An Architectural View of Game Theoretic Control

Thesis by

Ragavendran Gopalakrishnan

In Partial Fulfillment of the Requirements
for the Degree of
Master of Science

California Institute of Technology
Pasadena, California

2010
(Submitted May 27, 2010)

To my parents.

Acknowledgements

First, I would like to express my heartfelt gratitude to my advisor, Dr. Adam Wierman, whose valuable guidance during the past two years has been instrumental in shaping my approach towards research in computer science. Despite his busy schedules, he always makes himself available whenever I need him (physically or via e-mail). Whenever I have trouble understanding a new concept, he is patient and guides my thinking process, enabling me to work my way through it. The relationship we have established between ourselves is one that will continue for years to come. The second person I met in Caltech (the first being Adam), was Minghong. He joined the Ph.D. program with me, and since then, we've grown to become very good friends. My first research collaboration was with Jason, who was then a postdoctoral student. I'm thankful to him for the hours that we spent together, trying to figure things out and discuss potential directions for my future research. Urtzi, who visited Caltech during my early days, deserves a special mention for his encouraging advice on discovering my own strengths. I'm also thankful to Mathieu, my option representative, for helping me get oriented towards the degree program. I would also like to thank the members of the Rigorous Systems Research Group (RSRG) for collectively providing a sustaining research environment that keeps me motivated. Many thanks to the staff of Annenberg (and Jorgensen before that) – specifically Jeri, Maria and Sydney – who were quick with any administrative processing that I occasionally had to get done. And last but not the least, I'm indebted to my family (mom, dad and bro) for providing the mental support from afar, and for enduring the separation. This is not the end, and I'm sure I'm going to enjoy the rest of my time here at Caltech!

Abstract

Resource allocation has long been a fundamental research problem across several disciplines. While traditional approaches to this problem were centralized, recent research has focussed on distributed solutions for resource allocation, for reasons of scalability, reliability and efficiency in many real-world applications. Game-theoretic control is a promising new approach for distributed resource allocation. In this thesis, we describe how game-theoretic control can be viewed as having an intrinsic layered architecture, which provides a modularization that simplifies the control design. We illustrate this architectural view by presenting details about one particular instantiation using potential games as an interface. This example serves to highlight the strengths and limitations of the proposed architecture while also illustrating the relationship between game-theoretic control and other existing approaches to distributed resource allocation. We also demonstrate the power of this approach by reformulating the power control problem in sensor networks as a game-theoretic control problem in the potential games instantiation of our framework. This allows us to relax several assumptions made by previous contributions, and consider more complex objective functions.

Contents

Chapter 1

Introduction

Resource allocation is a fundamental problem that arises in nearly all computer systems. Traditional approaches to determine efficient allocations involved mostly centralized algorithms [2, 3, 19]. However, in many modern applications, these centralized algorithms are not applicable and/or desirable, due to their poor reliability, efficiency and scalability. As a result, increasingly, resource allocation is a problem that needs to be solved in a distributed, decentralized manner, e.g., power control and frequency selection problems in wireless networks [5, 7, 9, 18, 28] and coverage problems in sensor networks [10, 38]. Resultantly, there is a large and growing literature that focuses on developing distributed resource allocation protocols. This is an extremely diverse literature where protocols are designed using a wide variety of tools, e.g., distributed optimization [41, 60], distributed control [33, 48], physics-inspired control (e.g. Gibbs-sampler-based control) [28, 40], and game-theoretic control [4, 6, 38, 58].

In this thesis, we focus on game-theoretic control, which is a promising new approach for distributed resource allocation. The game-theoretic approach involves modeling the interactions of agents within a noncooperative game where the agents are 'self-interested'. This is motivated by the fact that the underlying decision making architecture in economic systems is identical to the desired decision making architecture in distributed engineering systems, i.e., local decisions based on local information where the global behavior emerges from a compilation of these local decisions. This parallel makes it possible to utilize the broad set of economic/game-theoretic tools in distributed control. However, a key distinction between game theory for economic systems and game theory for engineering systems is that decision makers are inherited in economic systems while decisions makers are designed in engineering systems. This difference means that using game-theoretic tools for distributed control requires a new perspective on the economic literature.

Applying game-theoretic control requires specifying decision makers, their respective choices, their objective/utility functions, and the learning rules for the agents. In this thesis, we focus on two of these components: (i) the design of the agents' utility functions, i.e., *utility design*, and (ii) the design of the distributed learning rules for the agents, i.e., *learning design*. The goal is to design the utility functions and the learning rules so that the emergent global behavior is desirable. There are wide-ranging advantages to the game-theoretic approach including robustness to failures and environmental disturbances, minimal communication requirements, and improved scalability.

Game-theoretic resource allocation designs are increasingly popular in a variety of wireless and sensor network applications, e.g., channel access control in wireless networks [7, 28], coverage problems in sensor networks [10, 38], and power control in both [5, 9, 18]. A comprehensive survey of applications can be found in [6]. However, nearly all of these designs are highly application-specific, with both the utility and learning designs crafted carefully for the specific setting. There have been only a few papers that focus on general designs and even these papers tend to focus on only one aspect of the design – either the utility design, e.g., [38, 39], or the learning design, e.g., [34, 35, 53].

Our contribution in this thesis is to present an 'architectural' view of game-theoretic control as a whole. We describe, at a high-level, our proposed architecture for game-theoretic control in Chapter 2. Then, to make the ideas more concrete, in Chapter 3 we present details about one particular instantiation of the architecture, based on potential games, which aligns with a number of current approaches. To bring out the strength of our approach, in Chapter 4, we use this instantiation to propose a solution for the power control problem in sensor networks in a much more general and powerful model than those considered in literature, allowing a complex objective function to be optimized in a distributed manner. Next, in Chapter 5, we provide some new results highlighting how two other existing approaches for distributed system design (distributed constraint optimization and Gibbs-sampler-based control) can be viewed as instances of designs in the potential-games-based architecture. Finally, in Chapter 6 we conclude with a discussion of a number of important research directions outside of the potential games framework that are suggested by the architectural view of game-theoretic control.

Chapter 2

A Layered Architecture

As described in the introduction, game-theoretic control has two key design tasks – utility design and learning design. While each of these tasks, by itself, is complex and subject to diverse application-specific constraints, game theoretic control is made even more difficult due to the complex interactions between the two – not all learning rules lead to desirable global behavior when matched with a particular utility design, and vice versa. As a result, most applications of game-theoretic control have performed application-specific *co-design* of the utility functions and the learning rule, e.g., [6, 21, 49, 50], which is typically a difficult task.

However, it is clearly desirable to decouple these two design choices. The central question that this thesis explores is how to achieve a *modularization/decoupling* of utility and learning design. Such a decoupling would allow for the development of a rich set of utility and learning designs from which a design can be chosen 'off the shelf' according to the requirements of the resource allocation problem being considered.

