure of the proof is to first study the behavior of the slower time scale part of (B.2) (corresponding to $\dot{\hat{u}}$) by 'freezing' the value of z to $z^e(\hat{u} + a\nu(\tau))$. This slower system itself will be analyzed, for a small value of ϵ, using averaging arguments [69, Chapter 10]. Finally, results from singular perturbation analysis will be used to analyze the whole system (B.2). Let us begin the analysis by 'freezing' z as described above; we obtain:[1]

$$\frac{d\hat{u}_i}{d\tau} = -\epsilon K_i \nu_i(\tau) W(\hat{u} + a\nu(\tau)),\ \forall\ i. \tag{B.3}$$

Let $T = 2\pi \times \text{LCM}\{1/\overline{\omega}_1, ..., 1/\overline{\omega}_n\}$, where LCM denotes the least common multiple.

Proposition B.2.1. *Under the hypothesis of Theorem 5, there exists $\epsilon^* > 0$ and $a_c^* > 0$ such that for all $\epsilon \in (0, \epsilon^*)$ and $a_c \in (0, a_c^*)$, there exists a unique, exponentially stable, $T-$periodic solution $\hat{u}^p(\tau)$ to* (B.3) *such that $\|\hat{u}^p(\tau) - u^*\| = O(\epsilon + a_c^2)$.*

Proof. Substitute $\tilde{u} = \hat{u} - u^*$ in (B.3) to obtain

$$\frac{d\tilde{u}_i}{d\tau} = -\epsilon K_i \nu_i(\tau) W(u^* + \tilde{u} + a\nu(\tau)),\ \forall\ i. \tag{B.4}$$

Now, consider the autonomous average system

$$\frac{d\tilde{u}_i^{\text{ave}}}{d\tau} = -\epsilon \frac{1}{T}\int_0^T K_i \nu_i(\tau) W(u^* + \tilde{u}^{\text{ave}} + a\nu(\tau))d\tau,\ \forall\ i. \tag{B.5}$$

[1]The multiplier $\frac{1}{n}$ in front of $W(\cdot)$ is dropped for clarity.

Substituting W with its Taylor expansion around u^* in (B.5),

$$\begin{aligned}\frac{d\tilde{u}_i^{\text{ave}}}{d\tau} = -\epsilon\frac{1}{T}\int_0^T K_i\nu_i(\tau)\Bigg\{ & W(u^*) + \sum_{j=1}^{n}\frac{\partial W(u^*)}{\partial u_j}(\tilde{u}_j^{\text{ave}} + a_j\nu_j(\tau)) \\ & + \frac{1}{2!}\sum_{j,k=1}^{n}\frac{\partial^2 W(u^*)}{\partial u_j\,\partial u_k}(\tilde{u}_j^{\text{ave}} + a_j\nu_j(\tau))(\tilde{u}_k^{\text{ave}} + a_k\nu_k(\tau)) \\ & + \frac{1}{3!}\sum_{j,k,l=1}^{n}\Bigg(\frac{\partial^2 W(u^*)}{\partial u_j\,\partial u_k\,\partial u_l}(\tilde{u}_j^{\text{ave}} + a_j\nu_j(\tau)) \\ & \times(\tilde{u}_k^{\text{ave}} + a_k\nu_k(\tau))(\tilde{u}_l^{\text{ave}} + a_l\nu_l(\tau))\Bigg) + \cdots\Bigg\}d\tau,\end{aligned} \tag{B.6}$$

for each i. Note that $\frac{\partial W(u^*)}{\partial u} = 0$, and $\int_0^T \nu_i(\tau)d\tau = 0$, $\frac{1}{T}\int_0^T \nu_i^2(\tau)d\tau = \frac{1}{2}\ \forall\ i$. Also, the following can be verified under the hypothesis on the dither signal frequencies [10, Appendix A]:

$$\begin{aligned}\int_0^T \nu_i(\tau)\nu_j(\tau)d\tau &= 0\ \forall\ i \neq j;\\ \int_0^T \nu_i(\tau)\nu_j(\tau)\nu_k(\tau)d\tau &= 0\ \forall\ i,j,k.\end{aligned}$$

