MANAGEMENT SCIENCE
Vol. 52, No. 5, May 2006, pp. 671–682
ISSN 0025-1909 | EISSN 1526-5501 | 06 | 5205 | 0671

DOI 10.1287/mnsc.1050.0496

The Contingent Effects of Transactive Memory: When Is It More Beneficial to Know What Others Know?

Yuqing Ren
Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213,
yren@andrew.cmu.edu

Kathleen M. Carley
Institute of Software Research International, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213,
kathleen.carley@cs.cmu.edu

Linda Argote
Tepper School of Business, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213,
argote@andrew.cmu.edu

Previous studies have provided evidence of the positive impact of transactive memory (TM) on group performance, such as the efficient storage and recall of knowledge and better product quality. This paper aims to unify the experimental research on TM and to extend it to more dynamic and diverse group settings. In this paper, we develop an empirically grounded computational model—ORGMEM—and apply it to explore the contingent effects of TM on group performance. The comparison between virtual experimental results and relevant laboratory experimental results demonstrates the validity of ORGMEM as a useful tool to study memory-related phenomena. Through a series of virtual experiments, we find that TM decreases group response time by facilitating knowledge retrieval processes and improves decision quality by informing task coordination and evaluation. Our results also suggest that the effects of TM are contingent upon group characteristics, such as group size and environment, as well as the dimension along which group performance is assessed. Overall, TM seems to be more beneficial to small groups using quality as the dependent variable, but more beneficial to large groups, groups in a dynamic task environment, and groups in a volatile knowledge environment using time as the dependent variable.

Key words: teams; transactive memory; group performance; contingency theory; knowledge management; computational modeling; organizational learning
History: Accepted by John W. Boudreau, organizational behavior, performance, strategy, and design; received July 27, 2004. This paper was with the authors 6 months for 2 revisions.

Introduction

Knowledge has become more and more crucial in determining the competitiveness of both firms and individuals (Grant 1996). Yet, specialization and the huge volume of information in modern society means that, people cannot possess all of the knowledge they need in their work. They rely on external assistance such as co-workers, books, databases, and the Internet to retrieve information and solve problems. To do so, they need to know where the required knowledge is located and be able to acquire it in a timely manner.

Wegner (1987) presented the concept of transactive memory (TM) as a shared system that people in close relationships develop for encoding, storing, and retrieving information from different domains. Scholars and practitioners have found the notion of social knowledge or metaknowledge to be a powerful concept (e.g., Argote et al. 2003, Stewart 1995). Both direct and indirect evidence of the positive effects of TM on group performance exists. For instance, TM facilitates the storage and recall of knowledge through interpersonal relationships (Wegner et al. 1991). People working in the same group tend to become specialized in different domains. By knowing who is good at what and directing new knowledge in a specific domain to the experts, group members are able to acquire and store knowledge more efficiently as a whole than as individuals. Groups also make better decisions when group members recognize the relative distribution of expertise within the group (Henry 1995, Hollenbeck et al. 1995, Littlepage et al. 1997). A series of laboratory experiments suggest that groups whose members are trained together recall more and perform better than those whose members are trained separately (Liang et al. 1995, Moreland et al. 1996). There is also evidence that TM improves group performance in field settings (e.g., Austin 2003, Faraj and Sproull 2000, Lewis 2003). Most studies so far have been laboratory experiments, in which two or three participants perform a single task in a stable environment (e.g., Hollingshead 1998, Liang et al. 1995). So, it is not clear to what extent the findings can be generalized to other group settings. For example, what happens if a group has more than three members, takes on multiple tasks, or operates in an environment where technologies change and people forget what they knew? To generalize, researchers would need to vary and examine factors such as group size, group task, and knowledge environment across a broad range of values. Studying these phenomena by running laboratory experiments using human participants would be extremely costly. For instance, running laboratory experiments using groups of six different sizes (3–35), varying the frequency at which groups change tasks (task volatility) and the rate at which knowledge decays (knowledge volatility) at three levels (low, medium, and high), and assume nine groups under each condition, researchers would need to recruit and coordinate a minimum of 10,000 human participants! This is a situation in which we think computational modeling can make a contribution.

In this paper, we examine the relationship between TM and group performance using computational modeling techniques. This comparatively new method complements and extends the existing literature in several regards. First, as the number of variables that are of interest increases, it becomes difficult for researchers to derive predictions from their intuitive thinking and theoretical reasoning (Kraut et al. 2004). Doing "what if" exercises using computational models enables researchers to add precision to theory building and identifying, articulating, and testing the underlying logic (Monge and Contractor 2003). Second, computational modeling can be used to verify and extend existing theories that were derived from logical reasoning or conventional research methods, especially theories that involve complex, dynamic, and nonlinear relationships (Carley and Prietula 1994). In conjunction with research paradigms, such as experimental and field methods, computational modeling provides a triangulated view of phenomena and enables researchers to test causal dynamic theories (Hulin and Ilgen 2000). Finally, computational modeling enables researchers to examine larger and richer settings, such as groups with 20 or 30 members, and to examine them with relatively greater ease.

The rest of this paper is organized as follows. In the next two sections, we describe the design and implementation of the computational model we developed, ORGMEM. Then, we present the measures of group performance and TM. Next, we validate the computational model by comparing virtual experimental results with previous laboratory findings. Finally, we apply the model to explore the contingent effects of TM on group performance.

The Computational Model: ORGMEM

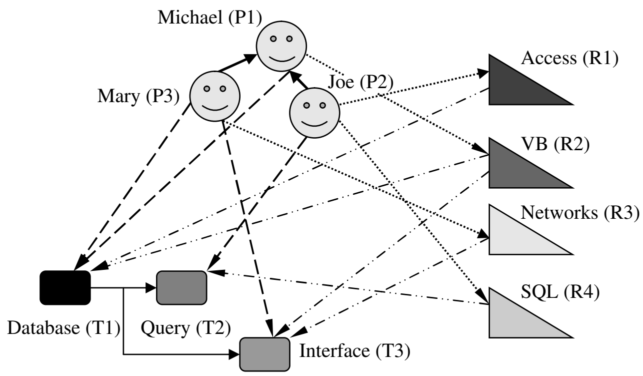
ORGMEM is a multiagent system that simulates interpersonal communication, information processing, and decision-making processes in organizations. ORGMEM agents are intelligent, adaptive, and heterogeneous (Ren 2001). Each agent has access to specifiable intellectual or physical resources, undertakes responsibilities for subtasks, and interacts with other agents. Each agent has TM about who talks to whom, who knows what, and who does what in the group. Over time, groups receive a series of tasks, which are divided among agents in the group. Agents work on assigned subtasks, search for required resources, and make decisions. As a result, group communication structures regarding who talks to whom, skill structures regarding who has access to what, and TM structures regarding who knows what change over time.