It turns out that this sort of modularization is possible for game-theoretic control – and many of the prior utility and learning designs can immediately be viewed within a modular architecture. In particular, game-theoretic control naturally falls into an 'hourglass' architecture, that is a cornerstone of computer system design. An hourglass architecture is a type of layered architecture where within the highest and lowest layers there is a large diversity of available designs, but near the middle, or waist, the design is highly constrained. The most famous example of such an 'hourglass' architecture is the IP network stack [31, 45, 61]; however, this architecture is quite common in computer systems and has also been observed in wide-ranging areas such as biology [15]. In the context of the network stack, there is large diversity at the application and physical layers, but the network layer is very restrictive (IP). The benefit of the small 'waist' of an hourglass architecture is that it provides a simple interface that allows the 'virtualization' of the lower layers for the higher layers, e.g., IP virtualizes the details of the network for applications.

In the context of game-theoretic control, there are diverse sets of utility designs and learning rules, and the goal is to simplify design by virtualizing the learning rules and utility designs from

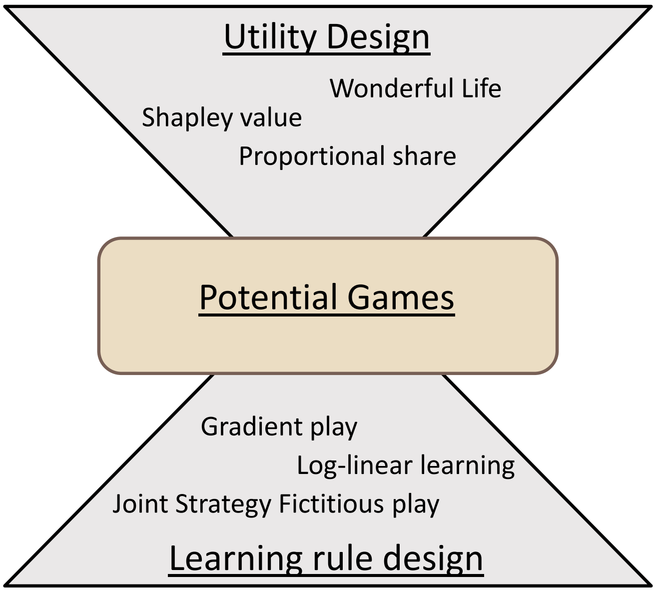

Figure 2.1: An illustration of the 'hourglass' architecture using potential games as the interface.

each other's perspective. To accomplish this, it is necessary to define a constrained interface that binds the two (like IP for the network stack). It is appealing to let the class of allowable games (that result from the utility designs) be the waist. Specifically, we impose a restriction on the class of games that could emerge from a utility design, and enforce the same restriction on the class of games for which a learning rule would work. This enforced structure then serves as the interface between utility design and learning design; thus providing the desired modularity.

In Chapter 3, we discuss a concrete example of this constrained interface, the class of *potential games*. (See Figure 2.1 for an illustration.) This interface requires utility designs to guarantee that the resulting game is a potential game and requires learning rules to guarantee to provide desirable behavior when run on a potential game. Thus, requiring the structure of potential games enforces additional *constraints* on utility and learning design, but also *deconstrains* at a higher level by allowing modularization. Though this layered architecture was not explicitly used in prior work, the modularization provided by potential games underlies many successful examples of game-theoretic control [9, 23, 30, 38]. The change in perspective provided by this architectural view is not simply superficial; it highlights that the utility and learning designs in these papers can be 'mixed and matched' while still obtaining the same performance.

Though we focus on potential games in much of this thesis, it is important to note that they are not the only choice for the interface – we discuss moving beyond potential games in Chapter 6.

Chapter 3

Layering via Potential Games

We now illustrate how using potential games as an interface provides modularization of utility and learning design. We focus on potential games because many recent applications of game-theoretic control have relied on potential games, e.g., [9, 23, 30, 38]. A key reason that potential games are a powerful choice for the interface is that they are a highly studied class of games in the economics literature, e.g., [16, 42, 52, 59, 66] and so there is a large literature of results that can be used in the context of game-theoretic control for both utility and learning design.

In this chapter, we highlight the variety of utility and learning designs that have been adapted from the economics literature and can be used interchangeably 'off the shelf', greatly simplifying the task of design. However, we also illustrate that layering via potential games has some limitations, which highlights the need to consider other interfaces as well. In order to illustrate these issues formally, we first define a simple resource allocation model.

3.1 Preliminaries

3.1.1 A model for resource allocation

We use a simple, but general, model of resource allocation problems in this thesis. Consider a set of distributed agents $N = \{1, \ldots, n\}$ and a set of resources $R = \{r_1, \ldots, r_m\}$ that are to be shared by the agents. Each agent $i \in N$ is capable of selecting potentially multiple resources in R; therefore, we say that agent i has action set $\mathcal{A}_i \subseteq 2^R$. An allocation, or an action profile, is represented by a tuple $a = (a_1, a_2, ..., a_n) \in \mathcal{A}$ where the set of possible allocations is denoted by $\mathcal{A} = \mathcal{A}_1 \times \ldots \times \mathcal{A}_n$. We will frequently denote an allocation a as (a_i, a_{-i}) where $a_{-i} \in \mathcal{A}_{-i} = \prod_{j \neq i} \mathcal{A}_j$ denotes the allocation of all agents except agent i.

The social welfare function $W(a)$ captures the global valuation of the agent allocation. In general, a resource allocation design seeks to find an allocation that optimizes the global welfare. In this work, we assume $W(a)$ is *linearly separable* across resources, i.e., $W(a) = \sum_{r \in R} W_r(\{a\}_r)$, where $\{a\}_r =$

$\{i \in N : r \in a_i\}$ is the set of agents that are allocated to resource r in a and $W_r : 2^N \to \mathbb{R}_+$ is the local welfare function for resource r. Hence, the welfare generated at a particular resource depends only on which agents are allocated to that resource. While separable welfare functions cannot model all resource allocation problems, they are useful for several important classes of problems including, but not limited to, routing over a network [51], vehicle target assignment problem [43], sensor coverage [10, 38], content distribution [25], network coding [36].

Further, we restrict our attention to *submodular* welfare functions, i.e., for each resource $r \in R$ and any player sets $X \subseteq Y \subseteq N$,

$$W_r(X) + W_r(Y) \geq W_r(X \cup Y) + W_r(X \cap Y).$$

A variety of resource allocation problems such as power control and coverage problems in sensor networks [9, 38], wireless access point assignment and frequency selection [28], outbreak detection [32], path planning [56], clustering [44], and influence maximization [29] all have submodular welfare functions.

