7 Computational Organization Theory

Kathleen M. Carley and Les Gasser

7.1 Introduction

From the hospital, to the schoolroom, to the boardroom people find that the actions they take affect and are affected by various organizations, and the norms, procedures, culture, and members of those organizations. In order to navigate through an organizational world, agents (human and artificial) need social and organizational intelligence. This organizational intelligence comprises many dimensions, including communication capabilities, knowledge about who knows what, knowledge about norms, procedures, and culture of the organization, and more.

The ability of an organization to act is certainly dependent on the intelligence of the agents within it. However, organizations, and multiagent systems in general, often show an intelligence and a set of capabilities that are distinct from the intelligence and capabilities of the agents within them. It is not difficult to find multiagent systems that display non-random and repeated patterns and processes of action, communication, knowledge, and memory (beyond the lifetime of a single agent) regardless of whether or not the agents are human. Said another way, many multiagent systems exhibit characteristics of organization, and sometimes of intentional organization design. Organization designs may emerge spontaneously or be imposed, and they can can structure activities and attention within a system or control the actions of a system as a corporate entity.

From country to country, culture to culture, task to task, and agent type to agent type, we find both differences and commonalties in the patterns and processes connecting individual agents and in the forms organizations take. In order to navigate through environments and achieve results not achievable by individual agents, or to exhibit capabilities not held by individual agents, organizations (and indeed all multiagent systems) need to act as intelligent information processors, capable of responding as a single corporate entity, and to coordinate individual agents using organizing principles or designs. Research in the computational organization area employs computational techniques to theorize about and analyze organizations and the processes of organizing.

The goal of this chapter is to describe what can be done and what others have done in this area: the underlying principles, assumptions, concerns, and the major streams of work. After reading this chapter you will have gained insight into the aims, findings and new possibilities of this field. Further, after reading this chapter you should have developed a preliminary understanding of the nature of computational organizational models and developed some of your own ideas about how to construct virtual experiments using such models.

7.1.1 What Is an Organization?

A classic response to the question "What is an organization?" is "I know it when I see it." Indeed, every text book in organizational theory provides a definition of organizations. Unfortunately, there is no wide consensus on the definition of "organization," and indeed as theorists reason about organizations trying to answer fundamentally different questions, they construct different definitions of the basic phenomenon. While there is no single definition of organizations that is uniformly agreed to, there are general tenets that are more or less shared. In general, organizations are characterized as:

- large-scale problem solving technologies
- comprised of multiple agents (human, artificial, or both)
- engaged in one or more tasks; organizations are systems of activity
- goal directed (however, goals can change, may not be articulable, and may not be shared by all organizational members)
- able to affect and be affected by their environment
- having knowledge, culture, memories, history, and capabilities distinct from any single agent
- having legal standing distinct from that of individual agents

One rationale for the existence of organizations qua organizations is that they exist to overcome the limitations of individual agency.[1] From this viewpoint, there are four basic limitations: cognitive, physical, temporal, and institutional.

1. Cognitive Limitations – Agents as boundedly rational actors have cognitive limitations and therefore must join together to achieve higher-levels of performance.
2. Physical Limitations – Agents are limited physically, both because of their physiology and because of the resources available to them, and therefore must coordinate their actions, e.g., to achieve higher-levels of productivity. All action takes place situated in specific space-time locations, and agents are limited (e.g. by relativity limits) in their access to other space-time locations; this fundamental locality means that distributed action is fundamentally a

1. Other rationales include human needs for social affiliation, and the simple non-teleological emergence of patterns of activity in complex environments. However, in this chapter the focus is on the standard information processing approach.

multiagent—and hence potentially organized—phenomenon.

3. Temporal Limitations – Agents are temporally limited and therefore must join together to achieve goals which transcend the lifetime of any one agent.
4. Institutional Limitations – Agents are legally or politically limited and therefore must attain organizational status to act as a corporate actor rather than as an individual actor.

There is a plethora of ways in which organizations are constituted to overcome limitations of individual agency. Researchers in various areas refer to the way in which an organization is organized as the form, structure, architecture or design of that organization. Decades of research in this area have repeatedly shown that there is no single correct or proper organizational design. Field and survey research on actual human organizations, laboratory experiments on human groups, virtual experiments using computational models, and analyses using mathematical models all point to the same conclusion. There is no single organizational design that yields the optimal performance under all conditions. Which organizational design is optimal depends on a variety of factors including the specific task or tasks being performed, the intelligence, cognitive capabilities, or training of the agents, the volatility of the environment, legal or political constraints on organizational design, and the type of outcome desired (e.g., efficiency, effectiveness, accuracy, or minimal costs). The recognition by researchers of how organizational performance differentially depends upon multiple factors has led to the development of "contingency theories" of organization. From an organizational engineering perspective, locating an optimal organizational design for a specific, multidimensional situation is key. Whereas, from a theoretical perspective locating the general principles and tradeoffs underlying organizational design in a multidimensional space is key.

Consequently, research in this area has often focused on the search for general principles of organizing and the conditions under which these principles do or do not apply. For example, two such linked principles are specialization and the division of labor. Specialization of task or occupation refers to the principle that individuals can become more effective when they are expert in particular activities requiring particular and limited types of knowledge. Division of labor refers to the principle that appropriate division of tasks, knowledge, and skills among agents in an organization can improve organizational performance; e.g., by limiting task and knowledge dependencies. In general, organizations which employ specific and productive instances of these principles are able to overcome the limitations of individual agency, coordinate individual actions, and leverage training costs, skill development, and resources in such a way that the organization as a whole achieves higher levels of performance than are otherwise achievable. However, over-specialization and excessive division can reduce performance and flexibility by de-skilling individuals, decreasing attention due to boredom, and increasing decision making time, and by actually increasing coordination costs in situations of uncertainty or failure.

7.1.2 What Is Computational Organization Theory?

Researchers in the field of Computational Organization Theory (COT) use mathematical and computational methods to study both human and automated organizations as computational entities. Human organizations can be viewed as inherently computational because many of their activities transform information from one form to another, and because organizational activity is frequently information-driven.

COT attempts to understand and model two distinct but complementary types of organization. The first is the natural or human organization which continually acquires, manipulates, and produces information (and possibly other material goods) through the joint, interlocked activities of people and automated information technologies. Second, COT studies artificial computational organizations generally comprised of multiple distributed agents which exhibit collective organizational properties (such as the need to act collectively, an assignment of tasks, the distribution of knowledge and ability across agents, and constraints on the connections and communication among agents). Researchers use computational analysis to develop a better understanding of the fundamental principles of organizing multiple information processing agents and the nature of organizations as computational entities. The general aims of research in this area is to build new concepts, theories, and knowledge about organizing and organization in the abstract, to develop tools and procedures for the validation and analysis of computational organizational models, and to reflect these computational abstractions back to actual organizational practice through both tools and knowledge.

Research in this area has resulted in a large number of models, each with its own special characteristics. Many of these models focus on specific aspects of organizational behavior. Some research projects with particular illustrative models are listed in Table 7.1. These models differ in the way in which individual cognition is modeled. For example, in the Organizational Consultant there are no individual cognitive agents; in Sugarscape, the agents have physical positions and follow simple rules to respond to each other and their environment; in VDT agents are modeled as simple processors with in- and out-boxes; in CORP a simple model of experiential learning is used; in ORGAHEAD both experiential learning and annealing are used to model the decision process; and in Plural-Soar and TAC Air Soar a fully articulated model of human cognition is used. Further, differences in these models are effected by whether or not the agents within them can learn (see also Chapter 6). These models differ in the degree to which the organizational design is captured; e.g., in Sugarscape organizational design is not considered, but emergent patterns and structures of activity are an outcome of the model; the Organizational Consultant covers design in terms of a set of features; the Garbage Can model, AAIS, the CORP model, and the Cultural Transmission model all consider only a small set of designs; whereas HITOP-A, ACTION, ORGAHEAD, and VDT admit a wide range of explicitly parameterized designs. Models also differ on the extent to which specific features of tasks are modeled. In the Garbage Can model the task is generic and simply requires energy, in the Cultural Transition

Model	Author
Garbage Can	Cohen, March and Olsen (1972)
AAIS	Masuch and LaPotin (1989)
CORP	Carley (1992)
HITOP-A	Majchrzak and Gasser (1992)
Plural-Soar	Carley et al. (1992)
VDT	Cohen (1992), Levitt et al. (1994)
TAC Air Soar	Tambe (1997)
Organizational Consultant	Baligh, Burton and Obel (1990, 1994)
ACTION	Gasser, Majchrzak, et al., (1993,94)
ORGAHEAD	Carley and Svoboda (1996)
TAEMS	Decker (1995,1996)
Sugarscape	Epstein and Axtell (1996)
Cultural Transmission	Harrison and Carrol (1991)

Table 7.1 Illustrative Models

model shared knowledge rather than the task itself is considered, in CORP and ORGAHEAD a detailed classification task is used as a generic simulation activity; ACTION captures features of 141 generic manufacturing tasks; In VDT specific features of routine design tasks in which the precedent ordering among subtasks and needed skills can be explored, and TAEMS extends this to non-routine tasks as well.