In view of these, for each i, (B.6) can be simplified

$$\begin{aligned}\frac{d\tilde{u}_i^{\text{ave}}}{d\tau} &= -\epsilon K_i\frac{1}{2!}\cdot\frac{a_i}{2}\Bigg(\sum_{k=1}^{n}\frac{\partial^2 W(u^*)}{\partial u_i\,\partial u_k}\tilde{u}_k^{\text{ave}} + \sum_{j=1}^{n}\frac{\partial^2 W(u^*)}{\partial u_j\,\partial u_i}\tilde{u}_j^{\text{ave}}\Bigg) \\ &\quad + \frac{1}{3!}\sum_{j=1}^{n}O(\epsilon a_c)\tilde{u}_i^{\text{ave}}\tilde{u}_j^{\text{ave}} + O(a_c^2)\sum_{p=4}^{\infty}\sum_{q=1,q\neq 2}^{p}O(\epsilon a_c^q\cdot(\tilde{u}^{\text{ave}})^{p-q}) \\ &= -\epsilon K_i\frac{a_i}{2}\sum_{k=1}^{n}\frac{\partial^2 W(u^*)}{\partial u_i\,\partial u_k}\tilde{u}_k^{\text{ave}} + \sum_{p=3}^{\infty}\sum_{q=1,q\neq 2}^{p}O(\epsilon a_c^q)\cdot O((\tilde{u}^{\text{ave}})^{p-q}),\end{aligned}$$

where $O(u^k)$ denotes $O(u_{i_1}\cdot u_{i_2}\cdot\cdots\cdot u_{i_k})$ terms. Thus, in vector notation the average system is given by (recall (3.5))

$$\frac{d\tilde{u}^{\text{ave}}}{d\tau} = -\frac{1}{2}\epsilon a_c\cdot D\cdot\frac{\partial^2 W(u^*)}{\partial u^2}\tilde{u}^{\text{ave}} + \sum_{p=3}^{\infty}\sum_{q=1,q\neq 2}^{p}O(\epsilon a_c^q)\cdot O((\tilde{u}^{\text{ave}})^{p-q}), \tag{B.7}$$

where D is a diagonal matrices with positive diagonal entries $D[i,i] = K_i\overline{a}_i$. Define

$$\Sigma_W = D\frac{\partial^2 W(u^*)}{\partial u^2}.$$

It is a nontrivial fact that the eigenvalues of the product of two positive definite matrices are real and positive; it can be found in [70, Theorem 3.32, pp. 126] or [71, Fact 8.18.17, pp. 518]. Consequently, $-\Sigma_W$ is Hurwitz.

Postulate the equilibrium of the average system (B.7) to be of the form $\tilde{u}^{\text{ave,eq}} = a_c b + O(a_c^2)$, where $b \in \mathbb{R}^n$, and substitute $\tilde{u}^{\text{ave,eq}}$ in (B.7) so that $-\epsilon a_c^2 \Sigma_W b + O(\epsilon a_c^3) = 0$. Comparing coefficients of a_c on both sides, since Σ_W is Hurwitz, we have $b = 0 \Rightarrow\ \tilde{u}^{\text{ave,eq}} = O(a_c^2)$.

Finally, the Jacobian of (B.7) is of the form

$$-\frac{1}{2}\epsilon a_c \Sigma_W + \sum_{p=3}^{\infty}\sum_{q=1,q\neq 2}^{p-1} O(\epsilon a_c^q)\cdot O((\tilde{u}^{\text{ave}})^{p-q-1}).$$

When evaluated at $\tilde{u}^{\text{ave,eq}}$ it is given by $-\epsilon a_c(\Sigma_W + O(a_c^2))$, which, for small enough a_c, is Hurwitz. We can thus conclude that, $\exists\ a_c^* > 0$ such that the equilibrium point $O(a_c^2)$ of the average system (B.5) is locally exponentially stable $\forall\ a_c \in (0, a_c^*)$.