Group Modeling

We define a group as a collection of individuals who are interdependent in their tasks and share responsibility for outcomes (Cohen and Bailey 1997). ORGMEM models groups as multiagent information-processing and decision-making units using the PCANNS scheme (Krackhardt and Carley 1998). PCANNS assumes that there exist three key elements in a group: people (P), resources (R), and tasks (T). Accordingly, there exist six relational primitives among these elements: (1) **P**recedence of tasks ($T \times T$), (2) **C**apabilities linking people to resources ($P \times R$), (3) **A**ssignment of tasks to people ($P \times T$), (4) **N**etworks among people ($P \times P$), (5) Resource **N**eeds of tasks ($R \times T$), and (6) **S**ubstitutes of resources ($R \times R$) (Carley et al. 2000). A group is represented as six relational matrices in which cell values are either 1 or 0. A value of 1 indicates that a connection exists between two elements; a value of 0 indicates that there is no connection. For example, the assignment matrix ($P \times T$) indicates who is assigned to which task. $A_{ij} = 1$ means that person i is assigned to task j and $A_{ij} = 0$ means that person i is not assigned to task j.

Figure 1 depicts a canonical example of the group representation schema. Suppose that a group works together to develop an online transaction website. Michael is the team leader who supervises two subordinates, Mary and Joe ($P \times P$ matrix). The group task involves three subtasks (database development, interface construction, and query implementation) and requires four areas of expertise (Microsoft Access,

Figure 1 Illustrative Group Representation Scheme and an Example Group



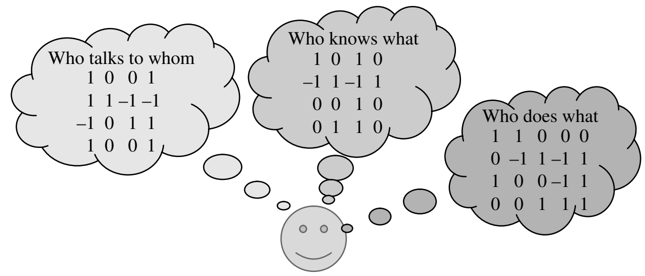
	People	Resources	Tasks
People	000	0100	100
	100	1001	010
	100	0010	101
Resources		0000	100
		0000	101
		0000	001
		0000	010
Tasks			011
			000
			000

Visual Basic (VB), SQL, and networks). According to the Capability matrix (P × R), Michael is knowledgeable in VB, Mary is knowledgeable in networks, and Joe is knowledgeable in Access and SQL. According to the Assignment matrix (P × T), Michael and Mary are responsible for database development, which has to be finished before the other two subtasks, as indicated in the Precedence matrix (T × T). The assignment matrix also indicates that Mary and Joe are assigned to work on interface construction and query implementation, respectively. The Substitute matrix (R × R) shows no link between any two areas of knowledge, which suggests that these areas of expertise are not fungible. Finally, the Needs matrix (R × T) indicates that the group requires knowledge about Access and VB to develop the database, knowledge about SQL to implement queries, and knowledge about VB and networks to construct the interface. Using PCANNS, the group setting and the interrelationships among people, resources, and tasks can be captured in the six relational matrices as shown in Figure 1.

Agent Modeling

In ORGMEM, each agent can be assigned a name and a title (analyst, manager, central executive office, or president), and can be given certain skills and responsibilities. Each agent can develop TM of who talks to whom (P × P), who has access to which resources (P × R), and who is assigned to which tasks (P × T), as shown in Figure 2. We represent TM using a trinary rather than a binary scheme to reflect three possible states of TM. A value of 1 indicates that the agent knows there is a connection between two elements. A value of −1 indicates that the agent knows there is no connection between two elements. A value of 0 indicates that the agent does not know about the connection.

Figure 2 Representation of TM in ORGMEM



In ORGMEM, as in real groups (Krauss and Fussell 1990), members construct and modify their TM through interpersonal communication and observation of others (see Figure 2). Suppose each member initially has only knowledge about his or her own connections to other people, resources, and tasks. Through interactions, members observe others' behavior and exchange information with others. For example, after person A teaches person B a technical solution, they both know that the other party has that bit of knowledge. Group members also communicate with each other about third parties. For example, person A may tell person B about his or her observation of person C's skills. Person B then gains this piece of knowledge and can communicate it to other people.

Agent Actions

Based on their attributes, agents are able to take actions, such as learning, searching for resources, and exchanging information. In the model, tasks symbolize generic "things to do" and arrive randomly with different resource requirements. Group tasks are assigned to individual agents based on the Assignment matrix (P × T). Each agent is responsible for a set of subtasks and works on these subtasks one at a time. During the process of task performance, agents can be involved in three types of individual actions: learning, forgetting, and resource searching; and three types of group actions: communication, coordination, and decision making.

Individual Action I: Learning. Before a group starts performing its tasks, group members can go through training to learn knowledge or skills. During the training session, either a supervisor or an experimenter-type agent demonstrates how to perform each subtask, and group members gain access to the knowledge, skills, or resources required to perform these tasks. Group members can be trained as a group or individually. Depending on the experimental manipulation and how the group is initially structured, group members may become experts in different domains, and specialization may emerge in the group. Group members also learn by interacting with other members of the group, especially when they approach others to seek knowledge or resources.

Individual Action II: Forgetting. Human beings forget because of a variety of reasons. A human being's memory consists of two parts: long-term memory and short-term memory (Newell and Simon 1972). In the process of learning, new knowledge is first stored in short-term memory. After the knowledge is repeated for a number of times, it is stored into long-term memory using an index structure. Every time a piece of knowledge is recalled, the linkage between the index and the knowledge is reinforced. However, if a piece of knowledge is not accessed for a long time, the linkage might become weak and even disappear. That is typically when and how individual forgetting happens. In our model, we assume that (1) a piece of knowledge gets forgotten if it has not been recalled for a fixed number of time periods, (2) different types of knowledge have different decay rates, for example, procedural knowledge has been shown to be less subject to decay than declarative knowledge (Cohen and Bacdayan 1996). This phenomenon is captured in the concept of knowledge volatility. More volatile knowledge has higher decay rates and can be forgotten in a shorter period of time. If a piece of knowledge has not been recalled for such a long time that no member has access to it, organizational forgetting happens. The forgotten knowledge is thrown into a "knowledge trash can" that simulates knowledge stored in the form of physical products, documents, and information systems (Argote 1999). This knowledge is retrievable, but to a lesser extent, as compared with knowledge in human beings' brains.