Research in this area has also resulted in several "generalist" models that can be used in a number of applications in addition to their use in organizational theory. For example, one useful general model of information-seeking, decision making, and problem-solving activity in organizations is distributed search. Since formal computational models of search are well understood, modeling organizational activity as search can provide a clear and tractable explanatory framework (see Chapter 4). New approaches to control or task allocation in distributed search frameworks can, by analogy, provide suggestive new approaches to these problems in human organizations, e.g., in the development of new organizational forms or for reasoning about the effects of alternative strategic decisions. In the end, distributed search models provide just one type of abstraction that is useful for reasoning about problems of both human organizations and computational ones, and so help to unify thinking about both types.

Computational organization theories are most often grounded in existing cognitive, knowledge-based, information-processing theories of individual behavior. However, COT extends this to an organizational level [60, for example] and gives precision to the notion of bounded rationality by specifying the nature of the boundaries [7]. The original information processing perspective basically argued simply that agents were boundedly rational, that information is ubiquitous in the organization, and that the organization itself becomes a computational system. Today there is a neo-information processing perspective on organizational behavior that extends and refines this early view. The basic tenets of this neo-information processing perspective on organizations are:

- Bounded rationality: Organizational agents are boundedly rational. There are two types of bounds—limits to capabilities and limits to knowledge. Capabilities depend on the agents' cognitive, computational, and/or physical architecture. Knowledge depends on the agents' ability to learn and the agents' intellectual history. The agents' position in an organization influences to which information an agent has access. Thus, an agents' knowledge of how to do specific tasks, of how its specific organization operates, and indeed of how organizations operate in general, is a function of what positions the agent has held.
- Information ubiquity: Within organizations large quantities of information in many different forms are widely distributed across multiple agents. The information may not necessarily be correct.
- Task orientation: Organizations and the agents within them are continually engaged in performing tasks. The tasks in which an organization and its constituent agents are engaged require these agents to communicate, build on, analyze, adapt or otherwise process organizational information using various technologies, and to search out new information and new solutions.
- Distributional constraints: Organizational performance is a function of what information is shared by whom, when, and of the process of searching for that information. An organization's culture is the distribution of the knowledge and processes across the agents within it. This distribution affects the extent and character of socially shared cognition, team mental models, group information processing, and concurrent information analysis.
- Uncertainty: Uncertainty about task outcomes, environmental conditions, and about many other aspects of organizational life influences organizational activity. Distributed computational models such as distributed search or distributed constraint satisfaction pose distribution itself as a source of uncertainty: distribution can render critical uncertainty-reducing information less available because of the cost of seeking, transmitting, or assimilating it, and because of the overhead of coordinating information needs across agents.
- Organizational intelligence: Organizational intelligence resides in the distribution of knowledge, processes, procedures across agents and the linkages among agents. Organizations redesign themselves and their vision of their environments on the basis of the information available to them, with the aim of enabling them to better search for or process information. Such redesign is part of organizational learning processes. It can alter an organization's intelligence, and may or may not improve organizational performance.
- Irrevocable change (path dependence): As agents and organizations learn, their intelligence is irrevocably restructured. This one-directional evolution means that the kind and order in which things are learned—particular histories—can have dramatic consequences.

- Necessity of communication: In order to function as a corporate unit, agents within an organization need to communicate. This communication may take place explicitly by sending and receiving messages or implicitly by perceiving the actions of others.

In addition to this neo-information-processing view of organizations researchers in this area share a series of implicit background assumptions. These are:

- Modelability: Organizational phenomena are modelable.
- Performance differential: It is possible to distinguish differences in organizational performance.
- Manipulability: Organization are entities that can be manipulated and transformed.
- Designability: Organizations are entities that can be designed. This is not to say that organizations do not evolve, nor that they cannot be found in nature, for assuredly both events occur. However, they can also be consciously designed and redesigned: organizational transformations can be purposeful and principled.
- Practicality: Organizational transformations (based on the design or manipulation of models) can be transferred into and implemented in actual practice.
- Pragmatism: The costs of modeling and researching organizations using computational methods are relatively lower than the costs of manipulating or researching similar aspects of actual organizations in vivo, and the benefits gained outweigh the costs.

These assumptions that underlie the research in computational organization theory are the result of a fundamentally interdisciplinary intellectual history. Research in this area draws on work in distributed artificial intelligence (DAI), multiagent systems, adaptive agents, organizational theory, communication theory, social networks, and information diffusion. One of the foundational works in this area is The Behavioral Theory of the Firm [13] in which a simple information processing model of an organization is used to address issues of design and performance. While the strongest roots are in the information processing [60, 48, 64, 19, 13]. and social information processing [59], tradition, current models also have roots in the areas of resource dependency [54], institutionalism [56], population ecology [31], and symbolic interaction [21]. Formalisms and specific measures of organizational design are drawn from the work in the areas of coordination [45], social networks [65], and distributed control [12, 16, 41].

7.1.3 Why Take a Computational Approach?

Organizations are heterogeneous, complex, dynamic nonlinear adaptive and evolving systems. Organizational action results from interactions among adaptive systems (both human and artificial), emergent structuration in response to non-linear processes, and detailed interactions among hundreds of factors. As such, they are poor candidates for analytical models. Because of the natural complexity of the object of study, existing models and theories of organization are often vague, intuitive, and under-specified. Scientific progress will be more readily achievable if the theories are more explicit and well defined. Computational theorizing helps to achieve this.

Computational analysis is an invaluable tool for theory building and examining issues of organizational dynamics as it enables the researcher to generate a set of precise, consistent and complete set of theoretical propositions from basic principles even when there are complex interactions among the relevant factors. Computational models allow researchers to show proofs of concept and to demonstrate whether or not completely modelable factors can generate certain phenomena. In this way, computational models can be used to show the potential legitimacy of various theoretical claims in organization science.

Theoretical computational models can be used to demonstrate lower bounds or tractability of organizational information processing phenomena (e.g., minimal information necessary to reach distributed agreement or awareness [29], or the tractability of an organizational decision or negotiation processes [57]. Experimental and empirically-based models can also provide computationally-plausible accounts of organizational activity [36, 15].

7.2 Organizational Concepts Useful in Modeling Organizations

In order to model an organization the following factors are generally modeled at some level of detail: agents comprising the organization, the organization's design or structure, tasks the organization carries out, any environment of the organization, the organization's material transformation and/or information processing technology, and any stressors on the organization (see Figure 7.1). Organizations can use different configurations of agents, designs, tasks, and technology to accomplish the same goal—this is the concept of "equifinality." In fact, one of the major issues in the computational organization area is determining what organizational designs make sense when and what are the relative costs and benefits of these various configurations that exhibit degrees of equifinality.

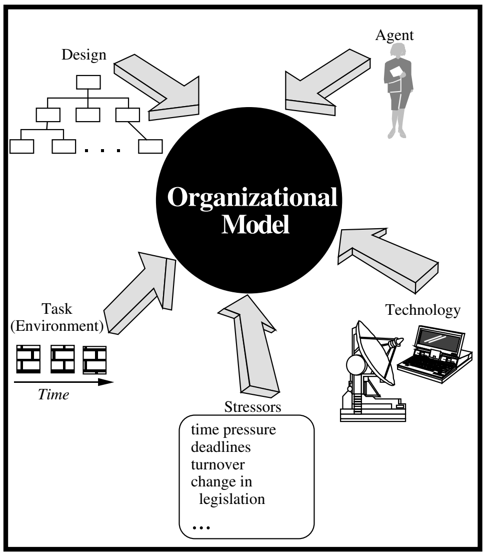

Figure 7.1 Necessary elements in an organizational model.

Models in the COT area vary dramatically in the level of detail in which agents, designs, tasks, and technology are modeled. The better or more detailed these underlying models, the more precise the predictions possible, but the greater the computational and modeling resources required. Models run the gamut from simple abstract models of generic decision making behavior (such as the Garbage Can Model and CORP) to detailed models of specific organizational decisions or decision making processes (such as VDT, HITOP-A and ACTION). For example, ACTION represents literally tens of thousands of organizational relationships among 14 different categories of organizational elements [23]. The simpler more abstract models are typically referred to as "intellective" models. These simpler models allow the researcher to use the model to simulate the general behavior of classes of organizations, policies, technologies, tasks or agents. For these models a central research goal is theory building: to discover general principles underlying organizational behavior.

The more detailed models are often referred to as "emulation" or "engineering" models. These detailed models may allow the researcher to use the model to emulate specific organizations by entering specific detailed parameters such as authority structures, detailed organizational procedures, or specific skill requirements. For these models a key research goal is organizational engineering: to examine whether or not the performance of a specific organization will be affected by making some specific change such as re-engineering the task in a particular way or adding a new technology.

7.2.1 Agent and Agency

In most COT models, organizations are composed of agents.[2] These agents may be human, artificial, or both. Agents take action, are constrained by their organizational role, and agents can make decisions. The actions of which the agents are capable, and the decisions that they make, depend on their capabilities and knowledge, the situation in which they are embedded, and the task(s) they are performing (see Figure 7.2).