3.1.2 Resource allocation games

Our goal is to utilize game theory to obtain distributed solutions to such resource allocation problems. This goal requires modeling the interactions of the agents in a noncooperative game theoretic environment where the agents act in a self-interested fashion. While we inherit the players N, the welfare function W, and the action sets $\{\mathcal{A}_i\}_{i \in N}$, we are left to design a utility function for each player of the form $U_i : \mathcal{A} \to \mathbb{R}$. A resource allocation game G is then defined by the tuple $G = (N, \{\mathcal{A}_i\}, \{U_i\}, W)$.

In general, a system designer has free reign in the design of utility functions; however, layering via potential games requires that the utility functions lead to a potential game. Formally, a game is called a *potential game*, if there exists a potential function $\Phi : \mathcal{A} \to \mathbb{R}$ such that $\forall\ i$, $\forall\ a_{-i} \in \mathcal{A}_{-i}$, and $\forall\ a_i, a_i' \in \mathcal{A}_i$:

$$\Phi(a_i, a_{-i}) - \Phi(a_i', a_{-i}) = U_i(a_i, a_{-i}) - U_i(a_i', a_{-i}).$$

Potential games possess several nice properties that can be utilized in distributed control. One such property is the guaranteed existence of a *pure Nash equilibrium* in any potential game. A *(pure Nash) equilibrium* is an action profile $a^* \in \mathcal{A}$ such that for each player i,

$$U_i(a_i^*, a_{-i}^*) \;=\; \max_{a_i \in \mathcal{A}_i} U_i(a_i, a_{-i}^*).$$

In a distributed system, a pure Nash equilibrium takes on the role of a stable operating point.

To discuss the efficiency of games we use the notions of Price of Anarchy (PoA) and Price of

Stability (PoS) [46], which compare the welfare of the set of equilibria to the globally optimal welfare. Let $\mathcal{G}$ denote a set of games and $\mathcal{S}(G)$ denote the set of equilibria for a game G. Then,

$$PoA(\mathcal{G}) = \inf_{G\in\mathcal{G}} \left(\min_{a^*\in\mathcal{S}(G)} \frac{W(a^*)}{\max_{a\in\mathcal{A}} W(a)} \right)$$

$$PoS(\mathcal{G}) = \inf_{G\in\mathcal{G}} \left(\max_{a^*\in\mathcal{S}(G)} \frac{W(a^*)}{\max_{a\in\mathcal{A}} W(a)} \right)$$

So, the PoA measures the worst-case efficiency of any equilibrium while the PoS measures the worst-case efficiency of the best equilibrium across all games.

3.2 Utility design

There are several desirable properties that a global planner must consider when designing utility functions. These properties include the following conditions on utility functions:

(a) computable using only local information,

(b) guarantee the existence of an equilibrium,

(c) computable in polynomial time,

(d) guarantee PoS=1,

(e) guarantee a PoA close to 1, and

(f) guarantee that utilities are budget-balanced, i.e., the sum of the utilities of all the players is equal to the social welfare for every action profile.

In the context of the considered resource allocation games, we want to design utility functions that are *linearly separable*, i.e., that satisfy

$$U_i(a_i, a_{-i}) = \sum_{r\in a_i} f_r(i, \{a\}_r),$$

where $f_r : N \times 2^N \rightarrow \mathbb{R}$, i.e., a player's utility is only dependent on the resources the players choose and the other players that choose the same resources.

Recent work [38] has observed that the task of utility design is strongly related to the cost sharing literature in economics. Here, we present a few promising utility designs that have emerged, though it is important to point out that there are a variety of other possible choices, e.g., [13, 38].

3.2.1 Wonderful life utility design (WLU)

Wolpert and Tumer [62] introduced this design in 1999, in the context of designing reward (utility) functions for distributed agents in reinforcement learning algorithms that would result in optimizing the global utility function (social welfare). This design defines the utility of each player i as their marginal contribution to the social welfare,

$$\begin{aligned} U_i(a_i, a_{-i}) &= W(a_i, a_{-i}) - W(\emptyset, a_{-i}) \\ &= \sum_{r \in a_i} \left(W_r(\{a\}_r) - W_r(\{a\}_r \setminus i) \right). \end{aligned}$$

Note that this is linearly separable. Additionally, it has been shown in [38] that WLU leads to a potential game with $\Phi = W$ [62]. As a consequence, the social optimum is also a Nash equilibrium and so, $PoS = 1$. Also, for submodular resource allocation games, $PoA = 1/2$. But, WLU is not budget-balanced.

3.2.2 Shapley value utility design (SVU)

Shapley [55] introduced this design in the context of coalitional games in 1953, aiming to propose the fairest allocation of collectively gained profits between the several collaborative agents. This design defines the utility of each player i as their Shapley value to the social welfare,

$$U_i(a_i, a_{-i}) = \sum_{r \in a_i} Sh_i(\{a\}_r; W_r)$$

where $Sh_i(\{a\}_r; W_r)$, the Shapley value of player i at resource r, is

$$\sum_{S \subseteq \{a\}_r : i \in S} \frac{(|\{a\}_r| - 2)!(|S| - 1)!}{|\{a\}_r|!} (W_r(S) - W_r(S \backslash \{i\})).$$

Notice that this utility design is also linearly separable. Furthermore, SVU leads to a potential game [59], with the following potential function:

$$\Phi(a) = \sum_{r \in R} \sum_{S \subseteq \{a\}_r} \frac{1}{|S|} \left(\sum_{T \subseteq S} (-1)^{|S|-|T|} W_r(T) \right)$$

Also, SVU is budget-balanced, and guarantees a $PoA = PoS = 1/2$ for submodular resource allocation games [38]. However, in general, SVU is not polynomial-time computable [14].

3.2.3 Weighted Shapley value utility design (WSVU)

This design [26, 55] is similar to the SVU design, except that each player i has as their utility, their *weighted* Shapley value to the social welfare,

$$U_i(a_i, a_{-i}) = \sum_{r \in a_i} \widetilde{Sh}_i(\{a\}_r; W_r)$$

where $\widetilde{Sh}_i(\{a\}_r; W_r)$, the weighted Shapley value of player i at resource r, is

$$\sum_{S \subseteq \{a\}_r : i \in S} \frac{w_i}{\sum_{j \in S} w_j} \left(\sum_{T \subseteq S} (-1)^{|S|-|T|} W_r(T) \right).$$

where $w_i \in \mathbb{R}_+$ denotes the weight of player i. Note that when all players have the same weight, WSVU reduces to SVU. WSVU also leads to a potential game, but with a potential function that doesn't have a clean closed form, and can only be computed recursively. Like SVU, its weighted version is also budget-balanced, and guarantees a $PoA = PoS = 1/2$ for submodular resource allocation games. WSVU is also not polynomial-time computable in general.

