GOMS, Distributed Cognition, And The Knowledge Structures Of Organizations

Robert L. West (rwest@hkucc.hku.hk)
Department of Psychology, University of Hong Kong, Hong Kong, SAR.

Alan Wong (H9352709@hkusua.hku.hk)
Department of Psychology, University of Hong Kong, Hong Kong, SAR.

Alonso H. Vera (vera@hkucc.hku.hk)
Department of Psychology, University of Hong Kong, Hong Kong, SAR.

Abstract

The idea that GOMS can be used to model HCI tasks within the organizational environment in which they occur is discussed and reviewed. An example in terms of satellite operations is provided.

Mantovani (1996) has proposed that the study of human computer interaction (HCI) is currently limited because we, "lack an integrated model of social context suitable for HCI research," (Mantovani, 1996). However, while it is true that social context has not often been explicitly addressed in the HCI literature, this does not mean that the modeling systems currently in use cannot accommodate social context. More specifically, we propose that GOMS can be used to model HCI tasks within the relevant contextual aspects of organizations, such as companies and institutions. To support this we review the relevant issues and present the initial results of a study incorporating this goal.

GOMS (Card, Moran, & Newell, 1983) is a modeling system designed to capture how experts execute well learned, routine tasks (e.g. word processing, satellite tracking). Essentially, it breaks down a task into Goals, Operators, Methods, and Selection rules (see John & Kieras, 1994 for a detailed review and discussion of the different variations of GOMS). Goals describe what the user wants to achieve; methods are combinations of sub-goals and operators used to achieve goals; selection rules are rules for selecting between methods; and operators are actions, either physical (e.g. move the mouse), perceptual (e.g. search the screen), or cognitive (e.g. add two numbers).

Although GOMS was originally developed to model individual humans interacting with computers, the unit of analysis is flexible. Thus a task performed by a room full of humans and computers could be described at the level of the interactions between the individual humans and computers, or at the level of the room, i.e. without reference to the specific agents involved. GOMS describes the knowledge level of the task. The granularity of the units involved depends on the goals of the researchers. Thus GOMS is equally suitable for describing the knowledge of an individual or the knowledge of a distributed cognitive system, such as described by Hutchins (1990).

As Olson and Olson (1990) note, the original formulation of GOMS had many limitations that have since been ameliorated. For example, the ability of GOMS to account for learning, errors, the limitations of working memory, and parallel processing have all been considerably improved (see Olson & Olson, 1990 for a review). What we are proposing is that organizations can be analyzed as distributed cognitive systems, and that GOMS can be used to describe the knowledge level of such systems, including the knowledge structures mediating task related social interactions (note, in this paper a social interaction is defined as any interaction between two or more people).

Organizations tend to involve a high amount of routine, well learned activity. In addition, we suggest that the social and cultural rules mediating interpersonal relationships within organizations are likewise routine in nature. Therefore, GOMS, which has a good track record for modeling routine behavior, seems an appropriate modeling choice. However, note, we are not suggesting that GOMS can be used for all social situations, e.g. GOMS may not be a good choice for modeling close personal relationships and the like.

In addition there are several important advantages to using GOMS. First, GOMS is a well established modeling tool with a host of studies demonstrating how to solve various specific problems. Second, modeling at the organizational level can entail a high degree of cross disciplinary work, involving diverse areas such as cognitive science, social psychology, game theory, sociology, and anthropology. Because GOMS is relatively easy to understand at the conceptual level, it is a good choice to serve as a common modeling language. Third, GOMS is currently used to create models of the types of systems commonly found in many organizations. If GOMS is used to model an organization's structure then the model will be compatible with existing GOMS models of specific sub-tasks within that organization. Fourth, GOMS can be used to address specific questions such as time estimates, the efficient use of resources, possible goal conflicts, the degree to which goals can be fulfilled, and whether or not an organization would be robust in the face of changing conditions.

In addition, this approach could also be applied to psychological and sociological questions, such as determining the nature of the employee environment (e.g. does it promote undue stress? under what conditions?) or the emergent functional properties of the organization (e.g. does it perpetuate racism?). The advantage of a GOMS model for these types of questions is that it can provide a process model of how such conditions arise, as well as how they feed back into the system.