Figure 7.2 is an adaptation of a scheme provided by Carley and Newell (1994) for thinking through the nature of agents and models employing such agents. In organizations, agents have some level of cognitive capability and occupy at a position in their organization. This position defines what task(s) the agent performs, with whom the agent must communicate, to whom the agent reports, who reports to the agent, and so forth. Agents have specific knowledge, skills, and capabilities. Classes of agents can be defined on the basis of differences in position, knowledge, skills, capabilities, or organizational roles. For example, one class of agents are

2. This need not be the case in general, and other approaches are being investigated. Most models take agents as the starting point, and compose organizations out of them. Some models, however, treat agents themselves as the emergent outcome of organizing processes. This is a conceptual necessary step to handle hierarchies of organizations and dynamic-organizations-as-agents. See further discussion below, and, e.g., [25, 33].

manager agents, another might be worker agents. Importantly, an agent's knowledge is potentially comprised not just of task-based or technical knowledge but also of social or organizational knowledge. Classes of agents, differing in their cognitive architecture and/or knowledge, would be capable of different actions.

From an artificial agent standpoint, what actions an agent can take is a function of the agent's cognitive capabilities and knowledge. Figure 7.2 is based on Carley and Newell's (1994) argument that the cognitive architecture serves to constrain what the agent can know and when the agent can know what and so constrains what types of actions are needed. Knowledge about the social and organizational world constrains what types of actions are possible. In Figure 7.2 as you move down each column the cognitive architecture becomes increasingly well specified and creates increasing need for more types of actions. As you move across each row the context becomes increasingly realistic and increasingly knowledge rich. An agent, in a particular cell, should be capable of taking all the actions to the left and up of its position. The MODEL SOCIAL AGENT, which is capable of all human actions, would be in the bottom right corner. Computational organizational models can be contrasted one with the other in terms of where they are at in this matrix of possibilities. For example, all models mentioned in Table 1 are positioned in the relevant cell in Figure 7.2.

Today, advances in the computational organization area are being achieved through the use of multiagent modeling techniques (see Chapters 1 and 2). In most organizational models, these multiple agents are viewed as cooperating together to achieve some collective goal such as producing more widgets, finding the best path through a maze, filling orders, or classifying objects (see Chapters 3 and 4). However, organizational agents need not be cooperative. Competition among agents may emerge for a variety of reasons. One of the most common examples is that of an organizational accounting system that reimburses individuals on the basis of individual contribution or performance and rather than on the overall organization performance. Moreover, competition among agents may even improve overall organizational performance on some tasks. Similarly, in most organizational models agents are viewed as essentially honest and as not knowingly communicating incorrect information or decisions. However, this need not be the case, particularly in organizations of people or in cases of information warfare. Ironically, for many organizational tasks, the task itself may be so ambiguous, that the agent may not be able to detect whether or not other agents are lying. Or, the task may be so constraining that lying is immediately obvious. MAS/DAI researchers have begun to investigate incentive models that link the rationality of truthfulness to the task structure of the domain and the constraints of the environment [57] but to date these models have not been incorporated into COT models.

Models of organizations have represented organizations as single decision makers and as collections of decision makers. Within the multiagent organizational models agents have been modeled in a variety of ways. For example, in the Garbage Can Model agents are characterized by a small set of parameters and vectors such as their energy, what problems they consider salient, what problems they can act on,

Knowledge
Increasingly Rich Situation →

Cognitive Architecture
Increasingly Limited Capabilities ↓

	Nonsocial Task	Multiple Agents	Real Interaction	Social Structural	Social Goals	Cultural Historical
Omniscient Agent	goal directed produces goods uses tools uses language	models of others turn taking exchange	face-to-face	class differences	organizational goals	historically situated
Rational Agent	reasons acquires learns	learns from others education negotiation	mis-communication	promotion social mobility	competition cooperation social cognition	emergent norms *Cultural Transmission*
Boundedly Rational Agent	satisfices task planning adaptation	group making	social planning coercion priority disputes	altruism uses networks for information boundary spanners *Garbage Can Model* *Sugarscape, AAIS*	delays gratification moral obligation *VDT* *TAEMS*	gate keeping role emergence *CORP, HITOP-A, ACTION, ORGAHEAD, Organizational Consultant*
Cognitive Agent	compulsiveness lack of awareness multi-tasking	group think	spontaneous exchange social interactions	automatic response to status cues	group conflict power struggles *TAC Air Soar* *Plural-Soar*	develop language institutional change
Emotional Cognitive Agent	habituation variable performance	protesting trust	play rapid emotional responce cons	campaining	team player	norm maintenance ritual maintenance advertising **MODEL SOCIAL AGENT**

Figure 7.2 Relating Cognitive Architecture and Knowledge to the Organizational Agent's Actions

and so forth. In this case the agent's abilities are represented as numerical arrays or values. For example, what problems are salient to an agent is represented by which cells in an agent by problem matrix are ones, and agent energy is simply a numeric value. In VDT the agent is modeled as an in-box, an out-box, a set of preferences for how to handle information, a set of skills, and so forth. In Plural-Soar each agent is modeled as a separate Soar agent. In this case the agent's knowledge is a set of rules in a series of problem spaces. In very sophisticated models, such as TAC Air Soar and Plural-Soar, agents are represented as complex, multilevel search processes, which have several interleaved levels and timescales of reasoning, and include both strategic and tactical modes.

7.2.2 Organizational Design

An organization's design can be conceptualized as a specific configuration of parameters that control the organization's behavior. The types of parameters generally considered include all of the configuration elements noted above (tasks, roles, organization structure, etc) as well as specific model-dependent parameters (inbox-sizes and delays for VDT, critical process variances for ACTION, etc.) Taken together, the parameters with their ranges of potential values define a parameter space, which in turn defines a space of potential organization designs. The process of designing an organization is in essence a process of refining and constraining that space to a single point (or to a set of points for a dynamically-restructurable organization).

Other commonly modeled design-oriented parameters include procedures and rules that embody organizational knowledge and the set of connections linking agents and tasks. Typical procedures and rules range from task-based rules, to accounting procedures, to procedures for hiring, firing, promoting, and moving agents about. The set of linkages among agents and tasks are often described as structures or networks. There can be many such structures in the organization. The structure perhaps most familiar to people who have worked organizations is the authority and communications structure. This is often represented in specific terms by an organization chart. However, the organization chart is most often simply the formal representation structure. In addition, there is a myriad of other interlinked structures that constrain and provide opportunities for action, and which are typically not perceived by those who work in the organization. One set of additional structures is the informal structure: the networks of interactions among agents such as friendship networks, and advice networks. In addition to these structures is are task structures (hierarchical and precedence ordering among subtasks, see also Chapter 3), task-resource structures (defining which resources are needed for which task), task-skill structures (defining what skills are needed for which task), resource access structures (defining which agent has access to which resources), skill structures (defining which agent has which skills), task assignment structures (defining which agent is assigned or is allowed to work on which subtasks), and so on. Modeling these structures in concert so as to account for and explain their interactions is a daunting and complex task for COT models.

Within the COT area the two most typical ways of conceptualizing the organization's design is as a set of attributes (such as centralized or decentralized communication, authority, skill, or other structures, or density of communication ties) or as a set of matrices. The attribute approach is used in HITOP-A, the Organizational Consultant, and AAIS. The matrix approach is used in the Garbage Can Model, CORP, VDT and ACTION. Illustrative structures in both their matrix and graphical network form are shown in Figure 7.3.

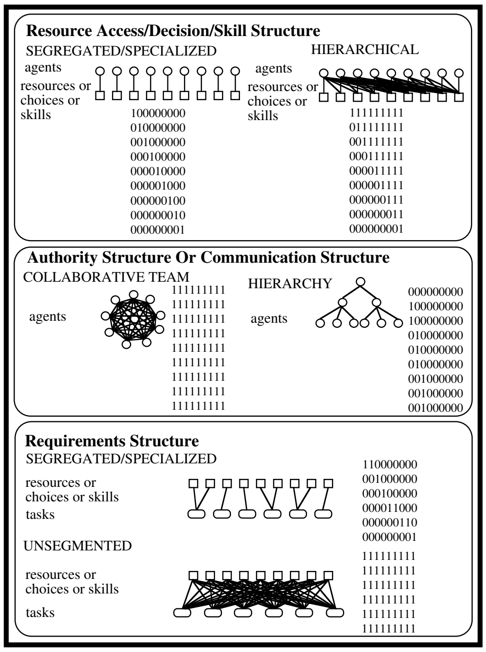

Figure 7.3 Illustrative structures that comprise an organization's design.

In Figure 7.3 two examples of each of three of the types of structures that comprise the overall design of an organization are shown. The resource access structure (top) links agents to resources; the decision access links agents to possible decision choices, and the skill structure links agents to their skills. Such structures show what is possible for each agent. A segregated or specialized structure implies that each agent is unique. This can imply little or no overlap in their mental models. A hierarchical structure at this level implies that there is one agent who has comprehensive access to, or knowledge about, the items in question. The authority and/or communication structures (middle) link agents to agents. In the authority structure the links show who has authority over whom and thus who reports to whom. In the communication structure the links show who talks to whom. In a collaborative structure all links are possible, in a hierarchy there is a central or apex agent. Finally, the requirements structure (bottom) links resources or skills with tasks. This structure shows what is needed to meet a certain task based goal. In the segregated or specialized structure each task has different requirements. In the unsegmented structure each task potentially requires all resources or skills. The structures shown are illustrative. Many other types of structures are possible. By thinking in terms of agents, resources (decisions or skills), tasks, and the linkages between them it is possible to describe analytically the set of possible designs and the designs as enacted by an organization.