3.3 Learning design

As with utility design, there are several desirable properties that a global planner must consider when designing distributed learning rules. These properties include:

(a) the asymptotic global behavior,

(b) equilibrium selection,

(c) informational dependencies, and

(d) convergence rates.

Learning design takes on the form of a one-shot repeated game where at each time $t \in \{0, 1, 2, \ldots\}$ each player $i \in N$ simultaneously chooses an action $a_i(t) \in \mathcal{A}_i$ according to probability distribution $p_i(t)$ and receives a utility $U_i(a(t))$ where $a(t) = (a_1(t), \ldots, a_n(t))$. We refer to $p_i(t)$ as the *strategy* of player i at time t and denote the probability that player i will play action a_i at time t by $p_i^{a_i}(t)$. A players strategy at time t can rely only on actions (and their corresponding utilities) from times $\{0, 1, 2, \ldots, t-1\}$.

In the most general form, this mechanism can be expressed as

$$p_i(t) = F_i(a(0), ..., a(t-1); U_i).$$

In general, a system designer would like to assign each agent a mechanism $F_i(\cdot)$ to ensure as many desirable properties as possible.

One of the benefits of using potential games as the interface is that there are several learning designs available in the literature, each providing different guarantees. Therefore, the system designer can choose the learning design that is most appropriate for the given application off the shelf without the need for application specific adjustments.

Here, we present a few promising learning designs that have emerged, though it is important to point out that there are a variety of other possible choices, e.g., [20, 53, 54, 66].

3.3.1 Joint strategy fictitious play (JSFP)

In many settings, such as routing over a network, the information that each agent has access to is extremely limited. The JSFP algorithm [35] works within these limitations to provide strong guarantees on the resulting asymptotic behavior. This algorithm requires that each agent maintains a hypothetical payoff for each action a_i of the form

$$V_i^{a_i}(t) = \sum_{\tau=0}^{t-1} \frac{1}{t} U_i(a_i, a_{-i}(\tau)).$$

Note that this hypothetical payoff can be computed recursively and that a player only needs access to the payoff for alternative actions at each time step. Using this hypothetical payoff, the strategy of player i at time t is of the form

$$p_i^{a_i(t-1)}(t) = \epsilon, \quad p_i^{a_i^*}(t) = 1 - \epsilon$$

where $a_i^* \in \arg\max_{a_i \in \mathcal{A}_i} V_i^{a_i}(t)$ and $\epsilon > 0$ is referred to as 'inertia'. For any potential game, if all players adhere to this strategy, then the global behavior will converge almost surely to a pure Nash equilibrium [35]. Hence, the efficiency is bounded by the price of anarchy of the utility design used.

3.3.2 Log-linear learning

In many settings, it is desirable to converge to a specific Nash equilibrium, rather than just any equilibrium. This is termed 'equilibrium selection' and is not provided by JSFP. Log-linear learning [8, 37] provides equilibrium selection, but it also requires more structure on the learning environment. Under log-linear learning, at each time t, one agent $i \in N$ is randomly selected to update its action while all other agents are required keep their actions fixed, i.e., $a_{-i}(t) = a_{-i}(t-1)$. Player i plays a strategy at time t of the form

$$p_i^{a_i}(t) \quad = \quad \frac{e^{\frac{1}{T}U_i(a_i,a_{-i}(t-1))}}{\sum_{a_i'\in\mathcal{A}_i} e^{\frac{1}{T}U_i(a_i',a_{-i}(t-1))}}$$

where $T \geq 0$ is a temperature coefficient. For any potential game, if all players adhere to this mechanism, then the global behavior obeys an ergodic Markov chain with a unique stationary distribution given by:

$$\mu^a \quad = \quad \frac{e^{\frac{1}{T}\Phi(a)}}{\sum_{a'\in\mathcal{A}} e^{\frac{1}{T}\Phi(a')}}.$$

As the process cools (anneals), i.e., $T \to 0^+$, all the weight of the stationary distribution falls on the action profiles that maximize the potential function, thus providing equilibrium selection [37]. Often, e.g., for WLU where $\Phi = W$, this guarantees that the global performance achieves the bound set by the PoS, highlighting the importance of the PoS as a measure for efficiency.

3.3.3 Gradient play

In gradient play [17, 54], each player maintains running averages (empirical frequencies) of the past actions of all players:

$$q_i^j(t) = \frac{1}{t}\sum_{\tau=0}^{t-1} \mathbb{1}\{a_i(\tau) = j\} \quad \forall\, j \in \mathcal{A}_i$$

where $\mathbb{1}$ denotes the indicator function. Let $p = (p_1, p_2, \ldots, p_n)$ denote a mixed strategy profile, where p_i denotes the vector of probabilities of choosing actions for player i. Define the expected utility function $U_i(p)$ as follows (note the overloading of U_i here):

$$U_i(p) = \sum_{a\in\mathcal{A}} \left(\prod_{k=1}^{n} p_k^{a_k}\right) U_i(a)$$

The strategy of each player at time t is then given by:

$$p_i(t) = \Pi_{\Delta_i}\left[q_i(t) + \rho_i(t)\ \nabla_{p_i} U_i(p)\big|_{p=q(t)}\right]$$

where $\Pi_{\Delta_i}[\cdot]$ denotes the orthogonal projection onto the $|\mathcal{A}_i|$-dimensional probability simplex, and $\rho_i(t) > 0$ is the step size at time t. It can been shown for potential games [17], that the global behavior will converge to a Nash equilibrium if all the players employ gradient play in an almost cyclical fashion with $\rho_i(t) \in \left[\epsilon_i, \frac{2}{K_i} - \epsilon_i\right]$, where $0 < \epsilon < \frac{2}{K_i}$, and $K_i > 0$ is a Lipschitz constant of the marginal payoff, i.e., $\forall\ q_i, q_i', q_{-i}$, $\left\| \nabla_{p_i} U_i(p)\big|_{p=(q_i,q_{-i})} - \nabla_{p_i} U_i(p)\big|_{p=(q_i',q_{-i})} \right\| \leq K_i \left\|q_i - q_i'\right\|$.

3.4 Limitations

To this point, we have highlighted that using potential games as an interface provides modularization and diverse options for utility and learning design. However, recent research [39] has identified that there are also fundamental limitations of layering via potential games. In particular, it is not possible for a utility design to achieve all of the properties one might desire.

For example, there are conflicts between maintaining a $PoS = 1$ and being budget-balanced. Specifically, a limitation of layering via potential games is that *any linearly separable, budget-balanced utility design that guarantees the existence of an equilibrium for all resource allocation games has* $PoS \leq \frac{1}{2}$ [39].

WLU and SVU provide an illustration of this limitation: SVU is linearly separable and budget-balanced but has $PoS = 1/2$ while WLU is linearly separable and has $PoS = 1$ but is not budget-balanced.

