Incentivizing Local Behavior in Distributed Systems

Jason R. Marden
Department of Electrical, Computer, and Energy Engineering
University of Colorado
jason.marden@colorado.edu

Abstract—Game theory is a well-established discipline in the social sciences that is primarily used for modeling social behavior. Traditionally, the preferences of the individual agents are modeled as utility functions and the resulting behavior is assumed to be an equilibrium concept associated with these modeled utility functions, e.g., Nash equilibrium. This is in stark contrast to the role of game theory in engineering systems where the goal is to design both the agents’ utility functions and an adaptation rule such that the resulting global behavior is desirable. The transition of game theory from a modeling tool for social systems to a design tool for engineering systems promotes several new research directions that we will discuss in this talk. In particular, this talk will focus on the question of how to design admissible agent utility functions such that the resulting game possesses desirable properties, e.g., the existence and efficiency of pure Nash equilibria. Our motivation for considering pure Nash equilibria stems from the fact that adaptation rules can frequently be derived which guarantee that the collective behavior will converge to such pure Nash equilibria.

I. OVERVIEW

This tutorial talk will survey results pertaining to utility design for distributed engineered systems. While utility design for engineered systems is a relatively new research area, utility design for societal systems has been studied extensively in the game theoretic literature, e.g., cost sharing problems [1], [2], [3], [4] and mechanism design [5]. Much like the engineering agenda, the underlying goal for utility design in societal systems is to augment players’ utility functions in an admissible fashion to induce desirable outcomes. However, the difference between the constraints and objectives in social and engineering systems requires looking at this literature from a new perspective.

It is worth contrasting utility design for distributed engineering systems with the well-developed field of economic mechanism design (e.g., [6]), which shares the goal of designing games in which self-interested behaviour leads to a desirable outcome. First, the primary challenge of economic mechanism design — the incentive-compatible elicitation of private preferences — is not relevant for utility design. For example, in a single-item auction, the willingness to pay of each bidder is a priori unknown, and the point of an auction is to determine who is willing to pay the most. By contrast, the point of utility design in engineering systems is to define the preferences of agents (i.e., programmable components) so that “self-interested” behavior in the resulting game leads to a good outcome. Second, the primary challenge of utility design for decentralized systems — the lack of centralized control — is not relevant in traditional mechanism design. For example, in mechanism design it is usually assumed that the designer can centrally allocate resources and select the most favorable of many equilibria. These two fundamental differences render most techniques of mechanism design unsuitable for the decentralized control problems studied here.

In spite of the differences between the constraints and objective in social and engineering systems, several results from the cost sharing literature have immediate applicability to distributed engineering systems [7], [8]. This tutorial will survey such results and highlight open questions in this area. Much of the material is drawn from the forthcoming article:


REFERENCES