By appropriate change of variables in (B.4) and (B.5) to move the equilibrium of (B.5) to the origin, we can apply [69, Theorem 10.4] to conclude that $\exists\ \epsilon^* > 0$, such that $\forall \epsilon \in (0, \epsilon^*)$, (B.4) has a unique, exponentially stable, periodic solution $\tilde{u}^{\text{p}}$, such that $\|\tilde{u}^{\text{p}}\| = O(\epsilon + a_c^2)$. □

Proof of Theorem 5. Let us translate the equilibrium of the reduced model (B.3) to the origin by making the coordinate transformation $v = \hat{u} - \hat{u}^{\text{p}}$. With this coordinate change, the system we wish to analyze, namely (B.2), is equivalently stated as (recall

$\hat{u}^{\mathrm{p}}$ satisfies (B.3))

$$\begin{aligned} \omega_c \frac{dz}{d\tau} &= g(z, v + \hat{u}^{\mathrm{p}}(\tau) + a\nu(\tau)) \\ \frac{dv_i}{d\tau} &= -\epsilon K_i \nu_i(\tau)(e_i^T z - W(\hat{u}^{\mathrm{p}}(\tau) + a\nu(\tau))),\ \forall\, i. \end{aligned} \tag{B.8}$$

While the reduced order system with z held frozen at its quasi equilibrium value $z^{\mathrm{eq}}(v + \hat{u}^{\mathrm{p}}(\tau) + \nu(\tau))$ is given, for all i, by

$$\frac{dv_i}{d\tau} = -\epsilon K_i \nu_i(\tau)(W(v + \hat{u}^{\mathrm{p}}(\tau) + a\nu(\tau)) - W(\hat{u}^{\mathrm{p}}(\tau) + a\nu(\tau))). \tag{B.9}$$

Due to Proposition B.2.1, system (B.9), for all i, has a locally exponentially stable equilibrium point at the origin for sufficiently small a_c and ϵ.

Following the notation of [69, Chapter 11], the 'boundary layer' system corresponding to (B.8) in the time scale $t = \tau/\omega_c$ with the change of coordinate $y = z - z^{\mathrm{eq}}(v + \hat{u}^{\mathrm{p}}(\tau) + a\nu(\tau))$, is given by

$$\frac{dy}{dt} = g(y + z^{\mathrm{eq}}(v + \hat{u}^{\mathrm{p}}(\tau) + a\nu(\tau)), v + \hat{u}^{\mathrm{p}}(\tau) + a\nu(\tau)), \tag{B.10}$$

where τ and v are considered constants. As a consequence of Theorem 4, the system (B.10) has a globally exponentially stable equilibrium at the origin, uniformly in (τ, v).

Finally, by applying Tikhonov's theorem [69, Theorem 11.2] to (B.8) with ω_c as the singular perturbation parameter, $\exists\, \omega_c^* > 0$ such that $\forall\, \omega_c \in (0, \omega_c^*)$, with $v(0)$ sufficiently close to the origin (and arbitrary $z(0)$ since the domain of attraction of the origin for (B.10) can be made arbitrarily large, $y(0)$ can be arbitrary), the solution of (B.8) is such that $v(t)$ converges to an $O(\omega_c)$ neighborhood of the origin, exponentially.

Thus, $\hat{u}$ converges exponentially to an $O(\omega_c)$ neighborhood of $\hat{u}^{\mathrm{P}}$ which, in turn, is in $O(a_c^2 + \epsilon)$ neighborhood of u^* for sufficiently small a_c and ϵ. Consequently, $\hat{u}$ converges to an $O(a_c^2 + \epsilon + \omega_c)$ neighborhood of u^* exponentially with appropriate choice of the parameters. The result follows by noting that $u = \hat{u} + \nu$. □

Bibliography

[1] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Transactions on Automatic Control*, vol. 31, no. 9, pp. 803–812, 1986.

[2] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 48–61, 2009.

[3] M. Rotkowitz and S. Lall, "A characterization of convex problems in decentralized control," *IEEE Transactions on Automatic Control*, vol. 51, pp. 274–286, Feb 2006.