Individual Action III: Resource Searching. To perform their tasks, agents need access to resources such as equipment, materials, or technical knowledge, while they may need to search for these resources within the group. Even if agents have the required resources, they can still choose to improve their skills or gain more resources by seeking help from others. If TM does not exist, agents search for resources by randomly asking others until they find the resources or have queried every member of the group. If TM exists, instead of searching aimlessly, agents mine their TM and approach those whom they think are more likely to have the required resources. Knowledge transfer is influenced by a variety of factors, such as the recipient's absorptive capacity (Cohen and Levinthal 1990) and characteristics of the knowledge and of the source (see Argote et al. 2003 for a review). In ORGMEM, we assume that how much knowledge a recipient can absorb from a source is inversely proportional to the difficulty level of the knowledge and directly proportional to the recipient's knowledge level and the source's knowledge level (see Equation 1 in the appendix).

Group Action I: Communication. Communication plays a key role in how knowledge is learned and retrieved in TM systems (Hollingshead 1998). In ORGMEM, communication is modeled as the process through which people share and exchange knowledge. The model assumes sufficient proximity among group members that communication is possible. Communication can be based on three mechanisms: random, relative similarity, and information seeking. Relative similarity models the process by which people talk to those who are similar to them or have knowledge in common with them; information seeking models the process by which people seek new knowledge by approaching people from different domains (Carley 1990). The likelihood of two agents interacting is calculated by comparing their TM and resource structure. Driven by relative similarity (information seeking), agent i is more likely to interact with those who are linked to people, resources, and tasks that are similar to (different from) those agent i is linked to (see Equations 2 and 3 in the appendix).

Group Action II: Coordination. TM has also been shown to facilitate task coordination in groups (Liang et al. 1995). By knowing each other's strengths and weaknesses, group members can make task assignments and cover each other's weak spots. Without the knowledge or shared consensus of who is good at doing what, group members would have to rely heavily on oral communication and negotiation to settle on an acceptable arrangement. In ORGMEM, we model task coordination as a process through which members assign and reassign tasks among themselves based on their knowledge of who is good at doing what. To assure fairness and a reasonable amount of work load for everyone, the agent who hands over a task to another agent is required to take over a task from the other party that this agent performs at least as well.

Group Actions III: Decision Making. During task performance, each member works on a set of subtasks assigned in the Assignment matrix ($P \times T$) by combining personal resources and resources acquired through within-group search. Members then integrate individual decisions to make a group decision. In this paper, we focus primarily on teams whose members have an equal say in group decision making. The only social cues that members utilize to weigh individual decisions are their perceptions of others' expertise. Members evaluate each other's expertise using a trust function. The $P \times R$ matrix in a person's TM indicates the skill level of every member in the group, represented by an integer falling in the range of $[0, 9]$. During group decision making, group members first construct a shared view of how knowledgeable each member is in performing his or her tasks. A trust coefficient is then derived from the shared view to determine how much weight to assign to each member's decision to come up with a group decision.

Model Implementation

ORGMEM is implemented using an object-oriented programming language, Java, because it provides the compatibility to run the model on different platforms and the flexibility to add new components. During each simulation, the simulated group begins with a particular design, undergoes a training session in which members perform a fixed number of tasks, and then operates through a working session during which members perform a designated number of tasks. A task is characterized by an N-bit binary string corresponding to N subtasks, and each bit indicates whether performing a task requires specific resources or not. If a subtask requires specific resources, the agent refers to the Needs matrix ($R \times T$) to find out which resources are required. The tasks in ORGMEM are quasi-repetitive in nature—members work on the same type of problems over time but some of the information, constraints, or parameters differ across tasks. For instance, a programming group whose primary task is coding software programs may work on different sections of the code using different tools or languages over time.

The initial state of the simulated group can be set to involve a specific number of agents, resources, and tasks. The initial relationships among agents, resources, and tasks can be left blank, randomly generated, or fully specified. Similarly, an agent's TM can be set to blank randomly guessed or fully specified. During both the training and working sessions, group members have opportunities to observe and communicate about each other's expertise. Group members jointly work on the task by searching for resources and applying these resources to make decisions. At the same time, group members communicate in pairs and exchange knowledge about who knows what. Task coordination happens at a specified frequency, every ten tasks, for instance, and knowledge that has not been accessed for long enough automatically decays out of group members' memory. During the working session, group performance and TM measures are recorded for each task and then averaged across the total number of tasks that a group performed.

Group Performance Measures

Group performance is measured by two variables used in previous studies: (1) time taken to finish group tasks and (2) quality of group operation or decision (Decker 1998, Liang et al. 1995). Timing is a crucial factor in organizational operation and decision making. Computationally, time is measured by counting the number of time periods elapsed from the initiation of a task to its completion. Quality is another key dimension for assessing group performance. In ORGMEM, quality is constructed with flexibility to capture different aspects of quality such as final product quality or group decision quality. We assume that a member's competence, measured as his or her resource set, determines the quality of his or her actions. We measure individual performance as match closeness between the requirements of a member's assignments and his or her resource set. Individual performance quality is a function of a member's working knowledge and how effectively he or she engages in resource searching, communicating, and group coordination. If a member has full knowledge to perform his or her assigned tasks, he or she receives the highest quality score. If a member has little of the knowledge required by his or her tasks, he or she receives a very low score. A group quality index is then calculated as a weighted average of individual performance. How much weight is assigned to each member is a function of how knowledgeable the group thinks that person is. If the group agrees that a member has adequate (or few) resources to perform his or her tasks, a high (or low) weight will be assigned. If the group has divergent opinions, a moderate weight will be assigned. Overall, group performance quality is jointly determined by resources available to the group members to perform their tasks and by how effectively the group members combine and integrate their resources and individual outcomes (Kunz et al. 1998).

TM Measures

TM density measures how much useful knowledge exists in TM. It is calculated by dividing the actual number of nonzero cells in an agent's TM matrix by the maximal possible number of nonzero cells. Nonzero information is useful in the sense that it indicates either there is a link or there is not a link between two elements. Group TM density can range from 0 when everyone knows nothing about others' knowledge to 1 when everyone in the group has complete knowledge of what others know. TM accuracy measures the percentage of accurate knowledge in TM. Inaccurate knowledge may come from several sources, one of which is out-of-date knowledge. The fact that agent i has access to knowledge k may be true at moment t, but not true after time $t+1$ if agent i

has "forgotten" k. Not being aware of this change, other agents may continue to regard agent i as the expert of knowledge k, and their knowledge about agent i's expertise in k becomes inaccurate. Computationally, TM accuracy is calculated by dividing the number of accurate nonzero cells by the total number of nonzero cells.

Model Validation

Computational models need to be validated before they can be applied to generate hypotheses or test theory (Carley 1996). The primary focus of model validation is to demonstrate the comparability of the simulated world in the computational model and the real world. We calibrated our model using data from a laboratory experiment using the radio assembly paradigm to study the role of TM in group training and group performance (Liang et al. 1995). We chose the radio assembly experimental setting because it is a well-developed paradigm, and has a reasonable level of comparability with the virtual experimental setting in ORGMEM. In both settings, group members can gain access to knowledge or resources required to perform group tasks. Group performance can be measured by the time that a group takes to complete its tasks and the quality of its performance. Group performance primarily depends on the knowledge or resources that members possess. Members gain knowledge of who knows what through observation and interaction, and TM density and accuracy are measured.

