A Computational Approach to Organizations and Organizing

Michael J. Prietula, Kathleen M. Carley, and Les Gasser

Organizations and groups permeate our lives, and their influence is growing. Indeed, each of us is so familiar with being in or being affected by organizations that we may tend to think we know how they work—as Giddens has pointed out, people must be in some sense fairly good social theorists just to get along in daily life (Giddens 1984). However, naive theories about organizations are often wrong, and it can be dangerous to rely upon them. Organizations are large complex highly volatile systems whose behavior affects and is affected by the environment in which they operate. Individuals can, but do not always, affect how organizations operate. The character of information technology that is available can affect organizational processes, and so on. For effective guidance in organization design, management, and operation, stronger principles, and the methods for generating and deriving them are needed.

Clearly, organizational behavior is affected by a large number of interacting factors. Organizational theory can be characterized as the study of how this multiplicity of factors combine to influence the behavior of organizations and the people and technologies comprising them. Some would even argue that across such factors there are general principles of organizing that are true for all groups and organizations regardless of whether the actors within the organizations are human or artificial. As such, organizational theory can also be characterized as the search for these general principles.

Computational organization theory (COT) is the study of organizations as computational entities. COT researchers view organizations as inherently computational because they are complex information processing systems. An organization as a computational system is composed of multiple distributed "agents" that exhibit organizational properties (such as the need to act collectively and struggles for power), are assigned tasks, technology, and resources, and across which knowledge, skills, and communicative capabilities are distributed. Computational organization theory focuses on understanding the general factors and nonlinear dynamics that affect individual and organizational behavior with a special attention on decision making, learning, and adaptation (Carley 1995). In computational organization models, information, personnel, decision responsibility, tasks, resources, and opportunity are distributed geographically, temporally, or structurally within, and sometimes between, organizations.

Computational organization theories are meso-level theories of organizations. An idealized characterization of organizational studies partitions research into two types. A macro perspective, (conventional organization theory) treats the complexities of individual behavior as largely irrelevant, or simplifies variety across individuals into an ideal individual type. A micro (i.e., organizational behavior) perspective focuses on the individual and often minimizes the constraints on action afforded by tasks and social-situations, and institutions. Computational organization theories are typically meso-level in the sense that they seek to explain and predict macro-level behavior, such as overall organizational performance, from micro-level actions, such as the interaction among agents, each of which are "cognitively" limited.

Computational analysis can help us to grasp some fundamentals of human information processing behavior (Simon 1973). Thus, computational modeling can be valuable for the study of organizations as collections of intelligent agents. Further, compared with experiments using human subjects, computational models are generally less noisy, easier to control, more flexible, more objective, and can be used to examine a larger variety of factors within less time. Computational analysis also makes it possible to determine whether or not important nonlinearities in behavior emerge as scope conditions are extended. For example, computational models may be larger (e.g., more agents) or may cover a longer period (more tasks) than can be covered in a human laboratory experiment. Ostrom (1988) argues that computer simulation offers a third symbol system in studying social science (with natural language and mathematics being the first two symbol systems) and notes that "computer simulation offers a substantial advantage to social psychologists attempting to develop theories of complex and interdependent social phenomena." This same advantage is true for organizational theorists, perhaps even more so given the nature of organizations.

Organizations are complex, dynamic, nonlinear adaptive and evolving systems. Organizational behavior results from interactions among a variety of adaptive agents (both human and artificial), emergent structuration in response to non-linear processes, and detailed interactions among a large number of other factors. As such, organizations, or at least many aspects of them, are poor candidates for analytical models. Thus, computational analysis becomes an invaluable tool for theory building as it enables the researcher to generate a set of theoretical propositions from basic principles even when there are complex interactions among the relevant factors. The computational model can be thought of as a hypothesis generator that generates a set of propositions that can be more easily assured precise and internally consistent. In addition, computational models allow researchers to demonstrate proof of concept; i.e., to demonstrate whether or not a set of factors that are completely modelable are capable by themselves of generating certain phenomena. Used in this way, computational models can be used to show the potential legitimacy of various theoretical claims in organization science. Finally, using computational models it is often possible to determine the logical consistency of a set of propositions and the extent to which theoretical conclusions actually follow from the underlying assumptions. In some areas, formal logic plays this role. However, currently, multi-agent logics are not sufficiently developed to play this role.