Notice that this limitation focuses on budget-balanced utility design. While being local[1] and having a PoS close to one are nearly always desirable, not all applications require budget-balanced utilities. Thus, in many cases this limitation may not be relevant. However, limitations of potential games have only begun to be studied, and there are likely more severe ones yet to be discovered.

[1]The notion of locality is meant to ensure that a design is practically implementable, i.e., it only depends on information local to the resource. Formally, a set of distribution rules $\{f_r\}$ is *local* if and only if for all r, $f_r(i, S)$ does not depend on any information about other resources, e.g., the number of resources, players not at r, the other actions available to players at r.

Chapter 4

An Example: Power Control in Sensor Networks

In this chapter, we consider a specific application, namely, power control in sensor networks, as an example in which to illustrate the power of the potential-games-based architecture. First, we briefly introduce the problem and survey existing results in Section 4.1. Then, in Section 4.2, we present our model for this problem, which is much more general and powerful than those existing in literature. Finally, in Section 4.3, we demonstrate how this problem can be viewed in the potential-games-based architecture for game-theoretic control, using which we propose a practical solution to the problem.

4.1 Related work

While the principal objective in sensor networks is to maximize the sensing capability of the network as a whole, there are other concerns. For example, participant nodes in sensor networks typically run on batteries which have limited power. Therefore, power management is a dominating problem. In most cases, these networks are multi-hop, meaning that nodes must forward packets for each other. So, in addition to power minimization, maintaining connectivity is also important. Typically, when there are multiple conflicting objectives like in this case, there are tradeoff parameters that are introduced in order to combine them into a single overall objective function.

One of the first distributed schemes for efficient power management in sensor networks appeared in [12]. They proposed *Span*, a randomized and distributed power-saving technique that reduces energy consumption without significantly diminishing the capacity or connectivity of the network. The key observation is that when a region of a shared-channel wireless network has a sufficient density of nodes, only a small number of them need be on at any time to forward traffic for active connections. Following this, several more distributed solutions have been published [22, 24, 27, 63, 64, 65]. However, all of them are based on heuristic arguments that may work well for specific application scenarios, but lack a theoretical performance guarantee.

Campos-Nañez, Garcia and Li [9] were the first to analyze the problem in a game theoretic setting. They came up with a distributed solution to this problem that was guaranteed to converge to an approximate solution (the Nash equilibrium of the game). The drawback is that their model is simplistic – the sensors, their sensing radii, and their cost of power are assumed to be fixed, only two operational sensing modes – 'on' and 'off' are available, every part of the sensing area is assumed to be equally important, the sensing attenuation with distance from the sensor is also ignored. In this model, a coverage function is defined, and a global objective function that trades off consumed power and coverage is constructed. The authors then formulate this problem as a game where the sensors are the players, the action set of each sensor is $\{\mathtt{on}, \mathtt{off}\}$, the social welfare function is the global objective function itself. What they indirectly do from this point onwards, is to employ the WLU design so that the optimal solution is always a Nash equilibrium. The authors then invoke the JSFP learning algorithm on this game so as to converge to a Nash equilibrium of the game in a purely distributed manner. They prove that full coverage is guaranteed in any Nash equilibrium of the game; however, no theoretical bound on the price of anarchy is provided.

4.2 A generalized model

Now, we present our model for the sensor network. It is much more general and powerful than the one in [9] in several key aspects:

(a) We relax the assumption that the sensors are fixed – each sensor can choose its position in a dynamic manner.

(b) We allow for different sensing radii – each sensor can choose its sensing radius as it sees fit.

(c) We permit any discrete number of operating levels for sensing (which also correspond to the available power levels).

(d) We explicitly incorporate the sensing attenuation with distance from the sensors.

(e) We allow for the sensing area to have parts of varying importance.

(f) We admit a family of cost functions for the power consumption.

Consider a set S of n sensors to be deployed over a fine grid specified by R, the set of tiny squares that make up the grid. The sensors constitute the players of the game. Each sensor i can choose the sector r_i at which to center itself, its sensing radius x_i (measured in number of sectors), and the power level p_i at which it performs the sensing. Clearly, p_i directly affects the probability of sensing an event by sensor i. To avoid more notations, we let p_i denote the peak sensing probability (the detection probability of the sensor i in the sector where it is centered, r_i). Hence, the action set

$\mathcal{A}_i$ for sensor i just contains all feasible tuples of the form (r_i, x_i, p_i). Let d_i^r denote the distance of sector r from sensor i, measured in number of sectors (also called the distance of sector r from the sector r_i). We allow the detection probability of a sensor to vary within its sensing area (this models the sensing attenuation) – let $p_i^r = f(p_i, d_i^r)$ denote the detection probability of sensor i at sector r. Let $c(x_i, p_i)$ denote the cost to sensor i of operating at power level p_i and sensing radius x_i. Finally, let $V(r)$ denote the importance of detecting an event in sector r. Given all these notations, for an arbitrary action profile a, the probability of detecting an event at sector r can be written as

$$P(r, a) = 1 - \prod_i (1 - p_i^r).$$

Then the social welfare garnered from sector r can be written as

$$W_r(a) = P(r, a)V(r),$$

and the total social welfare to be optimized is given by

$$W(a) = \sum_{r \in R} W_r(a) - \sum_{i \in S} c(x_i, p_i).$$

It must be noted that the state space is effectively two dimensional, because, once a sensor i chooses r_i and x_i, the optimum value of p_i is simply obtained by solving the optimization problem in a single variable (maximizing U_i).

4.3 Applying game-theoretic control

If we view this problem in the potential-games-based framework, our solution would simply involve picking a utility design and a learning rule that work in this framework. First, let us analyze the WLU design for assigning utilities for the individual sensors, which yields:

$$U_i(a) = \sum_{r \in a_i} f_r(i, a) - c(x_i, p_i),$$

where '$r \in a_i$' means that sector r is covered by the choice a_i, and

$$f_r(i, a) = W_r(a) - W_r(a_i^0, a_{-i}) = p_i^r \cdot V(r) \cdot \prod_{j \neq i} (1 - p_j^r).$$

Since WLU design guarantees a potential game, we can invoke any learning rule of our choice (depending on the desired performance) that works with potential games. For example, JSFP would result in quick convergence to a Nash equilibrium, whereas log-linear learning would result in slower convergence, but guarantee convergence to the best Nash equilibrium, which in the case of WLU design, would be the global social welfare maximizer. Now, the only question remains as to whether these utilities are calculable with local information (this is required by the learning algorithm). Since it is reasonable to assume that the communication radius is typically larger than the sensing radius and communication takes negligible power compared to sensing, if two sensors overlap in their sensing area, they can definitely communicate. From the expression for $f_r(i, a)$, it is clear that a sensor i only needs information from other sensors that cover its sensing area to compute its utility. Also, one can verify that the social welfare function is submodular for realistic functions f and c (for example, radially decaying f and increasing, convex c).