The radio assembly experiment consisted of two sessions: a training session and a testing session that happened a week apart. During the training session, an experimenter trained participants, either individually or in three-person groups, to assemble a radio. During the testing session, participants were instructed to assemble a radio together as fast as they could in groups—those trained in groups stayed in their original groups and those trained individually were randomly assigned to groups. The primary difference between the individual and group training condition was that members trained in groups had the opportunity to observe each other's behavior and communicate with one another, whereas participants trained individually did not have the opportunity. The main findings were (1) groups whose members trained together developed more complex and accurate TM, (2) groups whose members trained together did not take significantly more or less time to assemble the radios, (3) groups whose members trained together produced significantly better quality radios, (4) TM mediated the relationship between group training and group performance quality (Liang et al. 1995).

We randomly constructed 60 three-person groups with five resources and five subtasks to simulate the laboratory experimental setting. At the beginning of the simulation, group members' initial knowledge state was set blank and members were randomly assigned to the five subtasks. The groups then went through a training period of 50 tasks and a testing period of 100 tasks. Each group was simulated twice: once under individual training and once under group training.

Table 1 shows the results of paired t-tests of the differences in TM density and accuracy, time taken for groups to finish their tasks, and their performance quality between group training and individual training conditions. Consistent with previous laboratory studies, our virtual experimental results suggested that (1) group training helped to develop significantly more complex TM ($p < 0.001$), (2) group training helped to develop significantly more accurate TM ($p < 0.001$), (3) groups trained together took less time to finish their tasks although the difference was not significant ($p = 0.276$), and (4) groups trained together outperformed groups trained individually by making better quality decisions or products ($p < 0.001$).

To test the mediation effect of TM, we followed the procedures suggested by Baron and Kenny (1986). Because of the high correlation between TM density and accuracy (0.843, $p < 0.001$), we constructed a TM index by taking the average of the two. Because group training had no significant effect on the time taken for a group to accomplish its tasks, we focused exclusively on performance quality to test the mediation role of TM. The first equation, regressing quality (Q) on training condition (T), $Q = 0.113 + 0.022T$, was significant, $F(1, 118) = 75.19$, $P < 0.001$. Groups trained together performed better than groups trained individually. The second equation, regressing TM index (TMI) on training condition (T), $TMI = 0.621 + 0.231T$, was significant as well, $F(1, 118) = 1{,}222.67$, $P < 0.001$. Groups trained together developed both more complex and more accurate TM. The third equation, regressing quality (Q) on both training condition (T) and TM indexes (TMI), $Q = 0.048 - 0.002T + 0.104TMI$, was also significant, $F(2, 117) = 44.86$, $P < 0.001$. The coefficient for TM (TMI) was significant,

Table 1 **TM and Performance as a Function of Training Conditions**

Variables	N	df	Difference	t value	P value
TM density	60	59	0.265	38.87	<0.001
TM accuracy	60	59	0.198	18.80	<0.001
Time	60	59	−0.218	−1.10	0.276
Quality	60	59	0.022	8.93	<0.001

Note. The difference equals to (Group Training Condition − Individual Training Condition).

$t = 3.04$, $p < 0.01$, but the coefficient for training condition (T) was not significant, $t = -0.25$. This confirmed that TM mediated the effects of group training on group performance. In summary, our results from the computational model corresponded nicely to previously published laboratory results, indicating that ORGMEM is a valid tool to study TM-related phenomena in groups.

The Contingent Effects of TM

Although our results on three-person groups did not show a beneficial effect of TM on time, a previous study using larger groups showed that TM significantly decreased time taken for groups to finish their tasks (Ren 2001). The discrepancy between the two studies suggests that the effects of TM are contingent on group size. According to contingency theory (Galbraith 1973), there is no one best way to organize. The best way to organize is contingent upon the uncertainty and diversity of the basic tasks being performed by the organizational unit (Argote 1982, Duncan 1972). Accordingly, we expect that TM is not equally beneficial to all types of groups.

We chose three factors that have been studied in the literature: group size, task volatility, and knowledge volatility (So and Durfee 1998, Steiner 1972). We selected group size because search and coordination costs increase with size (Steiner 1972). Thus we expect that the benefits of TM might be more pronounced for groups that consist of many rather than few members. We varied group size from 3 to 35 using a rough interval of 6 (i.e., group size as 3, 9, 15, 21, 27, and 35).

The second contingency factor is task volatility. Task volatility reflects how frequently a group changes its tasks. We expect that TM might be more beneficial when tasks change than when they are stable. When members switch to a new task, it is less likely that they already possess the knowledge and skills to successfully complete the task. Thus, knowing whom to consult for advice is likely to be especially beneficial under changing task conditions. Three task conditions are simulated: never change, switch, and oscillate. We designed two sets of tasks that involve solving similar problems using different resources (e.g., software development using C versus software development using Java). Under the never change condition, groups perform one task throughout the experiment. Under the switch setting, groups change from task 1 to task 2 half way through the experiment. Under the oscillate condition, groups alternate constantly between task 1 and task 2.

The third contingency factor was knowledge volatility. Knowledge volatility reflects the decay rates of the knowledge required by group tasks. The more volatile the environment is, the sooner group members forget knowledge that has not been recently utilized. As knowledge volatility increases, group members are more likely to forget knowledge they possess. Knowing whom to ask for knowledge thus might be especially helpful when knowledge volatility is high. Three knowledge conditions are simulated: low, medium, and high. Under low volatility, no forgetting happens once an agent learns. Under medium volatility, knowledge gets forgotten if it has not been accessed for a large number of time periods. Under high volatility, knowledge gets forgotten if it has not been accessed for a small number of time periods. For simplicity, we examined groups that start with either blank TM (e.g., group members are strangers) or full TM (e.g., group members know everyone's areas of expertise). Altogether, there were $6 \times 3 \times 3 \times 2 = 108$ conditions.

We assume that there exist nine resources and twelve tasks in each group. Under each experimental condition, nine groups were constructed by randomly assigning four tasks and a fixed number of resources to group members. We referred to previous studies that simulated organizational structure and performance and picked the most commonly used values to set these values (Lin 1994). Because many groups in real organizations consist of members who join with knowledge and experience and do not need training, we skipped the training session in the contingency experiments. Instead, group members were randomly assigned one, three, or six pieces of resources in the beginning. The one resource condition simulates groups whose members join with little, and therefore rarely overlapping task-relevant knowledge as in a jury. The three resource condition simulates groups whose members join with moderate, and therefore somewhat overlapping task-relevant knowledge as in a software development team. The six resource condition simulates groups whose members join with extensive, and therefore heavily overlapping task-relevant knowledge as in a research project team. Once a group was created, members interacted, coordinated, and made decisions to perform 100 tasks. The time a group took to finish its tasks and the quality of its performance were recorded and averaged across the 100 tasks.