[4] C. Langbort, R. Chandra, and R. D'Andrea, "Distributed control design for systems interconnected over an arbitrary graph," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1502–1519, Sept 2004.

[5] F. Borrelli and T. Keviczky, "Distributed LQR design for identical dynamically decoupled systems," *IEEE Transactions on Automatic Control*, vol. 53, pp. 1901–1912, Sept 2008.

[6] Y. Shoham, R. Powers, and T. Grenager, "If multi-agent learning is the answer, what is the question?," *Artificial Intelligence*, vol. 171, no. 7, pp. 365–377, 2007.

[7] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, 1996.

[8] H. P. Young, *Strategic learning and its limits*. Oxford University Press, 2004.

[9] D. Fudenberg, *The theory of learning in games*. MIT press, 1998.

[10] P. Frihauf, M. Krstic, and T. Basar, "Nash equilibrium seeking in noncooperative games," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1192–1207, 2012.

[11] M. S. Stankovic, K. H. Johansson, and D. M. Stipanovic, "Distributed seeking of Nash equilibria with applications to mobile sensor networks," *IEEE Transactions on Automatic Control*, vol. 57, no. 4, pp. 904–919, 2012.

[12] J. Marden and J. S. Shamma, "Game theory and distributed control," in *Handbook of Game Theory* (H. P. Young and S. Zamir, eds.), vol. 4, Elsevier. To appear; preprint available at http://www.prism.gatech.edu/ jshamma3/downloads/2099gta.pdf.

[13] R. Gopalakrishnan, J. R. Marden, and A. Wierman, "An architectural view of game theoretic control," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, pp. 31–36, ACM, 2011.

[14] N. Li and J. R. Marden, "Designing games for distributed optimization," in *Proc. of 50th IEEE Conference on Decision and Control and European Control Conference*, pp. 2434–2440, 2011.

[15] M. Zhu and S. Martnez, "Distributed coverage games for energy-aware mobile sensor networks," *SIAM Journal on Control and Optimization*, vol. 51, no. 1, pp. 1–27, 2013.

[16] J. R. Marden, G. Arslan, and J. S. Shamma, "Cooperative control and potential games," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics,*, vol. 39, no. 6, pp. 1393–1407, 2009.

[17] J. R. Marden and J. S. Shamma, "Revisiting log-linear learning: Asynchrony, completeness and payoff-based implementation," *Games and Economic Behavior*, vol. 75, no. 2, pp. 788–808, 2012.

[18] J. R. Marden, H. P. Young, G. Arslan, and J. S. Shamma, "Payoff based dynamics for multi-player weakly acyclic games," *SIAM Journal on Control and Optimization*, vol. 48, pp. 373–396, Feb 2009.

[19] H. P. Young, "Learning by trial and error," *Games and Economic Behavior*, vol. 65, no. 2, pp. 626–643, 2009.

[20] L. Y. Pao and K. E. Johnson, "Control of wind turbines," *Control Systems, IEEE*, vol. 31, no. 2, pp. 44–62, 2011.

[21] K. E. Johnson and N. Thomas, "Wind farm control: addressing the aerodynamic interaction among wind turbines," in *Proc. of the American Control Conference, 2009*, pp. 2104–2109, 2009.

[22] E. Bitar and P. Seiler, "Coordinated control of a wind turbine array for power maximization," in *Proc. of the American Control Conference, 2013*, pp. 2898–2904, 2013.

[23] P. M. Gebraad, F. C. van Dam, and J.-W. van Wingerden, "A model-free distributed approach for wind plant control," in *Proc. of the American Control Conference, 2013*, pp. 628–633, 2013.

[24] K. E. Johnson and G. Fritsch, "Assessment of extremum seeking control for wind farm energy production," *Wind Engineering*, vol. 36, no. 6, pp. 701–716, 2012.

[25] J. Marden, S. Ruben, and L. Pao, "A model-free approach to wind farm control using game theoretic methods," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1207–1214, 2013.

[26] M. Churchfield and S. Lee, "Nwtc design codes (SOWFA)." http://wind.nrel.gov/designcodes/simulators/sowfa/. Last modified 14-March-2012; accessed 14-March-2012.