While we could have used SVU or WSVU designs instead of WLU, these designs, apart from their computational hardness, place impractical informational requirements on the agents for them to be able to compute their utilities, and as a result, any learning algorithm would not be realistically implementable.

So we have essentially proposed a solution (at a high level) for the power control problem in a very general model. The key observation here is that we could do this very easily because of the change in perspective towards game-theoretic control that is provided by our architectural view, and the modularization that it entails.

Chapter 5

Relationships to Other Approaches

In this section, we highlight some interesting connections between layering via potential games and other distributed system design tools. We show formally that two other tools (distributed constraint optimization and Gibbs-sampler-based control) can be viewed as instances of layering via potential games (though they do not fall into our simple illustrative model of resource allocation games). This relationship highlights that in retrospect one could have used the layered architecture presented in this thesis to design Gibbs-sampler based control modularly via 'off the shelf' utility and learning designs. Further, it highlights that other utility and learning designs can easily be swapped into these designs.

In addition, we want to highlight that there is a strong relationship between potential-games-based control and Lyapunov-based control. Specifically, it is possible to use a given Lyapunov function as the potential function of a game. Then, either WLU or SVU can be applied in combination with any of the learning rules described above to develop a game-theoretic design. Often, Lyaponov-based control (e.g., [47]) can be viewed as using WLU in combination with gradient play learning dynamics.

5.1 Gibbs-sampler-based control

Gibbs-sampler-based control [28, 40] is a popular physics-inspired approach for wireless protocol design. To introduce Gibbs-sampler based control we adapt the description of [28]. We represent a distributed system by an undirected graph $G(N, E)$ with $|N| = n$. Each node stores a state variable from a finite state space S. The state of the graph is $s = (s_1, \ldots, s_n)$. An energy function $\mathcal{E} : S^n \to \mathbb{R}$ represents the global cost of the system as a function of its state. The objective is to find a state of minimum energy. The Gibbs sampler approach provides an efficient solution for this problem, if $\mathcal{E}(s)$ is of the form

$$\mathcal{E}(s) = \sum_k \sum_{M \in C_k} V(M)$$

where C_k is the set of all cliques of order k, and $V : 2^N \to \mathbb{R}_+$ is defined such that $V(M)$ depends only on the states of nodes in M, and is zero if M is not a clique. We say that $\mathcal{E}$ derives from V (or conversely, V generates $\mathcal{E}$). We then define the *local energy* of a node $i \in N$ to be the sum of those terms in $\mathcal{E}(s)$ that involve s_i,

$$\mathcal{E}_i(s_i, (s_j)_{j \neq i}) = \sum_k \sum_{M \in C_k : i \in M} V(M)$$

The Gibbs measure associated with an energy function $\mathcal{E}$ and temperature $T > 0$ is defined as the following probability distribution on the states of the graph,

$$\pi(s) = e^{-\frac{\mathcal{E}(s)}{T}} / \left(\sum_{s' \in S^n} e^{-\frac{\mathcal{E}(s')}{T}} \right) \tag{5.1}$$

It is easy to observe that this distribution favors low-energy states when T is small. Moreover, it is a Markov random field – given the states of all its neighbors, the state of a node n is independent of all non-neighbor nodes $i \neq n$.

The *Gibbs sampler* is an iterative procedure where during each step, each node i, given the states of all other nodes, samples its new state from the following distribution on S,

$$\mu(s_i) = e^{-\frac{\mathcal{E}_i(s_i, (s_j)_{j \neq i})}{T}} / \left(\sum_{s'_i \in S} e^{-\frac{\mathcal{E}_i(s'_i, (s_j)_{j \neq i})}{T}} \right) \ , \ s_i \in S$$

When T is fixed, the Gibbs sampler converges to a steady state that is distributed according to (5.1).

Finally, for convergence to the global minimum of the energy function, we use the 'annealed' Gibbs sampler, which adds a small decrease of T to this algorithm at every step. When this decrease with time $t > 0$ is proportional to $1/\log(1+t)$, the system converges to a set of states of minimal global energy.

The above description of Gibbs-sampler-based control already highlights the connection with potential games – it is equivalent to using WLU in combination with log-linear learning. It is immediate to see that the Gibbs sampler is the same as the log-linear learning algorithm described earlier. To show that the utility design is equivalent to WLU, we construct a game as follows. Consider N to be the set of players, the common state space to be their action spaces, and the negative of the global energy function to be the social welfare function. Denote by $C_k(H)$, the set of cliques of order k in graph H. The WLU design is:

$$\begin{aligned} U_i(s) &= W(s) - W(\emptyset, s_{-i}) \\ &= -\mathcal{E}(s) + \mathcal{E}(\emptyset, s_{-i}) \\ &= -\sum_k \sum_{M \in C_k(G)} V(M) + \sum_k \sum_{M \in C_k(G-\{i\})} V(M) \end{aligned}$$

Since $M \in C_k(G - \{i\}) \iff M \in C_k(G)$ and $i \notin M$, all these terms cancel out, leaving only the terms in $-\mathcal{E}(s)$ for which $i \in M$. Hence,

$$U_i(s) = -\sum_k \sum_{M \in C_k : i \in M} V(M) = -\mathcal{E}_i(s)$$

5.2 Distributed constraint optimization

A constraint optimization problem is specified by a set of variables $N = \{1, \ldots, n\}$, each of which takes a value s_i from a finite state space [1] S, a set of constraints $C = \{c_1, c_2, \ldots, c_m\}$, and a global objective function $W : S^n \to \mathbb{R}$, that encodes the relative desirability of each possible state s of the system in S^n. A constraint $c = \langle N_c, R_c \rangle$ is specified by the set of variables $N_c \subseteq N$ over which it is defined, and a relation $R_c \subset S^{|N_c|}$ between those variables. Let $C(M)$ denote the set of constraints involving any of the variables in the set $M \subseteq N$. A function $U_c\left(s_c\right)$ specifies the reward for satisfying constraint c, where s_c is the configuration of the states of the variables in N_c. The global objective function is typically written as $W(s) = \sum_{c \in C} U_c(s_c)$. The problem is to find a global maximizer of W. Given this, a 'distributed' constraint optimization problem (DCOP) is produced when a set of autonomous agents each independently control the state of a variable.