Time as the Dependent Variable

As shown in Figure 3, groups with TM took less time to finish their tasks regardless of group size. Further, as groups got larger, TM seemed to be more beneficial. A two-way analysis of variance (ANOVA) (see Table 2) revealed a significant main effect for TM, $F(1, 960) = 86.72$; $p < 0.001$, a significant main effect for group size, $F(5, 960) = 20.02$; $p < 0.001$, and a significant interaction effect of the two, $F(5, 960) = 2.51$; $p < 0.05$. The Tukey test indicated that groups with 27 or 35 members suffered the most from the lack

Figure 3 Time Taken to Finish Group Tasks Under Different Group Sizes

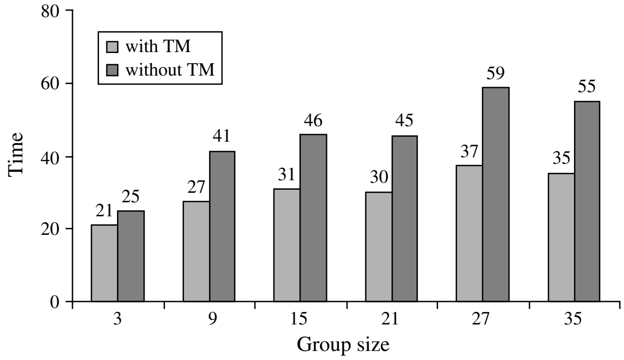


Table 2 ANOVA Summary of the Relationship Between TM, Group Size, and Time Taken to Finish Group Tasks

Source	*df*	*SS*	*MS*	*F*
TM exist (A)	1	53,799	53,799.38	86.72***
Group size (B)	5	62,114	12,422.71	20.02***
A × B interaction	5	7,793	1,558.54	2.51*
Error	960	595,560	620.38	
Total	971	719,266		

*$p < 0.05$, **$p < 0.01$, ***$p < 0.001$.

of TM, followed by groups of 9–21 members, and followed by groups with 3 members (Hatcher and Stepanski 1994).[1]

Similarly, the results in Figures 4 and 5 indicated that groups in different task or knowledge environments benefited differently from TM. Although groups with TM took less time to perform their tasks in all types of environments, performing in a comparatively more dynamic task environment or volatile knowledge environment, without TM, seemed to be more challenging. A three-way ANOVA analysis (see Table 3) revealed a significant main effect for TM, $F(1, 954) = 131.16$; $p < 0.001$, a significant main effect for task environment, $F(2, 954) = 86.41$; $p < 0.001$, and a significant main effect for knowledge environment, $F(2, 954) = 138.50$; $p < 0.001$. Also, Table 3 revealed a significant interaction effect between TM and task environment, $F(2, 954) = 7.93$; $p < 0.01$ and a significant interaction effect between TM and knowledge environment, $F(2, 954) = 29.28$; $p < 0.001$. The Tukey test showed that groups that constantly changed tasks suffered the most from the lack of TM, and groups that occasionally changed tasks suffered significantly less than groups that constantly changed tasks but significantly more than groups that never changed tasks. The Tukey tests showed no significant difference between groups under low and medium knowledge volatility, and both types of groups suffered significantly less from the lack of TM than groups under high knowledge volatility.

As shown in Table 3, the analysis revealed a significant three-way interaction between TM, task environment, and knowledge environment, $F(4, 954) = 2.59$; $p < 0.05$. Groups in a double volatile environment (high task volatility and high knowledge volatility) suffered the most from the lack of TM. As shown in Figures 6 and 7, TM is especially crucial to groups

[1] We tried an additional condition with a group size of 45. The results suggested that the effect on group size leveled off instead of dropping. Groups with a size of 45 took more time, although not significantly more time, than 35-person groups to finish their tasks both with (Mean Time = 38.31) and without (Mean Time = 61.92) TM.

Figure 4 Time Taken to Finish Group Tasks Under Different Task Environments

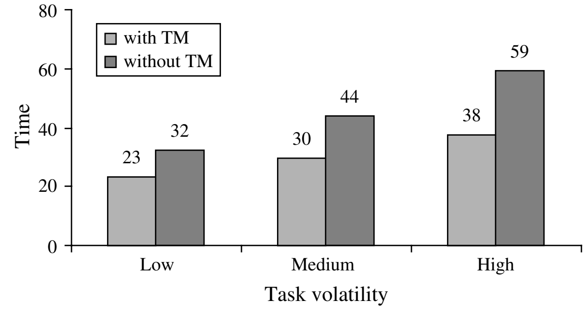


Figure 5 Time Taken to Finish Group Tasks Under Different Knowledge Environments

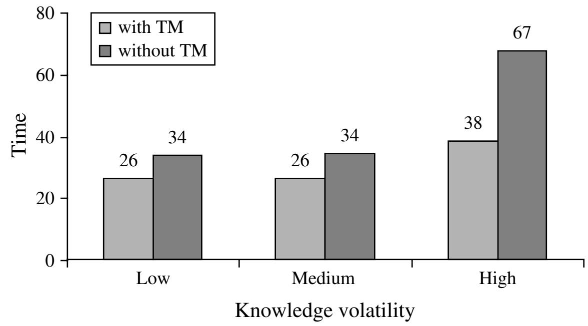


Table 3 ANOVA Summary of the Relationship Between TM, Task Environment, Knowledge Environment, and Time

Source	*df*	*SS*	*MS*	*F*
TM exist (A)	1	53,799	53,799.38	131.16***
Task environment (B)	2	70,889	35,444.48	86.41***
Knowledge environment (C)	2	113,618	56,809.19	138.50***
A × B interaction	2	6,502	3,250.76	7.93**
A × C interaction	2	24,017	12,008.61	29.28***
B × C interaction	4	54,894	13,723.45	33.46***
A × B × C interaction	4	4,246	1,061.48	2.59*
Error	954	391,301	410.17	
Total	971	719,266		

*$p < 0.05$, **$p < 0.01$, ***$p < 0.001$.