Over time, researchers have begun to embed knowledge about organizations' structures in the agents so that it constrains the sharing of results and other information among organizational agents (see Chapter 3). In early organization design experiments with the Distributed Vehicle Monitoring Testbed (DVMT) [16] structure was embedded in each agent by (1) temporarily restricting its problem-solving capabilities (i.e., "skills") to a subset of its full set of available capabilities by "shutting off" some—this established a specialized "role"; (2) providing communication and reasoning mechanisms that linked current problem-solving capabilities (the current role) to agents and thus drove communications dynamically, and (3) providing strategic-level mechanisms for dynamically reconfiguring agent roles and role-to-agent maps (selectively activating and deactivating agent skills), thus implementing changes in the organization structure simply by changing the knowledge and skills of the agents.

In other cases, an organization's structure is represented as a series of rules for when to communicate what to whom and how to structure the communication. Representing organization's structure as a series of procedures or rules also facilitates linking up models of structure with models of intelligent agents. This approach is taken in the team (or multiagent) Soar work [37, 62, 63, 6]. In much of this work, organizational structure changes in response to changes in the environment because built into the agent's knowledge base are a set of standard operating procedures for how to restructure the communication under various conditions.

7.2.3 Task

The organization and its members are engaged in one or more tasks at any point in time. Further, over time, the organization repeatedly engages in a sequence of often quite similar tasks. These tasks may be composed of subtasks which may themselves be further subdivided. These tasks may have dependencies among them. Thompson identified three such dependencies—pooled (the results from two or more tasks are jointly needed to perform a different task), sequential (two or more subtasks must be performed in a specified sequence), and reciprocal (two tasks depend jointly on each other). Organizations are expected to improve their performance if they match the rest of their structures to the underlying task structure. (This concept of high performance as structural alignment is a characteristic explicitly represented in the mutual alignment approach of the ACTION model.)

The set of tasks faced by the organization can be thought of as its environment (or problem space). For example, in the car industry, the task can be viewed as a repeated design task. Each year, the industry generates new models that are more or less similar to the year before through a process of design and redesign. Task environments vary in many dimensions. Among the most important of these dimensions are degree of repetition, volatility, bias, and complexity. Some of these concepts are illustrated graphically in Figure 7.4.

Repetition: whether the set of tasks are repetitive (agents do the same thing over and over), quasi-repetitive (agents do the same type of thing repeatedly but the specific instances or details are different), or non-repetitive (each task is unique). This is shown graphically in Figure 7.4 by each task being composed of a set of bricks, and the pattern of bricks being different for each task.

Volatility: the rapidity with which the task environment changes. For example, in

Figure 7.4 when the task environment oscillates between two types of tasks (such as selling swimming pools in the summer and selling Christmas goods in the winter), this oscillation can be rapid (high volatility) or slow (low volatility).

Bias: the extent to which all possible tasks, regardless of task features, have the same outcome or solution. For example, in the binary choice task, a biased environment would be one where most outcomes were to choose 1 (as opposed to an unbiased environment where 1's and 0's would be equally likely).

Complexity: the amount of information that needs to be processed to do the task. For example, in the warehouse task, as the number of items on the order, the number of stacks, and the number of items per stack increases the complexity of the task increases.

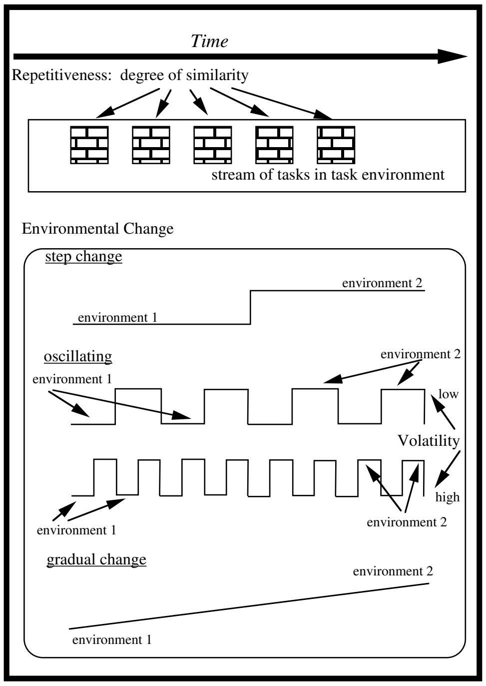

Figure 7.4 Characteristics of the task environment.

Typical task environment changes are step change, oscillating, and gradual (see Figure 7.4). In a "step change" environment, a discontinuity caused by legislation or technology alters the environment faced by the organization. For example, when a new manufacturing technology is introduced firms change from one task to another that is quite different. In an "oscillating" environment tasks or sets of tasks are chosen alternately from two different sets. Seasonal firms, for example, face oscillating environments. One such seasonal firm is the firm which sells swimming pools in the summer and Christmas goods in the winters. In an environment of gradual change, minor changes result in a gradual shift in the types of tasks faced by the organization. For example, the gradual aging and learning of students results in gradual changes in the types of problem sets the teachers must devise.

The performance of an organization can be measured with respect to the task or tasks it is performing. Three types of performance measures are commonly employed: effectiveness (is the task being performed well), efficiency (is the task being performed in such as way that output is maximized relative to some input), and perceived effectiveness (is the organization perceived as performing well by one or more stakeholders such as the general public, the government, the board of directors, or the media). For many tasks in which the product is generated by the group as a whole, while it might be possible to measure an organization's overall performance, in real human groups it is often impossible to objectively measure the actual contribution of any one member. Three aspects of effectiveness are: relative performance (how well is the organization performing compared to other organizations), accuracy (how many decisions are being made correctly), and timeliness (how rapidly are decisions begin made). For particular tasks or industries there are often entire literatures on how specifically to measure performance in specific situations. In general, for most models, multiple measures of performance are be gathered.

Within the COT area there are two strands of research on tasks. Some models, such as VDT, offer the capability of modeling a wide variety of organizational tasks, focusing on dependencies among subtasks but leaving the specific content of what is done in any particular subtask otherwise unspecified. In contrast, other models such as CORP are constrained to highly stylized and specific experimental tasks that, although retaining key features of actual tasks, differ in detail and complexity from those done in actual organizations. These highly stylized tasks are often referred to as canonical tasks, and they are valuable as research vehicles because researchers can share them among different modeling systems and environments and can more easily compare results. A set of such tasks is emerging for COT research. This set includes: the sugar-production task, the maze task, the binary classification task, the radar task, the warehouse task, the PTC task (production, transportation, and consumption), and the meeting scheduling task. In Figure 7.5 the binary choice task [5] and the warehouse task [6] are illustrated.

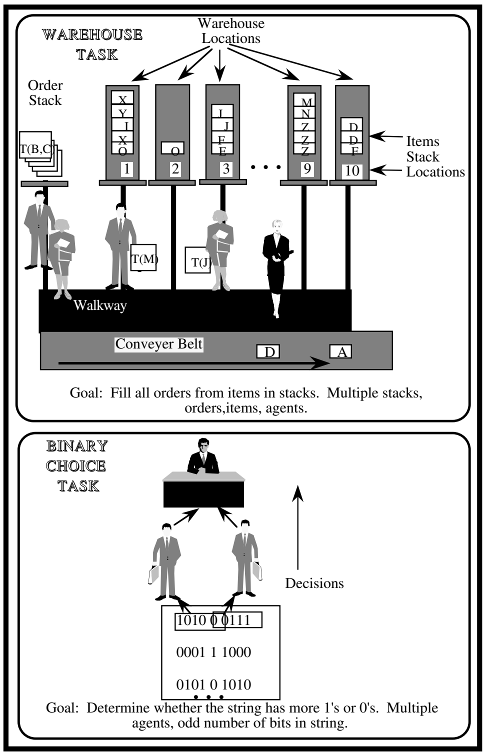

Figure 7.5 Illustrative tasks.

The warehouse task (Figure 7.5 top) is a simplified version of the more general search task. The key element of a search task is that there are a set of things being searched for, a set of locations where those things might be, and the organization must find all of the items. Organizational performance is based on the rapidity with which items are found and the effort spent in locating them. If the items cannot be depleted and the rule linking items to location does not change, then this problem, for a single agent, is simply a matter of remembering where things are. The task is complicated by agents being too boundedly rational to remember where things are or to act rapidly enough, by items being depleted or moved, and by information errors in what items are asked for and how items are represented at the various locations.