Computational techniques for theorizing about organizations and organizing are invaluable tools for both the researcher and the manager. Most of the work in this book focuses on the scientific rather than the administrative side of this enterprise. Nevertheless, some hints (Chapters 1, 7, 8 and 10) are given as to how these tools might ultimately be of use to managers. Essentially, these computational models, once appropriately validated, can be used as decision aids to help the manager think through the impacts of new information technologies, organizational redesigns, or the reengineering of tasks. Clearly the models are not quite there yet, but that is one of the ultimate objectives.

Given such a computational model, three general evaluation criteria can be brought to bear: sufficiency testing, process testing, and component analysis. Sufficiency testing is the weakest form of validation and focuses entirely on the outcome of the behavior. In essence, it states that a computational model should at least be able to produce the behavior it purports to explain and is similar to the achievement criterion in cognitive modeling (Simon and Baylor 1966). Chapters 1, 2, 3, 4, 5, 6, 9 and 10 meet this criteria. Process testing makes a stronger statement, and is similar to the process criterion in cognitive modeling (Simon and Baylor 1966). This test goes beyond showing that an unspecified set of mechanisms produces a certain result by demonstrating that particular mechanisms (or knowledge) can produce the behavior. In this test, comparisons are made at some level of abstraction between the model and a referent (a proposed gold standard). Chapters 1, 2, 3, 4 and 9 are movements in this direction. Component analysis examines specific contributions of the mechanisms or knowledge represented in the reasoning events. The relative impact of different components are contrasted. Chapters 1, 3, 4 and 9 have this characteristic. Computational theorists can build theories of organizations and organizing by hypothesizing about the behavior of their models, testing these hypotheses through a series of virtual experiments, generating a new consistent set of hypotheses derived from these computational results, and then testing these hypotheses with "real" data. The chapters that move farthest in this direction are Chapters 4 and 9.

Hypothesizing about the behavior of computational models may seem trivial and obvious, as the program components are well defined; however, the complexities of today's simulation systems inhibit accurate a priori specifications of their behavior. Even with simple models, such as Team-Soar (Chapter 3), the model of cooperation (Chapter 5), and the original garbage can model (Cohen, March, and Olsen 1972), new findings emerged that were not hypothesized given the program components and new insights were gained into organizational performance. Though predictions may be made concerning the behavior of such models, tests of actual program performance must be made to verify them. To this end, researchers often use these models to run virtual experiments, collecting data that is then analyzed graphically and statistically.

Within the field of computational organization theory, computational analysis is used to develop a better understanding of the fundamental principles of organizing multiple information processing agents and the nature of organizations as computational entities. Research in this area has two main foci. The first foci is has to do with building new concepts, theories, and knowledge about organizing and organization. Most of the chapters have this foci. The second foci has to do with developing tools and procedures for the validation. Chapters 7, 8 and 11 have this foci.