In [11], the authors formulate a 'DCOP game' by assigning each agent a private utility function $U_i(s)$, which is dependent on both its own state and the state of the other agents in the system. They choose these utilities to be the local effect of the agent on the global objective function, i.e., $U_i(s) = \sum_{c \in C(\{i\})} U_c(s_c)$. When utilities are assigned thus, the authors show that the DCOP game admits a potential function which is the same as the global objective function, thus allowing any of the learning designs described earlier to be applied. Here, we highlight that this DCOP game corresponds specifically to choosing a WLU design. Consider the autonomous agents of the DCOP to be the players, the common state space to be their action sets, and the global objective function

[1] We assume that all variables have the same state space for simplicity. The discussion can easily be generalized to the case of asymmetric state spaces.

to be the social welfare function. WLU then gives:

$$\begin{aligned} U_i(s) &= W(s) - W(\emptyset, s_{-i}) \\ &= \sum_{c \in C(N)} U_c(s_c) - \sum_{c \in C(N \setminus \{i\})} U_c(s_c) \\ &= \sum_{c \in C(\{i\})} U_c(s_c) \end{aligned}$$

which is exactly the utility function suggested for the DCOP game in [11]. The last step follows by observing that when variable i is not part of the DCOP, all constraints not containing i in the original DCOP are not affected.

Chapter 6

Conclusion

In this thesis, we have studied game-theoretic control, a promising new approach for distributed resource allocation. We have shown how game-theoretic control, which consists of designing utility functions and learning rules, can be viewed as having a layered hourglass architecture, where we separate the task of utility design from that of learning design by enforcing a constraint on the class of allowable games. We then illustrated this architectural view using potential games as a choice for the virtualization layer, highlighting available choices for utility and learning designs in this framework. Following this, we demonstrated the power of our approach by proposing a practical solution to the power control problem in sensor networks in a very general model. Finally, we showed that two other popular approaches, namely distributed constraint optimization and Gibbs-sampler-based control, can be viewed as instances of game-theoretic control when viewed in the context of potential-games-based architecture.

6.1 Future directions

Throughout this thesis, we have discussed only one concrete example of our game-theoretic control architecture – using potential games as an interface between utility and learning design. However, potential games are only one, very restrictive class of games, and current research has begun to uncover limitations of layering via potential games (see Section 3.4).

Thus, as we move forward, it is important to consider other options for the interface. Recent research is beginning to consider a variety of other classes of games as the basis for game-theoretic control. For example, [39] suggests 'state-based potential games', which are a limited form of Markov games, as a way to overcome the limitation of potential games described in Section 3.4. Other examples include [57], which proposes using conjectural equilibria in the context of multi-user power control, and [1] which proposes using oblivious equilibria in the context of large stochastic games. However, as yet, there is little understanding of the strengths and limitations of designs using these new classes of games. For example, which classes of games provide modularity when used as an interface? What is gained by broadening the interface from potential games to other classes? Is there a penalty for broadening the interface, e.g. slower convergence rates for learning?

Our hope is that the identification of an architectural view of game-theoretic control in this thesis can help formalize and motivate these important directions for the field. We propose that a better understanding of the strengths and weaknesses of differing interfaces will provide useful insight into how to choose the appropriate interface for a given class of applications. For example, for some applications, the limitation described in Section 3.4 may not be important, while for others, the limitation may cause design outside of potential games to be preferable.

Bibliography

[1] S. Adlakha, R. Johari, G. Weintraub, and A. Goldsmith. Oblivious equilibrium: an approximation to large population dynamic games with concave utility. In *Game Theory for Networks*, pages 68–69, 2009.

[2] A. A. Ageev and M. I. Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Combinatorial Optimization*, 8(3):307–328, 2004.

[3] R. K. Ahuja, A. Kumar, K. C. Jha, and J. B. Orlin. Exact and heuristic algorithms for the weapon-target assignment problem. *Operations Research*, 55(6):1136–1146, 2007.

[4] T. Alpcan, L. Pavel, and N. Stefanovic. A control theoretic approach to noncooperative game design. In *IEEE CDC*, pages 8575–8580, 2009.

[5] E. Altman and Z. Altman. S-modular games and power control in wireless networks. *Transactions on Automatic Control*, 48(5):839–842, 2003.

[6] E. Altman, T. Boulogne, R. El-Azouzi, T. Jiménez, and L. Wynter. A survey on networking games in telecommunications. *Computers and Operations Research*, 33(2):286–311, 2006.

[7] E. Altman, R. El-Azouzi, and T. Jiménez. Slotted ALOHA as a game with partial information. *Computer Networks*, 45(6):701–713, 2004.

[8] L. E. Blume. The statistical mechanics of strategic interaction. *Games and Economic Behavior*, 5(3):387–424, 1993.

[9] E. Campos-Náñez, E. Garcia, and C. Li. A game-theoretic approach to efficient power management in sensor networks. *Operations Research*, 56(3):552–561, 2008.

[10] C. G. Cassandras and W. Li. Sensor networks and cooperative control. *European Journal of Control*, 11(4-5):436–463, 2005.

[11] A. Chapman, A. Rogers, and N. R. Jennings. Benchmarking hybrid algorithms for distributed constraint optimization games. In *ACM OptMAS*, pages 1–11, 2008.

[12] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *Wireless Networks*, 8(5):481–494, 2002.

[13] H-L. Chen, T. Roughgarden, and G. Valiant. Designing networks with good equilibria. In *ACM-SIAM SODA*, pages 854–863, 2008.

[14] V. Conitzer and T. Sandholm. Computing shapley values, manipulating value division schemes, and checking core membership in multi-issue domains. In *AAAI*, pages 219–225, 2004.

[15] M. Csete and J. C. Doyle. Bow ties, metabolism and disease. *Trends in Biotechnology"*, 22(9):446–450, 2004.

[16] P. Dubey, O. Haimanko, and A. Zapechelnyuk. Strategic complements and substitutes, and potential games. *Games and Economic Behavior*, 54(1):77–94, 2006.

[17] Y. M. Ermoliev and S. D. Fløam. Learning in potential games. Technical report, International Institute for Applied Systems Analysis, 1997.

[18] D. Falomari, N. Mandayam, D. Goodman, and V. Shah. A new framework for power control in wireless data networks: games, utility, and pricing. In *Wireless Multimedia Network Technologies*, pages 289–310. Kluwer, 1999.

[19] U. Feige and J. Vondrak. Approximation algorithms for allocation problems: Improving the factor of 1 - 1/e. In *IEEE FOCS*, pages 667–676, 2006.

[20] D. Fudenberg and D. Levine. *The theory of learning in games*. MIT Press, 1998.

[21] Scutari. G., D. P. Palomar, and J. Pang. Flexible design of cognitive radio wireless systems: From game theory to variational inequality theory. *IEEE Signal Processing Magazine*, 26(5):107–123, September 2009.

[22] D. Ganesan, A. Cerpa, W. Ye, Y. Yu, J. Zhao, and D. Estrin. Networking issues in wireless sensor networks. *Parallel and Distributed Computing*, 64(7):799–814, 2004.