Figure 6 Three-Way Interaction Between TM and Task and Knowledge Environments (Without TM)

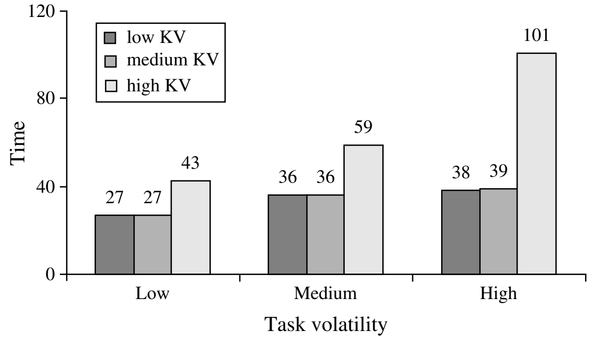
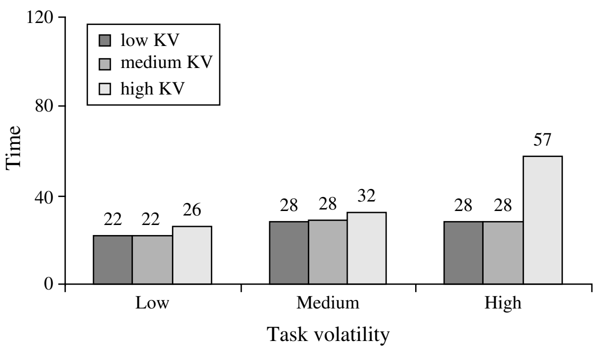


Figure 7 Three-Way Interaction Between TM and Task and Knowledge Environments (with TM)



that constantly shift among multiple tasks or projects that require different and easy-to-forget knowledge.

Quality as the Dependent Variable

A two-way ANOVA (see Table 4) of group performance quality revealed a significant main effect for TM, $F(1, 960) = 28.93$; $p < 0.001$, a significant main effect for group size, $F(5, 960) = 99.15$; $p < 0.001$, and a significant interaction effect of the two, $F(5, 960) = 3.71$; $p < 0.01$. The Tukey test showed that groups with fewer than 15 members benefited more from their TM systems than larger groups. As shown in Figure 8, although larger groups tended to perform better than smaller groups, smaller groups with the assistance of TM approached the performance quality of larger groups.

Table 4 ANOVA Summary of the Relationship Between TM, Group Size, and Quality

Source	*df*	*SS*	*MS*	*F*
TM exist (A)	1	0.40	0.40	28.93***
Group size (B)	5	6.84	1.37	99.15***
A × B interaction	5	0.26	0.05	3.71**
Error	960	13.24	0.01	
Total	971	20.73		

$^*p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

Figure 8 Group Performance Quality Under Different Group Sizes

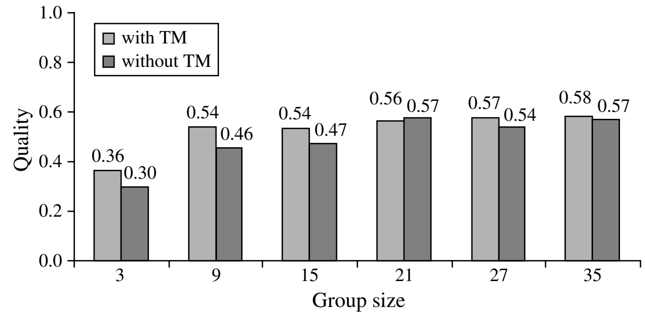


Figure 9 Group Performance Quality Under Different Task Environments

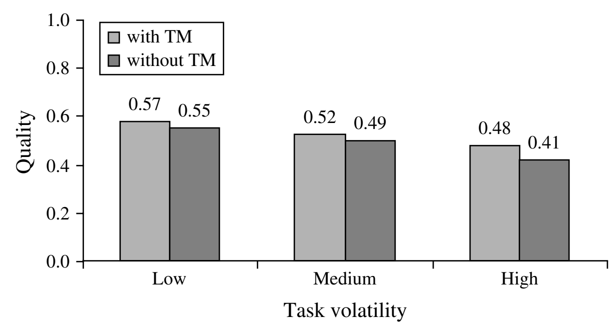
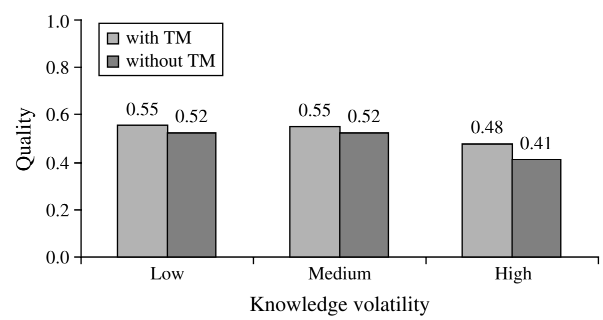


Figure 10 Group Performance Quality Under Different Knowledge Environments



Interestingly, contingent effects were not found for quality with groups in different task or knowledge environments, as shown in Figures 9 and 10. A three-way ANOVA (see Table 5) revealed a significant main effect for TM, $F(1, 954) = 25.18$; $p < 0.001$, a significant main effect for task environment, $F(2, 954) = 66.78$; $p < 0.001$, a significant main effect for knowledge environment, $F(2, 954) = 57.05$; $p < 0.001$, but no significant interactions between TM and either of the group environment variables. The results indicated that TM improved group performance quality across all conditions. Groups in a comparatively stable environment (less task switching and less knowledge forgetting) performed better regardless of whether they had a TM system or not.

Table 5 ANOVA Summary of the Relationship Between TM, Task Environment, Knowledge Environment, and Quality

Source	*df*	*SS*	*MS*	*F*
TM exist (A)	1	0.40	0.40	25.18***
Task environment (B)	2	2.12	1.06	66.78***
Knowledge environment (C)	2	1.81	0.90	57.05***
$A \times B$ interaction	2	0.07	0.04	2.22
$A \times C$ interaction	2	0.07	0.03	2.13
$B \times C$ interaction	4	1.15	0.29	18.20***
$A \times B \times C$ interaction	4	0.01	0.01	0.06
Error	954	15.11	0.02	
Total	971	20.73		

$^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.

Discussion

In this paper, we designed and implemented a computational model, ORGMEM, and applied it to explore the relationships between TM and group performance. Our virtual experimental results validate the model as a useful tool for studying TM in groups and correspond to previous findings on the positive effects of TM on group performance. Our results suggest that the effects of TM are contingent upon factors such as group size, task environment, knowledge environment, and the performance measure used. In terms of time that groups take to finish their tasks, larger groups, groups in a dynamic task environment, and groups in a volatile knowledge environment benefit more from knowing what others know than smaller groups and groups in more stable environments. In terms of performance quality, smaller groups benefit more from knowing what others know than larger groups.

Similar to other computational models, ORGMEM faces the challenge of balancing transparency (the extent to which it is clear how the model works) and veridicality (the extent to which the model works like the real world) (Carley 2002). As an initial attempt to study TM using computational modeling techniques, we started with a simple model. As a result, the model might not have captured all the interesting aspects of TM in reality, such as subjective judgment of others' knowledge based on category membership and task or knowledge specialization. Another limitation is that we simulate only flat team structures, in which all members are at the same level and there are no geographic or administrative barriers that inhibit interpersonal communication. Also, only two extreme states of TM are simulated in this paper: no TM versus full TM. We plan to gradually relax these assumptions and examine the effects of different TM structures in our future research.