[27] N. Ghods, P. Frihauf, and M. Krstic, "Multi-agent deployment in the plane using stochastic extremum seeking," in *Proc. of 49th IEEE Conference on Decision and Control*, pp. 5505–5510, 2010.

[28] N. Sandell Jr, P. Varaiya, M. Athans, and M. Safonov, "Survey of decentralized control methods for large scale systems," *IEEE Transactions on Automatic Control*, vol. 23, no. 2, pp. 108–128, 1978.

[29] Y.-C. Ho and K.-C. Chu, "Team decision theory and information structures in optimal control problems–part I," *IEEE Transactions on Automatic Control*, vol. 17, pp. 15–22, Feb 1972.

[30] V. Blondel and J. N. Tsitsiklis, "Np-hardness of some linear control design problems," *SIAM Journal on Control and Optimization*, vol. 35, no. 6, pp. 2118–2127, 1997.

[31] B. S. R. Pradelski and H. P. Young, "Learning efficient Nash equilibria in distributed systems," *Games and Economic behavior*, vol. 75, no. 2, pp. 882–897, 2012.

[32] J. R. Marden, H. P. Young, and L. Y. Pao, "Achieving Pareto optimality through distributed learning," in *Proc. of 51st Annual IEEE Conference on Decision and Control*, pp. 7419–7424, 2012.

[33] K. Aiba, "Waiting times in evolutionary dynamics with time-decreasing noise." preprint, 2011.

[34] H. P. Young, "The evolution of conventions," *Econometrica: Journal of the Econometric Society*, pp. 57–84, 1993.

[35] A. Menon and J. S. Baras, "Convergence guarantees for a decentralized algorithm achieving Pareto optimality," in *Proc. of the 2013 American Control Conference (ACC)*, pp. 1932–1937, 2013.

[36] A. Menon and J. S. Baras, "A distributed learning algorithm with bit-valued communications for multi-agent welfare optimization," in *Proc. of the 52nd Annual IEEE Conference on Decision and Control*, pp. 2406–2411, 2013.

[37] A. Menon and J. S. Baras, "A distributed learning algorithm with bit-valued communications for multi-agent welfare optimization," tech. rep., Institute of Systems Research Technical Report, 2013.

[38] Y. Tan, W. Moase, C. Manzie, D. Nesic, and I. Mareels, "Extremum seeking from 1922 to 2010," in *Prof. of the 29th Chinese Control Conference (CCC)*, pp. 14–26, IEEE, 2010.

[39] M. Krstić and H.-H. Wang, "Stability of extremum seeking feedback for general nonlinear dynamic systems," *Automatica*, vol. 36, no. 4, pp. 595–601, 2000.

[40] A. Menon and J. S. Baras, "Collaborative extremum seeking for welfare optimization." Submitted to the IEEE Conference on Decision and Control 2014.

[41] P. Varaiya, "Smart cars on smart roads: problems of control," *IEEE Transactions on Automatic Control*, vol. 38, pp. 195–207, Feb 1993.

[42] P. Seiler, A. Pant, and K. Hedrick, "Disturbance propagation in vehicle strings," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1835–1842, Oct 2004.

[43] P. Barooah and J. Hespanha, "Error amplification and disturbance propagation in vehicle strings with decentralized linear control," in *Proc. of the 44th IEEE CDC, 2005 and the European Control Conference, 2005*, pp. 4964–4969, Dec 2005.

[44] R. Middleton and J. Braslavsky, "String instability in classes of linear time invariant formation control with limited communication range," *IEEE Transactions on Automatic Control*, vol. 55, pp. 1519–1530, July 2010.

[45] H. Hao, P. Barooah, and J. Veerman, "Effect of network structure on the stability margin of large vehicle formation with distributed control," in *Proc. of the 49th IEEE CDC, 2010*, pp. 4783–4788, Dec 2010.

[46] H. Hao and P. Barooah, "On achieving size-independent stability margin of vehicular lattice formations with distributed control," *IEEE Transactions on Automatic Control*, vol. 57, no. 10, pp. 2688–2694, 2012.