Modeling Social Actions

In this section we consider the issue of modeling social actions, that is, the interactions between two or more people. From an individual's point of view, there are two broad types of social actions, 1) actions relating to other individuals, such as making the judgment, "do I trust this person?" and 2) actions relating to groups, such as making the judgment "will the market go up?"

In some cases, GOMS methods may be constructed to make these types of judgments. Such methods would reflect the knowledge level of the task. For example, to answer the question of whether the market will go up or down, an expert might execute the sub goals of gathering economic and political information. However, it may also be necessary to assume social operators for *gut level* decisions. For example, to answer the question, "what is the mood of the market?" an analyst may go with his or her *feeling* about it. Damasio's (1994) Somatic-Marker Hypothesis is a good candidate for understanding this type of process. Social operators can also be modeled using the mechanisms proposed in social psychology, combined with specific organizational, social and cultural knowledge. For example, Fishbein and Ajzen's (1974) reasoned action theory can be combined with real world data to predict attitudes towards alternative actions. Social operators could also be modeled using AI models, such as ACT-R (Anderson, 1993) or SOAR (Newell, 1990).

However, it is not always necessary to provide models of operators. As John (1995) notes, "Operators can be defined at many different levels of abstraction but most GOMS models define them at a concrete level, like button presses and menu selections." This is certainly useful since simple operations can be assumed to be performed correctly and within an approximate time span most of the time (Card et al, 1983), but operators needn't be limited to simple operations. In principle, any process can be made into an operator. For example, a model could be constructed in which an architect calls an operator to judge the aesthetic quality of a design. Thus by assuming a complex action exists as an operator it becomes possible to frame where and when it takes place within a GOMS model, as well as its functional significance. For example, the output of the aesthetic operator above could be used as the basis for a selection rule (e.g. if aesthetic, continue drawing; if not aesthetic, throw it away). The issue is one of finding the most useful level of operator abstraction for the task being modeled (Card et al, 1983).

It is also interesting to note that GOMS itself can be used as a model for certain types of social thought. Specifically, it can be argued that when one person wants to predict what another person will do in a particular situation, that they construct something very much like a GOMS model of the person in the situation and then mentally simulate it (e.g. see Kahneman & Tversky, 1982). In this sense people may be very similar to intelligent software agents that use GOMS models to predict the behavior of the user (e.g. Vera and Rosenblatt, 1995). If this is the case, then it follows that GOMS would be particularly appropriate for modeling this type of behavior.

The Importance of Goals

As Mantovani (1996) notes, people have socially and culturally based goals. Without considering such higher level goals, most GOMS models implicitly assume that the user's highest goal is to do the task as well as possible. To get at the organizational, social and cultural goals of a user it is necessary to ask why the employee wants to complete the task. In many cases the answer will be that the employee will benefit from doing it (i.e. they will be paid, they will avoid being fired). In this case the goal hierarchy need go no higher. However, in some cases, the higher level goal structure may be more complex. In particular when an employee is free to choose between various tasks it is necessary to understand his or her higher level goals in order to predict what he or she will do next. In addition, in the case of multiple employees, it may be important to understand how the higher level goal structure of an employee interacts with the higher level goal structures of other employees. This type of analysis could be approached from a game theory perspective (i.e. assume payoffs for achieving goals and that employees will act rationally) or a social cognition perspective (i.e. modeling based on questionnaires, observation, etc.).

Another interesting and related issue is determining the goals of an organization. Organizations have goals, for example, an environmental consulting firm may state their goal as being to create a clean environment. However, it is interesting to note that individuals can work within an organization without adopting the higher level goals of the organization. For example, a person could work for an environmental consulting firm simply for the money. The question as to whether the structure of an organization is such that the goals of the individuals result in the stated goal of the organization is an interesting and important one, especially for governmental and other public service organizations.

Also, as noted in the introduction, firms may have goals implicit in their structure. For example, sociologists often refer to institutionalized racism. The benefit of using a GOMS to examine this type of issue is that it can provide a highly specific process model of *why* various goals exist within a system. This is because goals are triggered by a clear chain of events involving higher level goals and selection rules.