This paper provides useful information for both scholars and practitioners. First, our virtual experiments examining different group settings imply that well-developed TM systems are more beneficial to some groups than others. Larger groups suffer more in terms of time from not having a TM system than groups of smaller sizes. The more people consulted, the longer time it takes. By contrast, only a few group members need to be queried in small groups as compared to large groups to locate a piece of knowledge. As an extreme case, in three-person groups, the time taken to search for specific knowledge is so trivial that even if all group members are queried, the time is inconsequential. Thus our results suggest that managers should pay more attention to foster TM in large groups if time is a critical factor by promoting interaction opportunities for employees to get to know each other and build ties through which they can acquire resources later.

Second, the virtual experimental results suggest that groups in a dynamic task environment take significantly longer to complete their tasks because of the lack of TM than groups in a comparatively stable environment. When group tasks change more frequently, group members need to search for resources more frequently. Our robustness check on the effect of task volatility suggests that although the effect of high task volatility (switching tasks constantly) is robust across all conditions, the effect of medium task volatility (changing tasks once in the middle) is partially determined by the direction of the change. Groups take less time to adjust to new tasks when they change from complicated tasks that require extensive resources to less resource-demanding tasks than when they change in the other direction. Similarly, our results suggest that groups in a volatile knowledge environment suffer more from not having TM than groups in a comparatively stable environment. When knowledge is less volatile, group members do not need to search for resources as frequently as they would in a more volatile environment.

Third, the three-way interaction effects of TM, task environment, and knowledge environment on time suggest that TM is most beneficial to groups in a double volatile environment, such as software development teams. Software development typically involves development cycles, in which different pieces of software are designed, implemented, and tested using different tools and applications. As the software gets scaled and complicated, the knowledge associated with its development becomes so detailed and complex that it can be easily forgotten. Very often programmers complain about having to go back to the original documentation or co-workers to understand a piece of code only two months after they worked on it. Our results imply that TM could play an important role in determining the performance of software development teams, which corresponds to previous findings from the field (e.g., Faraj and Sproull 2000).

Another interesting finding is the general lack of contingent effects of TM on group performance quality. Our results suggest that both TM and being in a stable environment lead to better performance. In contrast to our finding that TM is more beneficial to large groups than small groups in shortening performance time, our analyses using quality as the dependent measure suggest the opposite—knowing others' expertise helps small groups more than large groups make better quality decisions. These discrepancies may be the result of somewhat different mechanisms through which TM affects the two dimensions of group performance. Time is primarily a function of the resource searching process. Quality depends on both individual activity (resource searching) and group activities (task coordination and group decision making), especially the latter. While TM facilitates resource searching in both large and small groups, large groups suffer more from its absence because more extensive searches are required to locate a piece of knowledge. Concerning quality, large groups may benefit less from TM processes such as task coordination and expertise evaluation because they have greater difficulty in obtaining and maintaining an up-to-date shared view of expertise distribution than small groups.

In conclusion, our virtual experimental results suggest that group size and environment matter in understanding the effects of TM on group performance. Although TM, in general, improves group performance, it benefits different types of groups along different dimensions. If the primary goal is to get work done as quickly as possible, TM is more beneficial to large groups, groups in a highly dynamic task environment, and groups in a highly volatile knowledge environment. If the primary goal is to make optimal decisions, TM is more beneficial to small groups. Thus, managers should not only promote the idea of knowing what others know and provide opportunities for gaining the knowledge, but also align their policies or practices with specific organizational goals, processes, and circumstances.

Acknowledgments

This work was supported by the National Science Foundation KDI IIS 9980109, the U.S. Army Research Institute for the Behavioral and Social Sciences, "Personnel Turnover and Team Performance" (L. Argote, J. Levine, K. Carley, and R. Moreland) 2000-2003, the Center of Computational Analysis of Social and Organizational Systems (CASOS), and the Center for Organizational Innovation and Learning at Carnegie Mellon University. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the National Science Foundation or U.S. Army Research Institute. The authors thank Mark Fichman, Sara Kiesler, Susan Fussell for helpful comments, and Nicole Lazar for statistical suggestions.

Appendix

Knowledge Transfer. Let agent i's knowledge in domain r at time (t) be denoted by $S_{ir}(t)$, agent j's knowledge in domain r at time (t) be $S_{jr}(t)$, and the maximum knowledge in domain r be M_r. Let the difficult level of domain r be denoted by α_r. What agent i knows at time $(t+1)$ can be calculated as

$$S_{ir}(t+1) = S_{ir}(t) + \alpha_r * S_{jr}(t) * S_{ir}(t) \quad \text{s.t.} \quad 0 \le S_{ir}(t) \le M_r \text{ and } 0 \le \alpha_r \le 1. \tag{1}$$

Communication Probability. Let $S_{ir}(t)$ be agent i's knowledge in domain r and $S_{jr}(t)$ be agent j's knowledge in domain r, $RS_{ij}(t)$, the probability that agent i will interact with agent j based on relative similarity, can be calculated as

$$RS_{ij}(t) = \frac{\sum_{r=1}^{R} \min(S_{ir}(t), S_{jr}(t))}{\sum_{k=1}^{I} \sum_{r=1}^{R} \min(S_{ir}(t), S_{kr}(t))} \quad \text{s.t.} \quad 0 \le RS_{ij}(t) \le 1. \tag{2}$$

The probability that agent i will interact with agent j based on information seeking, IS_{ij}, can be calculated by dividing the relative expertise of agent j compared to agent i with the sum of relative expertise of everyone else in the group compared to agent i.

$$IS_{ij}(t) = \frac{\sum_{r=1}^{R} (S_{ir}(t) = 0 \ \&\ S_{jr}(t) \ne 0)}{\sum_{k=1}^{I} \sum_{r=1}^{R} (S_{ir}(t) = 0 \ \&\ S_{jr}(t) \ne 0)} \quad \text{s.t.} \quad 0 \le IS_{ij}(t) \le 1. \tag{3}$$

References

Argote, L. 1982. Input uncertainty and organizational coordination in hospital emergency units. *Admin. Sci. Quart.* **27**(3) 420–434.

Argote, L. 1999. *Organizational Learning: Creating, Retaining and Transferring Knowledge*. Kluwer, Norwell, MA.

Argote, L., W. McEvily, R. Reagans. 2003. Managing knowledge in organizations: An integrative framework and review of emerging themes. *Management Sci.* **49**(4) 571–582.

Austin, J. 2003. Transactive memory in organizational groups: The effects of content, consensus, specialization, and accuracy in group performance. *J. Appl. Psych.* **88**(5) 866–878.