[47] B. Bamieh, M. Jovanovic, P. Mitra, and S. Patterson, "Coherence in large-scale networks: Dimension-dependent limitations of local feedback," *IEEE Transactions on Automatic Control*, vol. 57, no. 9, pp. 2235–2249, 2012.

[48] A. Menon and J. S. Baras, "Expander families as information patterns for distributed control of vehicle platoons," in *Proc. of 3rd IFAC Workshop on*

Distributed Estimation and Control in Networked Systems, pp. 288–293, Santa Barbara, California, September, 14-15, 2012.

[49] X. Tan, W. Xi, and J. S. Baras, "Decentralized coordination of autonomous swarms using parallel Gibbs sampling," *Automatica*, vol. 46, no. 12, pp. 2068–2076, 2010.

[50] W. Xi, X. Tan, and J. S. Baras, "Gibbs sampler-based coordination of autonomous swarms," *Automatica*, vol. 42, no. 7, pp. 1107–1119, 2006.

[51] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 721–741, 1984.

[52] P. J. M. Laarhoven and E. H. L. Aarts, *Simulated Annealing: Theory and Applications*, vol. 37 of Mathematics and its Applications. Springer, 2010.

[53] J. Robles, "Evolution with changing mutation rates," *Journal of Economic Theory*, vol. 79, no. 2, pp. 207–223, 1998.

[54] M. Pak, "Stochastic stability and time-dependent mutations," *Games and Economic Behavior*, vol. 64, no. 2, pp. 650–665, 2008.

[55] D. Mitra, F. Romeo, and A. Sangiovanni-Vincentelli, "Convergence and finite-time behavior of simulated annealing," *Advances in Applied Probability*, pp. 747–771, 1986.

[56] R. Freeman, P. Yang, and K. Lynch, "Stability and convergence properties of dynamic average consensus estimators," in *45th IEEE Conference on Decision and Control, 2006*, pp. 338–343, Dec 2006.

[57] I. Matei and J. Baras, "A non-heuristic distributed algorithm for non-convex constrained optimization," tech. rep., Institute for Systems Research, 2013.

[58] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

[59] A. Ghaffari, M. Krstic, and S. Seshagiri, "Power optimization and control in wind energy conversion systems using extremum seeking," in *Proc. of the American Control Conference*, pp. 2241–2246, 2013.

[60] Y. Tan, D. Nešić, and I. Mareels, "On non-local stability properties of extremum seeking control," *Automatica*, vol. 42, no. 6, pp. 889–903, 2006.

[61] L. Glielmo, "Vehicle-to-vehicle/vehicle-to-infrastructure control," in *The Impact of Control Technology* (T. Samad and A. Annaswamy, eds.), p. 246, IEEE Control Systems Society, 2011.

[62] S. Hoory, N. Linial, and A. Wigderson, "Expander graphs and their applications," *Bulletin of the American Mathematical Society*, vol. 43, no. 4, pp. 439–561, 2006.

[63] M. Krebs and A. Shaheen, *Expander Families and Cayley Graphs: A Beginner's Guide*. Oxford Univ. Press, 2011.

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

[65] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[66] A. R. Teel and D. Popovic, "Solving smooth and nonsmooth multivariable extremum seeking problems by the methods of nonlinear programming," in *Proc. of the 2001 American Control Conference*, vol. 3, pp. 2394–2399, IEEE, 2001.

[67] A. R. Mesquita, J. P. Hespanha, and K. Åström, "Optimotaxis: A stochastic multi-agent optimization procedure with point measurements," in *Hybrid Systems: Computation and Control*, pp. 358–371, Springer, 2008.

[68] P. Brémaud, *Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues*, vol. 31. Springer-Verlag, 1999.

[69] H. K. Khalil, *Nonlinear systems*, vol. 3. Prentice hall Upper Saddle River, 2002.

[70] J. R. Schott, *Matrix analysis for statistics*. Wiley, second ed., 2005.

[71] D. S. Bernstein, *Matrix mathematics: theory, facts, and formulas*. Princeton University Press, second ed., 2009.