[23] A. Garcia, D. Reaume, and R. L. Smith. Fictitious play for finding system optimal routings in dynamic traffic networks. *Transportation Research Part B: Methodological*, 34(2):147–156, 2000.

[24] B. Godfrey and D. Ratajczak. Naps: Scalable, robust topology management in wireless ad hoc networks. In *Information Processing in Sensor Networks*, pages 443–451, 2004.

[25] M. X. Goemans, L. (E.) Li, V. S. Mirrokni, and M. Thottan. Market sharing games applied to content distribution in ad hoc networks. *Selected Areas in Communications*, 24(5):1020–1033, 2006.

[26] G. Haeringer. A new weight scheme for the shapley value. *Mathematical Social Sciences*, 52(1):88 – 98, 2006.

[27] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh. Energy-efficient surveillance system using wireless sensor networks. In *MobiSys*, pages 270–283, 2004.

[28] B. Kauffmann, F. Baccelli, A. Chaintreau, V. Mhatre, K. Papagiannaki, and C. Diot. Measurement-based self organization of interfering 802.11 wireless access networks. In *IEEE INFOCOM*, pages 1451–1459, 2007.

[29] D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *ACM KDD*, pages 137–146, 2003.

[30] R. S. Komali and A. B. MacKenzie. Distributed topology control in ad-hoc networks: a game theoretic perspective. In *IEEE CCNC*, pages 563–568, 2006.

[31] J. F. Kurose and K. W. Ross. *Computer networking: A top-down approach*. Addison-Wesley, 2009.

[32] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. S. Glance. Cost-effective outbreak detection in networks. In *ACM KDD*, pages 420–429, 2007.

[33] L. (E.) Li, J. Y. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer. A cone-based distributed topology-control algorithm for wireless multi-hop networks. *Transactions on Networking*, 13(1):147–159, 2005.

[34] J. R. Marden, G. Arslan, and J. S. Shamma. Regret based dynamics: convergence in weakly acyclic games. In *ACM AAMAS*, pages 1–8, 2007.

[35] J. R. Marden, G. Arslan, and J. S. Shamma. Joint strategy fictitious play with inertia for potential games. *Transactions on Automatic Control*, 54(2):208–220, Feb 2009.

[36] J. R. Marden and M. Effros. The price of selfishness in network coding. In *NetCod*, pages 18–23, 2009.

[37] J. R. Marden and J. S. Shamma. Revisiting log-linear learning: Asynchrony, completeness and a payoff-based implementation. *Under submission*.

[38] J. R. Marden and A. Wierman. Distributed welfare games. *Under submission*.

[39] J. R. Marden and A. Wierman. Overcoming limitations of game-theoretic distributed control. In *IEEE CDC*, pages 6466–6471, 2009.

[40] V. Mhatre, K. Papagiannaki, and F. Baccelli. Interference mitigation through power control in high density 802.11 wlans. In *IEEE INFOCOM*, pages 535–543, 2007.

[41] A. Mishra, V. Shrivastava, D. Agrawal, S. Banerjee, and S. Ganguly. Distributed channel management in uncoordinated wireless environments. In *MOBICOM*, pages 170–181, 2006.

[42] D. Monderer and L. S. Shapley. Fictitious play property for games with identical interests. *Economic Theory*, 68(1):258–265, 1996.

[43] R. A. Murphey. Target-based weapon target assignment problems. In *Nonlinear assignment problems: Algorithms and applications*, pages 39–53. Kluwer, 2000.

[44] M. Narasimhan, N. Jojic, and J. Bilmes. Q-clustering. In *Advances in Neural Information Processing Systems 18*, pages 979–986. MIT Press, 2006.

[45] Nat.Research Council Comm. on the Internet in the Evolving Info. Infrastructure. The Internet's coming of age, 2001.

[46] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic game theory*. Cambridge University Press, 2007.

[47] A. Papachristodoulou. In *IEEE CDC*, volume 1, pages 1029–1034, 17-17 2004.

[48] P. Parag, S. Shakkottai, and J-F. Chamberland. Value-aware resource allocation for service guarantees in networks. In *IEEE INFOCOM*, 2010.

[49] L. Pavel. An extension of duality to a game-theoretic framework. *Automatica*, 43:226–237, 2007.

[50] A. Rantzer. Distributed control using decompositions and games. In *IEEE CDC*, pages 13–13, 2008.

[51] T. Roughgarden. *Selfish routing and the price of anarchy*. MIT Press, 2005.

[52] W. H. Sandholm. Potential games with continuous player sets. *Journal of Economic Theory*, 97(1):81–108, 2001.

[53] D. Shah and J. Shin. Dynamics in congestion games. In *ACM SIGMETRICS/Performance*, 2010.

[54] J. S. Shamma and G. Arslan. Dynamic fictitious play, dynamic gradient play, and distributed convergence to Nash equilibria. *Transactions on Automatic Control*, 50(3):312–327, 2005.

[55] L. S. Shapley. A value for n-person games. In *Contributions to the theory of games – II*. Princeton University Press, 1953.

[56] A. Singh, A. Krause, C. Guestrin, W. J. Kaiser, and M. A. Batalin. Efficient planning of informative paths for multiple robots. In *IJCAI*, pages 2204–2211, 2007.

[57] Y. Su and M. van der Schaar. Conjectural equilibrium in multiuser power control games. *Transactions on Signal Processing*, 57(9):3638–3650, 2009.

[58] Y. Su and M. van der Schaar. A new perspective on multi-user power control games in interference channels. *Transactions on Wireless Communications*, 8(6):2910–2919, 2009.

[59] T. Ui. Shapley value representation of potential games. *Games and Economic Behavior*, 31(1):121–135, 2000.

[60] E. G. Villegas, R. V. Ferré, and J. Josep Paradells. Frequency assignments in IEEE 802.11 WLANs with efficient spectrum sharing. *Wireless Communications and Mobile Computing*, 9(8):1125–1140, 2008.

[61] W. Willinger and J. C. Doyle. Robustness and the internet: design and evolution. In *Robust design: a repertoire of biological, ecological, and engineering case studies*. Oxford University Press, 2005.

[62] D. H. Wolpert and K. Tumer. An introduction to collective intelligence. In *Handbook of Agent technology. AAAI*, 1999.

[63] G. Xing, C. Lu, Y. Zhang, Q. Huang, and R. Pless. Minimum power configuration for wireless communication in sensor networks. *Transactions on Sensor Networks*, 3(2):11, 2007.

[64] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. D. Gill. Integrated coverage and connectivity configuration for energy conservation in sensor networks. *Transactions on Sensor Networks*, 1(1):36–72, 2005.

[65] T. Yan, T. He, and J. A. Stankovic. Differentiated surveillance for sensor networks. In *SenSys*, pages 51–62, 2003.

[66] H. P. Young. *Strategic learning and its limits*. Oxford University Press, 2005.