Baron, R. M., D. A. Kenny. 1986. The moderator-mediator variable distinction in social psychological research: Conceptual, strategic, and statistical considerations. *J. Personality Social Psych.* **51** 1173–1182.

Carley, K. 1990. Group stability: A socio-cognitive approach. *Adv. Group Processes* **7** 1–44.

Carley, K. M. 1996. Validating computational models. Working paper, Carnegie Mellon University, Pittsburgh, PA.

Carley, K. M. 2002. Simulating society: The tension between transparency and veridicality. *Agent 2002 Conf.*, Chicago, IL.

Carley, K. M., M. J. Prietula. 1994. Computational organization theory. K. M. Carley, M. J. Prietula, eds. *Computational Organization Theory*. Lawrence Erlbaum Associates, Hillsdale, NJ, xi–xvii.

Carley, K. M., Y. Ren, D. Krackhardt. 2000. Measuring and modeling change in C3I architectures. *Internat. Sympos. Command Control Res. Tech.*, Monterey, CA.

Cohen, M. D., P. Bacdayan. 1996. Organizational routines are stored as procedural memory: Evidence from a laboratory study. M. D. Cohen, L. S. Sproull, eds. *Organizational Learning*, Vol. 18. Sage Publications, Thousand Oaks, CA, 403–429.

Cohen, S. G., D. E. Bailey. 1997. What makes teams work: Group effectiveness research from the shop floor to the executive suite. *J. Management* **23**(3) 239–290.

Cohen, W. M., D. A. Levinthal. 1990. Absorptive capacity: A new perspective on learning and innovation. *Admin. Sci. Quart.* **35** 128–152.

Decker, K. S. 1998. Task environment centered simulation. M. J. Prietula, K. M. Carley, L. Gasser, eds. *Simulating Organizations: Computational Models of Institution and Groups*, Vol. 6. AAAI Press, Menlo Park, CA.

Duncan, R. B. 1972. Characteristics of organizational environments and perceived environmental uncertainty. *Admin. Sci. Quart.* **17**(3) 313–327.

Faraj, S., L. Sproull. 2000. Coordinating expertise in software development teams. *Management Sci.* **46**(12) 1554–1568.

Galbraith, J. 1973. *Designing Complex Organizations*. Addison-Wesley, Reading, MA.

Grant, R. M. 1996. Toward a knowledge-based theory of the firm. *Organ. Sci.* **17**(Special Issue) 109–122.

Hatcher, L., E. J. Stepanski. 1994. *A Step-by-Step Approach to Using the SAS System for Univariate and Multivariate Statistics*, Vol. 6. SAS Publishing, Cary, NC.

Henry, R. A. 1995. Improving group judgment accuracy: Information sharing and determining the best member. *Organ. Behavior Human Decision Processes* **62**(2) 190–197.

Hollenbeck, J. R., D. R. Ilgen, D. J. Sego, J. Hedlund, D. A. Mafor, J. Philips. 1995. Multilevel theory of team decision making: Decision performance in teams incorporating distributed expertise. *J. Appl. Psych.* **80** 292–316.

Hollingshead, A. B. 1998. Communication, learning, and retrieval in transactive memory systems. *J. Experiment. Social Psych.* **34** 423–442.

Hulin, C., D. R. Ilgen. 2000. Introduction to computational modeling in organizations: The good that modeling does. D. R. Ilgen, C. L. Hulin, eds. *Computational Modeling of Behavior in Organizations*, Vol. 1. American Psychological Association, Washington, DC, 3–18.

Krackhardt, D., K. M. Carley. 1998. PCANS model of structure in organizations. *1998 Internat. Sympos. Command Control Res. Tech.*, Monterey, CA.

Krauss, R. M., S. R. Fussell. 1990. Mutual knowledge and communicative effectiveness. J. Galegher, R. E. Kraut, C. Egido, eds. *Intellectual Teamwork: Social and Technical Bases of Collaborative Work.* Erlbaum, Hillsdale, NJ.

Kraut, R., S. Slaughter, A. Espinosa. 2004. A challenge to group research: Using group theory to social engineering. *2004 Acad. Management Meeting*, New Orleans, LA.

Kunz, J. C., R. E. Levitt, Y. Jin. 1998. The virtual design team: A computational simulation model of project organizations. *Comm. Assoc. Comput. Mach.* **41**(11) 84–92.

Lewis, K. 2003. Measuring transactive memory systems in the field: Scale development and validation. *J. Appl. Psych.* **88**(4) 587–604.

Liang, D. W., R. Moreland, L. Argote. 1995. Group versus individual training and group performance: The mediating role of transactive memory. *Personality Soc. Psych. Bull.* **21**(4) 384–393.

Lin, Z. 1994. A theoretical evaluation of measures of organizational design: Interrelationship and performance predictability. K. M. Carley, M. J. Prietula, eds. *Computational Organization Theory*, Vol. 6. Lawrence Erlbaum Associates, Hillsdale, NJ, 113–160.

Littlepage, G. E., W. Robison, K. Reddington. 1997. Effects of task experience and group experience on group performance, member ability, and recognition of expertise. *Organ. Behavior Human Decision Processes* **69** 133–147.

Monge, P. R., N. S. Contractor. 2003. *Theories of Communication Networks*. Oxford University Press, Oxford, UK.

Moreland, R. L., L. Argote, R. Krishnan. 1996. Socially shared cognition at work: Transactive memory and group performance. J. L. Nye, A. M. Brower, eds. *What's Social About Social Cognition? Research on Socially Shared Cognition in Small Groups*. Sage Publications, Thousand Oaks, CA, 57–84.

Newell, A., H. A. Simon. 1972. *Human Problem Solving*. Prentice-Hall, Englewood Cliffs, NJ.

Ren, Y. 2001. Simulating the role of transactive memory in group performance. *4th Internat. Organ. Learn. Knowledge Management Conf.*, London, Canada.

So, Y., E. H. Durfee. 1998. Designing organizations for computational agents. M. J. Prietula, K. M. Carley, L. Gasser, eds. *Simulating Organizations: Computational Models of Institution and Groups*, Vol. 3. AAAI Press, Cambridge, MA, 47–66.

Steiner, I. D. 1972. *Group Process and Productivity*. Academic Press, New York.

Stewart, T. A. 1995. Mapping corporate brainpower. *Fortune* (October 30) 209–212.

Wegner, D. M. 1987. Transactive memory: A contemporary analysis of the group mind. B. Mullen, G. R. Goethals, eds. *Theories of Group Behavior*, Vol. 9. Springer-Verlag, New York, 185–208.

Wegner, D. M., R. Erber, P. Raymond. 1991. Transactive memory in close relationships. *J. Personality Social Psych.* **61**(6) 923–929.