The Top Down Effects of Goals

The manner in which higher level organizational, social or cultural goals can effect the behavior of lower level methods and operators is of critical importance. GOMS is modular in nature and thus based on the traditional cognitive science assumption that we can abstract simple problem spaces from a complex world and deal with them in relative isolation (Vera & Simon, 1993). Mantovani (1996), making the case for his interpretation of the role of social and cultural factors, argues that this is *never* the case, even for very simple actions such as tying shoes.

> when we wear shoes, we usually have some project in our mind regarding some activity which is relevant with respect to our current interests and requires wearing shoes. Thus tying laces and wearing shoes are simple activities which depend on actors' cultural models, for example, models of healthy behavior generate broader projects like keeping fit by running in the park in the morning. (Mantovani, 1996).

However, the traditional cognitive science perspective argues that simple actions such as shoe tying can be treated in isolation and that is it is unnecessary to understand the context shoe tying beyond the goal of wanting one's shoes to be tied. A person could tie his or her shoes as part of fulfilling many divergent goals (e.g. a run in the park, a night at the opera, or overthrowing a dictatorship) and the process would remain essentially the same. This is true in two senses. First, in terms of the measurable outcome the laces would be tied in approximately the same way each time. The results might be affected by factors such as time pressure and memory load (e.g. forgetting to tie ones shoes or tying them sloppily), but these are not direct effects of the social context, rather they are mediated by basic cognitive variables (e.g. memory and processing speed). The second sense in which this is true is in terms of the knowledge and motor skills deployed to achieve the result. Again, allowing for mediating factors such as memory load and time, we would expect the process of tying shoes to remain constant once it has been well learned.

Low level operators (such as clicking a mouse on an icon) will generally be unaffected by higher level social/cultural goals. However, in cases in which a low level operator is affected, the effect will be mediated by well defined, variables such as memory load, demands on attention, speed/accuracy tradeoffs, and so on. This type of effect, when it occurs, may be an important consideration in modeling the system. Specifically, if a low level operator error can cause a significant problem, then the factors that mediate the likelihood of such an error should be modeled in. An example of this approach is the use of GOMS to predict the effect of work load on working memory (Card et al., 1983).

In the case of social operators we argue that the same approach can be taken. That is, for the most part social operators will not be affected by higher level goals, but when they are it will be through the mediation of a limited number of variables. In fact, this is the approach used in Social Cognition. Cognitive variables, such as those mentioned above, can be used to predict mediators such as stress, which is a major determinant of social functioning (e.g. a social operator, such as, "behave politely towards the customer," could be severely disrupted by stress). Other socially mediating factors, such as threats to self esteem may also play an important role.

Thus operators are defined in terms of the general factors that could effect their operation, as well as in terms of how they effect the general factors. To keep track of this in a model, an index can be attached to each employee (or customer, or client) to record the general factors impinging on them at each step in the process. For example, an index could be used to keep track of memory load or stress due to time pressure. This type of approach could be useful in terms of modeling systems in which human error can cause serious results.

Spin-off Effects: An Example

Although we argue that lower level methods and operators can be treated in relative isolation from higher level goals, their appropriateness for fulfilling specific higher level goals is still a potentially important issue. Specifically, evaluating methods solely in terms of how they satisfy the immediate goal may not lead to the optimal solution. This is because methods can have spin-off effects which may be important for satisfying goals elsewhere in the system.

For example, one of the authors (RLW) is working on the HCI for an electronic Chinese-English dictionary designed for those who cannot read Chinese. Initially the highest level goal in the model for this task was to look up a Chinese character and find its meaning in English. This generated a lot of high tech suggestions, such as scanning the character and using a neural network to identify it. However, by identifying the higher level goal of the intended users, which in this case was to learn to read Chinese, it was realized that some form of the traditional system (the radical search method), would probably be better. This is because the traditional system requires the user to parse the characters into their meaningful, pictographic components, which studies show play an important role in character recognition. Thus a process for learning the structure of Chinese characters is situated in the process of looking them up in the dictionary.

Of course, there are always ever higher goals being generated by ever higher system structures. Obviously we cannot consider every possibility in a model. However, we have a better chance of finding spin-off effects within an organization if we possess a model of the organization. With a detailed model, spin off effects can be located through simulating different versions of the model (e.g. inserting different HCI structures for specific tasks), or simply by studying the model.