The binary choice task (Figure 7.5 bottom) is a simplified version of the more general classification choice task. The key element of a classification choice task is that there is a set of incoming information, a set of possible outcomes, and the organization must classify the incoming information to determine which outcome is most likely to be correct, and then the organization is told what was the correct classification. Organizational performance is based on the rapidity and accuracy with which a problem is classified. If the rule linking incoming information to choice does not change dynamically, then this problem for a single agent, is simply a matter of remembering what outcome is associated with which incoming pattern of information. This task is complicated by agents being too boundedly rational to be able to handle all of the information, by information errors or incomplete information, and by errors or latency in the feedback.

For both tasks, the goal is not to find the optimal way of solving this task. Rather, the goal of a computational organization theory model is to do the task as an organization would and to show the relative tradeoffs between different types of organizational designs, the relative impact of various communication technologies, or the relative benefits of different mixtures of agents.

From an organizational perspective, the question is how to structure a group of agents (what resource structure, what authority structure, what task structure, etc...) so that the organization as a whole exhibits high performance despite these, and other, types of complications (or stressors).

7.2.4 Technology

Research on organizational behavior needs to account for the role of information and telecommunication technologies. Most COT models ignore issues of technology within the organization. In a few cases, researchers have begun to look at how different technologies for processing, storing, retrieving, or communicating information might affect the actions of the individuals within the organization and/or the organization's overall performance. Two different approaches to examining the impact of technologies have appeared: technology as tool and technology as agent.

One approach to modeling technology is to treat it as a tool and to differentiate tools in terms of the attributes such as access, speed, synchronicity, and recordability. This approach is taken in the Virtual Design Team (VDT) [42, 35]. Within VDT, the organizational agents have a suite of communication technologies available to them like telephone, face-to-face meetings, and email. The agent also has a preference to send certain information via certain technologies. Using VDT the researcher can begin to examine how altering the way in which memos are sent, e.g., by paper or email, may affect the speed with which the organization solves a task and the amount of rework that needs to be done.

A second approach to modeling technology is to treat it as an artificial agent (and as altering the information processing capabilities of existing human agents who have access to the technology). This approach is taken in the constructural model

[36]. Within the constructural model the agents have an information processing or communication technology available to them which affects how many other agents they can simultaneously communicate with, whether the communicated message must be the same to all receivers, how much information the agent can retain, and so on. The technology itself may act as an agent with particular information processing and communicative capabilities. For example, a book as an artificial agent can interact with many other agents at the same time, send a different message to each receiver, can survive longer in the population than human agents, and, unlike human agents, cannot learn. As another example, a website as an artificial agent can interact with many other agents at the same time, send a different message to each receiver, has a shorter lifespan than its creator, and can "learn" by a specific subset of interactants adding information to the site. Within a computational model of an organization, such artificial agents can be represented simply as a separate class of agents with capabilities and knowledge that differ from humans and each other in systematic ways.

7.3 Dynamics

Many of the key issues in the COT area center around organizational dynamics. How do organizations learn? How do organizations evolve? What new organizational designs will emerge in the next decade or century? To examine such issues researchers use many tools, ranging from empirical observation and explanation of the behavior of complex agent-based models such as the DVMT, TAEMS, or Plural-Soar on dynamically-evolving tasks, to complex adaptive approaches or optimization approaches, such as genetic algorithms, genetic programming, neural networks, and simulated annealing. This work demonstrates that interactions between agent architecture, the way agents are coordinated, and the way the agents and the coordination structure adapt and change over time affect organizational behavior. Another common result is that for collections of even minimally intelligent agents organization often spontaneously emerges. Indeed, many studies show how structures that can be characterized as hierarchies sometimes spontaneously emerge.

Organizations are seen to be potentially dynamic in many ways. Organizations are capable of being redesigned or re-engineered moving the organization in a configuration space, making changes such as what agent reports to what other agent, or which agent handles what tasks. Agents within an organization are capable of changing; e.g., by learning or, in the case of artificial agents, by reconfiguring themselves or acquiring new knowledge. The processes, communications, or types of interactions can change, and so forth. There is a variety of processes that affect an organization's ability to adapt. For example, in order to achieve new levels of performance organizations often need to engage in an exploration process where they examine new technologies, new ways of doing business, new designs, and so on. However, organizations can also improve performance by exploiting current knowledge and technologies and getting better at what they do best. Exploration and exploitation are somewhat at odds with each other and organizations need to balance these two forces for change [46]. Organizational change is not guaranteed to improve performance. Organizations are typically more likely to redesign themselves when they are new and such changes may in turn cause the organization to fail. Such early failure is referred to as the liability of newness.

A variety of different approaches to organizational dynamics has been taken. To illustrate some of the issues involved two different approaches to will be briefly described. The first approach is based on the idea of flexible agents—agents which can restructure themselves in response to changes in the environment. The second approach is based on the idea of dual-level learning—organizations in which agent level and structural level learning occur in response to changes in the environment.

Ishida, Gasser and Yokoo [33, 34] (see also [25]) demonstrated the potential for flexible organizational structures to emerge from flexible agents. The basic idea underlying their approach is that the agents are flexible entities and knowledge interactions are the stable foundation of social grouping. Agents were modeled as sets of problem-solving production rules; i.e. mappings of individual rules to rule-collections. These mappings were flexible—rules could migrate from collection to collection (from agent to agent). Knowledge interactions were modeled as individual production rules. That is, an individual production rule is a way of saying that under certain meta-level control (rule-firing) conditions, certain particular knowledge (modeled as a LHS clause or input) interacts with certain other knowledge (another specific LHS clause or input) to produce specific additional knowledge (a particular RHS clause). This new RHS clause maps into (interacts to produce) an LHS clause for another rule or to an output. These production rules never changed. Agents flexibly emerged out of a fabric of interactions—production rules—which got mapped and re-mapped to rule collections. Agents re-configured their local knowledge and the structure or pattern of interactions among themselves, in response to changes in the environment. The actual number and character of agents changed over time as did the organizational structure. Which agents reported to which other agents was clearly not fixed, since the content and boundaries of agents themselves were not fixed. Experimental simulations showed that highly effective organizations tended to learn adaptations over time; i.e., on repeated exposure to similar (oscillating) environmental changes these organizations went through fewer adaptations over time. In this sense the organizations of agents learned how to learn.

Carley and Svoboda [9] used the ORGAHEAD model of organizational change to demonstrate the importance of learning histories and that organizations in which both the agents and the structure were flexible and could learn over time were not guaranteed to improve their performance. The ORGAHEAD model is based on the social conception of organizations as inherently complex, computational and adaptive in which knowledge and learning occurs at multiple levels. Within ORGAHEAD organizational action results from both the behavior of multiple agents and the structure (or network) connecting these agents and the way in which knowledge is distributed among these agents. Agents learn through gaining new knowledge through experience. This is implemented using a standard experiential learning model in which agents add new knowledge to old knowledge and continually update the probability with which they take certain actions based on the likelihood that the proposed action led to the desired outcome in the past. Learning occurs at the structural level—by altering procedures and linkages among the agents and tasks (such as who reports to whom and who does what)—as the organization redesigns and restructures itself. This strategic learning is implemented as a simulated annealing algorithm. In this case there is a chief executive officer (the annealer) that proposes a change, evaluates the potential impact of this change on the organization by trying to anticipate what will happen in the future, and then decides whether or not to accept the change according to the Metropolis criteria. According to the Metropolis criteria the change is always accepted if it is anticipated to improve performance and is accepted but with decreasing likelihood over time if the change is risky and is anticipated to decrease performance. The results from these studies show that not all change is advantageous. Individual and structural learning clash; e.g., organizations re-engineer themselves for better performance only to lose the lessons of experience learned by various agents as those new agents are moved to different tasks or leave the organization. Because of such learning clashes change often results in maladaptation. The history of how and when the organization changes is as much a determinant of the organization's performance as the organization's design. And, truly adaptive organizations, those whose performance actually improves over time are those organizations which engage in a meta-learning to balance change due to learning at the structural level with change due to gathering new experiences at the individual agent level.

7.4 Methodological Issues

There are numerous methodological issues involved in the development and testing of computational models of organizations and organizing. There are three, however, that require special mention. The first, is the use of virtual experiments to examine the implications of the computational model. The second, has to do with validation, verification and the relation of the computational models to data on organizations. The third, has to do with the role of development tools and frameworks to mitigate the costs of developing these models.

7.4.1 Virtual Experiments and Data Collection

Unlike early models in this area, such as the Garbage Can Model, today's computational models of organizations are often sufficiently complex that they cannot be completely analyzed. For example, the parameter space of set of options is frequently so extensive that the researcher cannot explore all possible input combinations to determine the performance of the system. Nevertheless, a large number of combinations need to be explored as the performance of the system may change drastically for different combinations of inputs. One of the reasons that modern models are so complex is that organizations themselves are complex. Another, is that models are often designed by representing process. As such, the same model can be used to address a number of questions about organizations and organizing.

To address this problem, researchers in this area run virtual experiments. A virtual experiment is an experiment in which the results are gathered via simulation. In running a virtual experiment the researcher sets up a series of simulations to address a specific question. For the virtual experiment the researcher chooses a small set of parameters—perhaps three—and then varies these systematically over some range. All other parameters are typically held constant or allowed to vary randomly in classic Monte Carlo fashion. Statistical procedures for designing and analyzing the resulting data can be used for virtual experiments just as they can for experiments using humans in the laboratory.