This book is divided into four sections. Each of these sections corresponds to a major area in which there is on-going research. Section One is titled "Organizations as Multi-Agent Systems." Human (or human-derivative) organizations, like games, are "artificial" in the sense that they are crafted by humans (Simon 1981). Organizational behavior emerges both from the artificial construct that constrains individual interactions and the natural limits on human (or agent) behavior. This type of emergent behavior is seen in the results presented in Section One. All three chapters in this section draw heavily on work in artificial intelligence. Unlike most games, however, organizations are highly volatile with no specifiable (or perhaps predictable) equilibria. Indeed, within organizations it is the norm that the rules change, the players change, and the situations change. This volatility is due in large part to the agents which comprise them and the way in which the agents interact. Hence, within organizations, the form of the rules and procedures depends on the agents and their personal history as they respond to the changing environment. The first chapter (Chapter 1, "Web-Bots, Trust, and Organizational Science"), by Carley and Prietula, describes an experiment in which a strong model of artificial intelligence called Soar (Laird, Newell, and Rosenbloom 1987) is used to explore the significant issue of trust between intelligent agents. Not only is this a unique direction of inquiry, but the study produces quantifiable data on deliberation and communication derived directly from the theoretical stance articulated in code. In the second chapter of this section (Chapter 2, "Team-Soar: A Model for Team Decision Making"), Kang, Waisel, and Wallace also incorporate a Soar approach. In this chapter, the authors model a naval command and control team tasked with making critical decisions regarding the hostility of an incoming aircraft. Events such as those occurring with the U.S.S. Stark and the U.S.S. Vincennes have been traced to dysfunctional team behaviors. Their approach is to simulate and analyze possible sources of team dysfunctionality to improve team decision making behaviors. In the third chapter in Section One (Chapter 3, "Designing Organizations for Computational Agents"), So and Durfee describe a framework for understanding organizational design design for computational agents, use that framework for analyzing the expected performance of a class of organizations, and describe how the analyses can be applied to predict performance for a distributed information gathering task. An interesting component of this chapter is the concept of organizations re-designing themselves, addressing an emerging critical problem in network administration.

Section Two ("Organizations and External Conditions") explores the relationship between organizational action, agent behavior, and environmental volatility. Part of the volatility within organizations comes from the advent of new technologies. Further, organizations often try to employ technologies to curb the impact of other forms of organizational volatility. Lin (Chapter 4, "The Choice between Accuracy and Errors: A Contingency Analysis of External Conditions and Organizational Decision Making Performance") uses a version of the radar task described in Chapter 2, but focuses on exploring the relationship between organizational performance, organizational designs, and environmental properties. It appears that the reliability of an organization resides in the fit between the design and the task the choice of design becomes a strategic decision between what type of errors the organization is willing to accept or minimize. Huberman and Glance (Chapter 5, "Fluctuating Efforts and Sustainable Cooperation") show that when individuals confronted with a social dilemma contribute to the common good with an effort that fluctuates in time, they can generate an average utility to the group that decreases in time. This paradoxical behavior takes place in spite of the fact that typically individuals are found to be contributing at any one time. This phenomenon is the result of an intermittency effect, whereby unlikely bursts of defection determine the average behavior of the group. Thus, typical behavior of individuals comprising a group, can be inconsistent with a groups average properties. In the final chapter in this section (Chapter 6, "Task Environment Centered Simulation"), Decker describes the TÆMS framework (Task Analysis, Environment Modeling, and Simulation) to model and simulate complex, computationally intensive task environments as multiple levels of abstraction, and from multiple viewpoints. TÆMS is a tool for building and testing computation theories of coordination. This framework permits researchers to mathematically analyze (when possible) and quantitative simulate (when necessary) the behavior of multi-agent systems with respect to interesting characteristics of their task environment. As such, it is a testbed for exploring centralized, parallel, or distributed control algorithms, negotiation strategies, and organizational designs. To illustrate TÆMS, Decker investigates a simple question: Is there a difference between performance due to either the choice of organizational structure or the decomposibility of the technology?