For example, imagine that the computational organizational model allows the researcher to control the number of agents, the way agents make decisions (following standard operating procedures or based on experience, how the agents can send messages (such as face-to-face one-on-one or group meetings, email to one other agent or to all other agents), the complexity of the task, the complexity of the organization's authority and communication structure, and a variety of other parameters or options. Such a model could be used to address a number of research questions including: (1) How large does the organization need to be to reap the benefits of email? And (2) for the same task are there different combinations of technology, authority structure, and communication structure that lead to the same level of performance? To address the first question the researcher might vary the size of the organization from say 2 to 200 in increments of 20 (11 cells) and may consider all four communication technologies. This would by a 11x4 experimental design. To address the second question the researcher might consider all four communication technologies, two different authority structures (e.g., team and hierarchy), and two different communication structures (e.g., a completely connected structure like everyone-to- everyone and one that follows the authority structure (only communication is to or from manager). This would be a 4x2x2 design. In each case some number of simulations would be needed to be run for each cell, with the number chosen based on the required power of the test.

7.4.2 Validation and Verification

Computational organization theory is a type of grounded theory [28]. That is, the models that embody the theory are informed by and tested against empirical data. This grounding is done using various validation and verification procedures. In the COT area three types of validation are particularly important: theoretical, external, and cross-model. Theoretical verification has to do with determining whether the model is an adequate conceptualization of the real world for assessing the key issue being addressed. The adequacy of the conceptualization is often determined on the basis of whether or not a set of situation experts consider the model to have captured the main factors that they observe in organizations. External validation has to do with determining whether or not the results from the virtual experiments match the results from the real world. Finally, cross-model validation has to do with determining whether or not the results from one computational model map on to, and/or extend, the results of another model.[3]

For both theoretical and external validation the real world may be a human organization, a laboratory experiment, or an organization of artificial agents, and so on. Organizations leave "traces" of their activities such as accounting records, stockholder reports, technical connections among parts, operating procedures, web pages, etc. These can be analyzed using computational methods. Such data can also be captured, mapped, analyzed, and linked to other computational models either as input or as data against which to validate the computational models. Such data helps to form and test computational theories of organization and organizing.

7.4.3 Computational Frameworks

One of the pressing issues in the COT area is the development of a general testbed or framework that has the appropriate building blocks to minimize the time required to develop organizational and social models. A variety of tools are beginning to appear; as yet, however, no one tool dominates. Among the existing tools are: MACE, SDML, Multiagent Soar, and SWARM.

MACE

MACE [20, 21] was one of the first general (domain-independent) testbeds for modeling multiagent systems. It was one of the first truly concurrent distributed object systems built. MACE introduced the idea of using agents for all phases of system construction, user interaction, and management of experiments, as well as for the basis of the modeled system itself. For example, "user interface agents" were used as actual asynchronous user-to-system and system-to-user representatives, interpreters, translators, displayers, managers, and so forth. This feature meant that the testbed and the experiment were an integrated multiagent organization for interacting with the experimenter and for testing ideas about the structure of organizations of artificial agents.

MACE also included explicit social modeling concepts drawn from sociological theory. One such idea was the notion of recursive composition of agents so that a group can itself be treated as an agent with distributed internal structure. In other words, agents, groups and groups of groups all have "agency"; i.e., a

3. Cross-model validation is also called docking [2].

set of specialized knowledge and a set of possible actions that can be taken. The second idea is that of the "social worlds." Herbert Blumer, Anselm Strauss, and other symbolic interactionists introduced the notion that individual people negotiate their lives by operating within social worlds which constrain both what they need to know and with whom they interact. In MACE, social worlds were operationalized as knowledge-based agent boundaries. Each agent defined a set of "acquaintances." This acquaintanceship knowledge, rather than explicit constraints, laws or testbed programming structures, defined the boundaries of communication and interaction, and hence the social structure. This concept provides a clean semantic model for flexible agent organizations. Finally, MACE used "modeling other agents" as its foundation of social knowledge and social structure, drawing on the ideas of G.H. Mead and the symbolic interactionists. Here the concept of 'taking the role of the other' served as a unifying principle for mind, self, and society over time. Acquaintances were also based on Hewitt's [32] ideas of Actor acquaintances (which were a much simpler notion, basically just message addresses for actors). MACE included specific facilities to model a number of features of other agents (including goals, roles, skills, etc.) in special a acquaintance database, and it used these to structure individual interactions and thus to establish social structure defined as patterns of interaction over time. This idea of modeling others and acquaintances has now become commonplace within MAS and DAI research; however, few researchers recognize the link they are making to social theory.

SDML

SDML (Strictly Declarative Modeling Language) [53, 17, 52] is a multiagent object-oriented language for modeling organizations. SDML is particularly suited for modeling multiagent systems in which the agents interact in a team (flat) or hierarchical organizational structure. SDML is effectively theory-neutral with respect to the cognitive capabilities of the agent. It is flexible enough to represent both simple agents and more sophisticated agents as well as the linkages among them. SDML currently includes various libraries for alternate architectures such as genetic programming and Soar. These libraries facilitate exploring the interaction between agent cognition and organizational design.

Key social ideas are captured in the SDML architecture. For example, social agents are capable of distinguishing between explanation and action. The declarative representation within SDML makes this possible. Within SDML agents in the same class can be represented by sharing rules between them. Another key idea in organizations is that within the organization there are predefined linkages among agents and predefined roles in which knowledge is embedded and that constrain behavior. From this perspective, structure is patterns of positions or roles over time. This notion of structure is integral to SDML as within SDML the structure of the multiagent system is represented as a container hierarchy. For example, agents may be contained within divisions which are contained within organizations. Containers and their associated agents are also linked by an inheritance hierarchy. Change in agents and in the linkages among them is made possible by controlling the time levels associated with agent and container data bases.

Multiagent Soar

Soar is a computational architecture for general intelligence [38]. Agents are goal directed and can be characterized in terms of their goals, problem spaces, states, operators, and associated preferences. Preferences can be used to represent shared norms or cultural choices about the existence of, acceptability of, or relative ranking of goals, states, problem spaces and operators. The agent's goals need not be articulable and can be automatically generated or consciously selected by the agent as deliberation ensues. The agent's long term knowledge base is a set of rules. The agent's short term memory is the set of information currently in working memory. Soar was designed as a specification of key psychological ideas such as bounded rationality. As such, Soar can be thought of as a unified theory of cognition. Indeed, empirical research on Soar suggests that in many instances its behavior is comparable to that of humans both in how well it does and in what errors are made.

Multiagent Soar is an approach to modeling teams as collections of Soar agents [37, 62, 63, 6]. The current Soar system facilitates inter-agent communication and does not require each agent to be a separate processor. Multiagent soar is built around three core social ideas: internal models of other agents, cognitive social structures, and communication. In multiagent Soar models, each team member is a Soar agent with a mental model of what other agents either know or will do in certain circumstances. This knowledge may include expectations about the other agents' goals, preferences, and so forth and allows the agent to anticipate what it thinks others will do. Further, each Soar agent in the team has embedded in its knowledge (its set of rules) a cognitive social structure. A cognitive social structure is an agent's perception of who interacts with whom, how, and about what. Finally, each Soar agent in the team has knowledge about how to communicate and what to communicate when and to whom, and how to compose and parse messages. Communication in these models is handled by passing communique's with specific task-related content. Within the multiagent Soar models, agent's typically monitor their environment and so can be interrupted by communications from other agents, changes in the environment, or changes in what other agents are present.

SWARM

SWARM is a multiagent simulation language for modeling collections of concurrently interacting agents in a dynamic environment [61, 51, 1]. SWARM emerged from of work in computational biology. As such, SWARM is particularly suited to exploring complex systems composed of large numbers of relatively simple agents which interact as they seek to optimize some function. Within SWARM the agents can, to an extent, dynamically restructure themselves to accommodate changes in the input data and the objective function. In a SWARM model it must be possible to define the objective function. SWARM agents can act either synchronously and asynchronously. Consequently, many different technological or biological constraints on communication and adaptation can be modeled within SWARM. One of the intended applications of SWARM is to artificial life applications. That is, one of the goals of SWARM models is to demonstrate that certain complex group level behaviors can emerge from concurrent interactions between agents who by themselves are not capable of exhibiting that complex behavior. One of the intents of a SWARM model is to "grow" realistic looking social behaviors. Todate, there has been little attempt to empirically validate whether the behaviors grown in SWARM models are comparable to those seen in human systems.

The key social idea that is captured in SWARM is the logic of collective intelligence. That is, over time systems of SWARM agents come to exhibit collective intelligence over and above the simple aggregation of agent knowledge. This notion of emergent intelligence is central to the science of complexity. A second key idea that is captured in SWARM is evolution. That is, there are large populations of agents who can engage in reproductive activities and cease to exist.

7.5 Conclusions

Computational organization theory (COT) is the study of organizations as computational entities. As noted, the computational organization is seen as taking two complementary forms: [1] the natural or human organization which is replete with information and the need to process it and [2] computational systems composed of multiple distributed agents which have organizational properties. Computational analysis is used to develop a better understanding of organizing and organizations.

Organization is seen to arise from the need to overcome the various limitations on individual agency—cognitive, physical, temporal, and institutional. Organizations, however, are complex entities in which one or more agents are engaged in one or more tasks and where knowledge, capabilities and semantics are distributed. Thus, each organization has a design, a set of networks and procedures linking agents, tasks, resources, and skills that describes these various distributions.

Computational organizational models are grounded operational theories. In other words, unlike traditional DAI or multiagent models COT models draw on and have integrated into them empirical knowledge from organization science about how human organizations operate and about basic principles for organizing. Much of this work follows in the information processing tradition. Many of the COT models are models composed of other embedded models. In these multi-level models, the traditional distinction between normative and descriptive often becomes blurred. For example, the models may be descriptive at the individual level—describing individuals as boundedly rational, with various built in cognitive biases—but normative at the structural level—finding the best organizational design subject to a set of task based or procedural constraints.

Computational analysis is not simply a service to organizational theorizing; rather, computational organizational theorizing is actually pushing the research envelope in terms of computational tools and techniques. COT makes contributions to mainstream AI and CS, including fostering progress on such issues as: large scale qualitative simulation, comparison and extension of optimization procedures (particularly procedures suited to extremely complex and possibly changing performance surfaces); aggregation/disaggregation of distributed objects; on-line/off-line coordination algorithms; organizational and multiagent learning; semantic heterogeneity; and understanding/managing the tradeoff between agent quantity and computational complexity. Research in this area requires further development of the scientific infrastructure including developing: easy-to-use cost-effective computational tool kits for designing and building computational models of organizations, teams, and social systems (e.g., a multiagent oriented language with built in task objects and communication); multiagent logics; intelligent tools for analyzing computational models; validation procedures, protocols, and canonical data sets; managerial decision aids based on computational organization models; and protocols and standards for inter-agent communication. Key theoretical concerns in this area center around determining: what coordination structures are best for what types of agents and tasks; whether hybrid models (such as a joint annealer and genetic programming model) are better models for exploring organizational issues and for locating new organizational designs; representations for, and management of, uncertainty in organizational systems; the interactions among, and the relative advantages and disadvantages of various types of adaptation, evolution, learning, and flexibility; measures of organizational design; the existence of, or limitations of, fundamental principles of organizing; the tradeoffs for system performance of task-based, agent-based, and structure-based coordination schemes; representations for information and communication technology in computational models; and the relation between distributed semantics and knowledge on teamwork, organizational culture and performance.

Three directions that are particularly important for future research are organizational design, organizational dynamics and organizational cognition. The key issue under organizational design is not what is the appropriate division of labor, nor is it how should agents be coordinated. Rather, there is a growing understanding that there is a complex interaction among task, agent cognition or capabilities, and the other structural and procedural elements of organizational design. As such, the issue is finding what combinations of types of agents, structures (patterns of interactions among agents), and ways of organizing the task are most likely to meet the organization's goal. The key issue for organizational dynamics is not whether or not organizations adapt. Rather, the issues center on how to encourage effective learning, how to change to improve performance, how to retain capabilities and knowledge as the organization changes to address changes in the environment, and what new designs are possible. As to organizational cognition (perception, memory) there are a variety of issues ranging from how to represent organizational knowledge, to what level of sharing (of knowledge, procedures, or semantics) is necessary and by which agents to ensure effective organizational performance.

7.6 Exercises

1. *[Level 1]* Provide a critical discussion of the following statement. You do not need to know organizational theory to create good models of organizations. Anyone who has ever worked in an organization can develop such models.
2. *[Level 1]* Provide a critical discussion of the following questions. How does the organizational design and the task bound the agent? What are typical constraints and opportunities afforded the agent by the design and task? Provide at least five examples for both design and task.
3. *[Level 1]* For an organization that you are familiar what types of agents exist in that organization, what are their limitations.
4. *[Level 1]* Develop a measure of coordination based on structures like those shown in Figure 7.3.
5. *[Level 2]* Develop a simple model of a small group of agents (1,2 or 3) trying to collectively solve a simple canonical task such as the binary choice task or the maze task. What additional issues are involved, and what extra features does the model need, as you move from 1 to 2 to 3 agents working together to do the task? How is performance affected by the increase in the number of agents?
6. *[Level 3]* Reimplement and extend in one or more ways the garbage can model of organizational choice [11]. There are many possible extensions, some of which have been discussed in the literature. Possible extensions include, but are not limited to the following: adding a formal organization authority structure, having agents work on multiple tasks simultaneously, altering the task so that it requires specific skills and not just energy to be completed, and allowing agent turnover. Show that your model can replicate the original results reported by Cohen, March and Olsen (i.e., dock the models [2]). Then show which results are altered, or what additional results are possible, given your extension.
7. *[Level 3]* Reimplement and extend in one or more ways the CORP model of organizational performance [43]. There are many possible extensions, some of which have been discussed in the literature. Possible extensions include, but are not limited to the following: adding an informal communication structure, having agents work on multiple tasks simultaneously, allowing agents to be promoted, altering incoming information so that it is potentially incomplete or erroneous, altering the nature of the feedback (e.g., by delaying it or making it more ambiguous), and making the agents competitive (e.g., make agents try to maximize the performance relative to other's performance). Show that your

model can replicate the original results reported by Lin and Carley (i.e., dock the models). Then show which results are altered, or what additional results are possible, given your extension.

8. *[Level 3]* For a small organization (5 to 30 people) develop a description of its design. What is the formal organization chart? What is the informal advice network (who goes to whom for work related advice)? What are the main tasks and subtasks being accomplished? Develop a task dependency graph matrix. What are the skills or resources needed to do those tasks? Develop a resource/skill access matrix and a resource/skill requirements matrix. What were the major difficulties you encountered in locating this information for an actual organization?
9. *[Level 4]* Develop a comprehensive representation scheme for task or a multiagent language for doing task based models of organizations. Consider how task is represented in various organizational models. What are the limitations or features of the various representation schemes? What features should be built into your approach? Demonstrate the strength of your approach by reimplementing one or more existing COT models.
10. *[Level 4]* Develop a general purpose approach for modeling telecommunication and information processing technologies in the organization. What are the critical features or components of these technologies that must be modeled? How does your approach contrast with the technology as agent approach and the technology as feature approach? What are the limitations and advantages of your approach?

7.7 References

1. R. Axelrod. Advancing the art of simulation in the social sciences. Working Paper 97-05-048, Santa Fe Institute, 1997.
2. R. Axtell, R. Axelrod, J. M. Epstein, and M. D. Cohen. Aligning simulation models: A case study and results. *Computational and Mathematical Organization Theory*, 1(2): 123–142, 1996.
3. H. H. Baligh, R. M. Burton and B. Obel. Devising expert systems in organization theory: The organizational consultant. In M. Masuch, editor, *Organization, Management, and Expert Systems*, pages 35–57. Walter De Gruyter, Berlin, 1990.
4. H. H. Baligh, R. M. Burton and B. Obel. Validating the organizational consultant on the fly. In K. M. Carley and M. J. Prietula, editors, *Computational Organization Theory*, pages 179–194. Lawrence Erlbaum Associates, Hillsdale, NJ, 1994.
5. K. M. Carley. Organizational learning and personnel turnover. *Organization Science*, 3(1): 2–46, 1992.
6. K. M. Carley, J. Kjaer-Hansen, M. Prietula and A. Newell. Plural-Soar: A prolegomenon to Artificial Agents and Organizational Behavior. In M. Masuch and M. Warglien, editors, *Distributed Intelligence: Applications in Human*

Organizations, pages 87–118. Elsevier Science Publications, Amsterdam, The Netherlands. 1992.