The chapters in Section Three ("Organizations and Information Technology") address issues of technology, but within the realm of information technology and information systems. Most computational models of organizations do not consider the role of information technology, Thus, the chapters in Section Three represent initial forays into the how to think formally about the role of information technology in organizations. Fox, Barbuceanu and Lin (Chapter 7, "An Organizational Ontology for Enterprise Modeling") begin to address the next generation of Enterprise Model. Specifically, they propose that the next generation be a common sense enterprise model, which possess the capability to deduce answers to queries requiring relatively shallow knowledge of the domain. Thus, a key component of future information systems is an enterprise model that goes well-beyond the capabilities of current database or enterprise systems. Key to the articulation of such a model are the fundamental ontologies upon which the model is defined. The authors present a discussion of their approach to defining ontologies and ontological competence in their pursuit of the next generation enterprise model. In Chapter 8 ("Modeling, Simulating, and Enacting Complex Organizational Processes: A Life Cycle Approach"), Scacchi describes the approach and mechanisms to support the engineering of organizational processes throughout their life cycle. Organizations are, in part, defined by their processes. As events change (e.g., technology, tasks, environment) an organization may have to review and redefine its processes and process streams. Scacchi describes a knowledge-based computing environment, the articulator, that supports the defining and simulation of complex organizational processes. Kaplan and Carley (Chapter 9, "An Approach to Modeling Communication and Information Technology in Organizations") describe the communicating and information technology (COMIT) computational framework used to investigate information processing impacts of changing either the information technology or the communication structure on organizational performance. COMIT generates aggregate and detailed statistics on the number and duration of actions (e.g., communication, information lookup) and task completion quality. To illustrate, the authors describe a study in which levels of technology (high, low) are crossed with levels of experience and work structure (solo, collaborative). Their results suggest that technology, training, and organizational design can interact in complex ways to influence performance, and that computational approaches as COMIT can help reveal those complexities and their effects. In the final chapter of this section (Chapter 10, "Organizational Mnemonics: Exploring the Role of Information Technology in Collective Remembering and Forgetting"), Sandoe presents a conceptual model of organizational remembering and forgetting, and describes a simulation derived from the conceptual model. Sandoe argues that organizational remembering (and forgetting) occurs in three ways: an organization can remember (1) structurally, through the establishment of rules, roles, policies; (2) mutually, through advisory relationships among its members, and (3) technologically, through the creation of physical or symbolic artifacts. Sandoe then conducts a study where three organizational forms (hierarchy, network, hub) are simulated and tested with respect to environmental turbulence, turnover, and cost.

In the concluding essay ("Validating and Docking: An Overview, Summary and Challenge"), Burton addresses the chapter contributions in three contexts. First, Burton discusses the chapters with respect to the important issue of validity in the context of a framework which summarizes a computational model along three dimensions: its purpose, its process, and the analysis of its results (Burton and Obel 1996). Second, Burton categorizes the chapters according to a scheme developed by Carley (1995) which situates a computational simulation with respect to its explanatory role, of which four are proposed: organizational design, organizational learning, organizations and information technology, and organizational evolution and change. Finally, Burton speculates on the contribution of the collective in the context of "docking" or aligning simulation models for comparative purposes (Axtell, Axelrod, Epstein and Cohen 1996).

Computational organizational theorists are trying to use computational techniques to develop a firm scientific base for the study of organizations. As noted, organizations are often complex, nonlinear, adaptive systems. The natural complexity of organizations is reflected in the fact that many of the existing models and theories of organization are vague, intuitive, and under-specified. The more explicit and well-defined these theories, the greater our ability to make scientific progress. Computational theorizing about organizations helps to achieve this. The chapters in this book contribute to this endeavor. These chapters are the outgrowth of the tremendous outpouring of work in this area in the second half of the twentieth century since Cyert and March's *A Behavioral Theory of the Firm* in 1963. Recent work in this area combines traditional organizational concerns with performance, design, and adaptation with technique and approaches informed by work in the area of distributed artificial intelligence (Bond and Gasser 1988, Gasser and Huhns 1989). However, the computational organization theories of today, unlike much of the early work in distributed artificial intelligence work, are often grounded in existing cognitive, knowledge-based, information-processing theories of individual behavior and information processing, institutional, population ecology, or other models of organizations. Computational organization theorists extend the work on individual behavior to the organizational level (e.g., Simon 1947). This combination and extension gives precision to the notion of bounded rationality by specifying the nature of the boundaries and the role of social and historical information in defining organizational action (Carley and Newell 1994, Carley and Prietula 1994). This book contributes to our understanding of both organizations and organizing and provides illustrations for how to conduct research in this area.