7. K. M. Carley and A. Newell. The nature of the social agent. *Journal of Mathematical Sociology*, 19(4): 221–262, 1994.
8. K. M. Carley and M. J. Prietula, editors. *Computational Organization Theory.* Lawrence Erlbaum Associates. Hillsdale, NJ, 1994.
9. K. M. Carley and D. M. Svoboda, Modeling organizational adaptation as a simulated annealing process. *Sociological Methods and Research*, 25(1): 138-168, 1996.
10. G. P. Cohen. *The Virtual Design Team: An Information Processing Model of Coordination in Project Design Teams.* PhD thesis, Stanford University, Department of Civil Engineering. Stanford, CA, 1992.
11. M. D. Cohen, J. G. March and J. P. Olsen. A garbage can model of organizational choice, *Administrative Sciences Quarterly* 17(1): 1–25, 1972.
12. D. D. Corkill. *A Framework for organizational self- design in distributed problem solving networks.* PhD Thesis, University of Massachusetts, Dept. of Computer and Information Science, Amherst, MA, 1982.
13. R. Cyert and J. G. March. *A Behavioral Theory of the Firm*. 2nd Edition. Blackwell Publishers, Cambridge, MA, 1992[1963].
14. K. Decker, 1995 A framework for modeling task environment. Chapter 5 in Environment-Centered Analysis and Design of Coordination Mechanisms. Ph.D. Dissertation, University of Massachusetts.
15. K. Decker, 1996, TAEMS: A framework for environment centered analysis and design of coordination mechanisms. In *Foundations of Distributed Artificial Intelligence,* G.M.P. O'Hare and N.R. Jennings, eds. New York: John Wiley and Sons.
16. E. H. Durfee, V. R. Lesser and D. D. Corkill. Coherent cooperations among communicating problem solvers. *IEEE Transactions on Computers,* C-36: 1275-1291, 1987.
17. B. Edmonds, M. Scott and S. Wallis. Logic, reasoning and a programming language for simulating economic and business processes with artificially intelligent agents. In P. Ein-Dor, editor, *Artificial Intelligence in Economics and Management*, pages 221–230. Kluwer Academic Publishers, Boston, MA, 1996.
18. J. M. Epstein and R. Axtell. *Growing Artificial Societies.* MIT Press, Cambridge, MA, 1997.
19. J. R. Galbraith, *Designing Complex Organizations,* Addison-Wesley Publishing Company, 1973.
20. L. Gasser, C. Braganza and N. Herman. MACE: A flexible testbed for distributed AI research. In M. N. Huhns, editor, *Distributed Artificial Intelligence*, pages 119-152. Pitman Publishers, 1987a.
21. L. Gasser, C. Braganza and N. Herman, Implementing distributed AI systems using MACE. In *Proceedings of the 3rd IEEE Conference on Artificial Intelligence Applications,* pages 315–320. Orlando, FL, February, 1987b. (Also in Readings in DAI).
22. L. Gasser and A. Majchrzak. ACTION Integrates manufacturing strategy, design, and planning. In P. Kidd and W. Karwowski, editors, *Ergonomics of Hybrid Automated Systems IV*, IOS Press, Netherlands, 1994.

23. L. Gasser, I. Hulthage, B. Leverich, J. Lieb, and A. Majchrzak, "Organizations as Complex, Dynamic Design Problems", in M. Filgueiras and L. Damas, eds. *Progress in Artificial Intelligence.* Lecture Notes in Artificial Intelligence 727, Springer Verlag, 1993.
24. L. Gasser and A. Majchrzak. HITOP-A: Coordination, infrastructure, and enterprise integration. In em Proceedings of the First International Conference on Enterprise Integration, Hilton Head, South Carolina: MIT Press, Cambridge, MA, 1992.
25. L. Gasser and T. Ishida. A dynamic organizational architecture for adaptive problem solving, In *Proceedings of the National Conference on AI*, pages 52-58. July, 1991.
26. N. S. Glance and B. A. Huberman. The outbreak of cooperation. *Journal of Mathematical Sociology*, 17(4): 281–302, 1993.
27. N. S. Glance and B. Huberman. Social dilemmas and fluid organizations. In K. M. Carley and M. J. Prietula, editors, *Computational Organization Theory*, pages 217–240.Lawrence Erlbaum Associates. Hillsdale, NJ, 1994.
28. B. Glaser and A. L. Strauss, *The Discovery of Grounded Theory: Strategies for Qualitative Research*, Walter de Gruyter, 1967.
29. R. Fagin, J. Y. Halpern, and Y. Moses, *Reasoning About Knowledge*, MIT Press, Cambridge, MA, 1995.
30. J. R. Harrison and G. R. Carrol. Keeping the faith: A model of cultural transmission in formal organizations. *Administrative Science Quarterly*, 36: 552–582, 1991.
31. M. T. Hannan. and J. Freeman. *Organizational Ecology.* Harvard University Press, Cambridge, MA, 1989.
32. C. E. Hewitt, "Viewing Control Structures as Patterns of Passing Messages," *Artificial Intelligence*, 8(3): 323–364, 1977.
33. T. Ishida, L. Gasser and M. Yokoo. Organization self-design of distributed production systems. *IEEE Transactions on Data and Knowledge Engineering*, 4(2): 123–134, 1992a.
34. T. Ishida, L. Gasser and M. Yokoo. An organizational approach to real-time continuous problem solving. *Japanese Society of Artificial Intelligence Journal*, 7(2): 300–308, 1992b (In Japanese).
35. Y. Jin and R. Levitt. The Virtual Design Team: A computational model of project organizations. *Computational and Mathematical Organization Theory*, 2(3): 171–196, 1996.
36. D. S. Kaufer and K. M. Carley. *Communication at a Distance: The Effect of Print on Socio-Cultural Organization and Change.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1993.
37. J. E. Laird, Jones, and Nielsen. Knowledge-based multi-agent coordination. *Presence,* in press.
38. J. E. Laird, A. Newell, and P. S. Rosenbloom. Soar: An architecture for general intelligence. *Artificial Intelligence,* 33:(1): 1-64, 1987.
39. J. E. Laird, P. S. Rosenbloom, and A. Newell. *Universal Subgoaling and Chunking: The Automatic Generation and Learning of Goal Hierarchies.* Boston, Massachusetts: Kluwer Academic Publishers. 1986.
40. T. K. Lant and S. J. Mezias. Managing discontinuous change: A simulation study of organizational learning and entrepreneurial strategies, *Strategic Management Journal*, 11: 147–179, 1990.

41. V. R. Lesser and D. D. Corkill. Functionally accurate. cooperative distributed systems. In A. H. Bond and L. Gasser, editors, *Readings in Distributed Artificial Intelligence*, Morgan Kaufmann, Inc., San Mateo, CA, 1988.

42. R. E. Levitt, G. P. Cohen, J. C. Kunz, C. I. Nass, T. Christiansen and Y. Jin. The 'Virtual Design Team': simulating how organization structure and information processing tools affect team performance. In K. M. Carley and M. J. Prietula, editors, *Computational Organization Theory* pages 1–18. Lawrence Erlbaum Associates, Hillsdale, NJ, 1994.

43. Z. Lin, and KṀ. Carley. Organizational response: The cost performance tradeoff. *Management Science,* 43(2): 217–234, 1997.

44. A. Majchrzak and L. Gasser, HITOP-A: A tool to facilitate interdisciplinary manufacturing systems design, *International Journal of Human Factors in Manufacturing*, 2(3): 255–276, 1992.

45. T. W. Malone. Modeling coordination in organizations and markets. *Management Science*, 33: 1317–1332, 1986.

46. J. G. March. Exploration and exploitation in organizational learning. In M. D. Cohen and L. S. Sproull, editors, *Organizational Learning*. Sage, Thousand Oaks, CA, 1996.

47. J. G. March and R. Weissinger-Baylon, editors. *Ambiguity and Command: Organizational Perspectives on Military Decision Making*. Pitman, Boston, MA,1996.

48. J. G. March and H. A. Simon. *Organizations.* Wiley, 1958.

49. M. Masuch and M. Warglien. *Artificial Intelligence in Organization and Management Theory* Elsevier Science Publishers, Amsterdam, The Netherlands,1992.

50. M. Masuch and P. LaPotin (1989). Beyond Garbage Cans: An AI Model of Organizational Choice. *Administrative Science Quarterly*, 34: 38–67.

51. N. Minar, R. Burkhart, C. Langton, M. Askenazi. The Swarm simulation system: A toolkit for building multi-agent simulations. Working Paper 96-06-042. Santa Fe Institute.1996.

52. S. Moss and O. Kuznetsova. Modelling the process of market emergence. In J. W. Owsinski and Z. Nahorski, editors, *Modelling and analysing economies in transition*, pages 125–138. MODEST, Warsaw. 1996.

53. S. Moss and B. Edmonds. A formal preference-state model with qualitative market judgements. *Omega—the international journal of management science*, 25(2): 155–169, 1997.

54. J. Pfeffer and G. R. Salancik. *The external control of organizations : a resource dependence perspective.* Harper and Row, New York, 1978.

55. J. F. Padgett. Managing garbage can hierarchies. *Administrative Science Quarterly*, 25(4): 583-604, 1980.

56. W. W. Powell and P. J. DiMaggio. *The New Institutionalism in Organizational Analysis.* The University of Chicago Press, Chicago, IL, 1991.

57. J. SṘosenschein and G. Zlotkin. *Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers*, MIT Press, Cambridge, MA, 1994.

58. G. R. Salancik and H. Leblebici. Variety and form in organizing transactions: A generative grammar of organization. *Research in the Sociology of Organizations*, 6:1–31, 1988.

59. G. R. Salancik and J. Pfeffer. A social information professing approach to job attitudes and task design. *Administrative Science Quarterly*, 23:224–253. 1978.
60. H. A. Simon, *Administrative Behavior.* Free Press, New York, 1947.
61. J. Stites. Complexity. *Omni*, 16(8):42–52. 1994.
62. M. Tambe. Teamwork in real-world, dynamic environments. In *Proceedings of the International Conference on Multi-agent Systems*, 1996.
63. M. Tambe. Agent Architectures for flexible, practical teamwork. In Proceedings of the AAAI American Association of Artificial Intelligence, 1997.
64. J. Thompson. *Organizations in Action.* McGraw- Hill, New York, NY, 1967.
65. S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications.* Cambridge University Press, New York, NY, 1994.