Reactivity to the Environment

Traditionally, GOMS models have assumed a pristine task environment, one in which interruptions unrelated to the specific task being modeled do not occur. However, in a social setting (i.e. within an organization), the user can be interrupted and information injected that can alter minor or major components of the task. Thus tasks are situated within a social/cultural/organizational environment.

The issue is one of reactivity to the environment, which involves two aspects of GOMS modeling. The first aspect is the goal stack. A shallow goal stack increases reactivity by allowing the system to more frequently run checks on the environment between executing goal stacks (e.g. John & Vera, 1992). The second aspect is the level of abstraction involved in defining the operators (Card et al, 1983). Obviously, operators defined at a gross level of granularity will tend to overlook opportunities in which a person could be interrupted.

Another approach to dealing with this problem is to adopt a parallel processing approach, such as CPM-GOMS (Gray, John & Atwood, 1993). CPM-GOMS uses a schedule chart to represent simultaneously occurring activity, which can be analyzed using a critical path analysis. Using this approach the environment can be monitored in parallel while other tasks are going on. For example, an engineer waiting for some plans might work on a side task while monitoring for news that the plans have arrived.

Modeling Satellite Maneuvers

Currently we are working on a GOMS model of satellite maneuvering, a task that is very routine in nature but also demands a high level of reactivity to the environment. The satellite technicians (hence forth STs) must pay attention to the computer interface system as well as to each other. In addition, the maneuvers take place within the larger context of the satellite management organization. Here we report how we have approached modeling this activity with regard to performing an attitude maneuver.

Method

Unobtrusive observations of satellite attitude maneuvers were conducted. The satellite technicians were all fully trained with an average of 6 months job experience. Two observers took notes during the task. Other sources of data included the task manual, checklists and other handbooks. Separate interviews were also conducted with individual operators after the missions were completed.

The Model

The completion of the maneuver required the fulfillment of seven tasks:

1. Configure system: The satellite maneuver task is semi-automated. The STs must call up a program at the beginning of the scenario which consists of batches of commands and instructions that guide operator's behaviors.
2. Prepare for phase check: This includes selecting data channels for information collection, switching on the printer and refilling the printer paper.
3. Execute phase check: The purpose of this is to ensure synchronization with the satellite. The STs specify the necessary parameters then starts the printer. At the end of phase check, a time-graph is charted and measurements are taken.
4. Prepare maneuver: The STs specify time of the maneuver execution and prepares for it (e.g. automated notation control and antenna pointing are switched off).
5. Execute maneuver: The actual maneuver process is completely automated. However the printer must start running 10 seconds before its execution in order for data collection.
6. Finish maneuver: When the execution ends, the STs check the data and return the spin rate and temperature of the satellite to normal, and the automated notation control and antenna pointing systems to their usual status.
7. Attend alarm: Alarms occurs anytime throughout the maneuver task. The STs have to acknowledge and analyze them individually before continuing the normal task.

Although routine and largely automated, it was observed that the STs actions were intimately dependent on cues provided by the computer interface (primarily the monitor screen). The interface captured many aspects of the maneuver, breaking them down into much smaller sub-goals, and cueing the STs when appropriate. Hence, although the task structure appeared retrospectively as a large serial plan, the STs were actually highly reactive to their environment. This was corroborated by the observations that, 1) STs constantly referred to their monitor screens, 2) STs waited for the external cue before taking action, and 3) when alarms unexpectedly interrupted the scenario, the ST typically completed the sub-goal he was engaged in, dealt with the problem specified by the alarm, and resumed his normal course of action by looking for cues on the interface.

To model the STs' reactivity to system cues we adopted the strategy of using very shallow goal stacks, prompted by the system cues. For example, when the STs perceive the cue "APE to manual" the sub-goal of turning off the antenna pointing system is pushed onto the goal stack. Note the unit task is very small, in this case consisting of only two operators, "enter command" and "verify command," allowing the STs to frequently return to monitoring the environment.

The STs, modeled in this way, do not need to understand the relationship between the sub-goals to successfully accomplish the maneuver. However, this approach could not fully account for the STs behavior. First, some of the sub-goals were not cued by the interface. These were, turn on printer, run printer, collect data, start counting, and select data channels. As a consequence, in one observation the STs forgot to run the printer during the phase check, and the procedure had to be repeated. Second, although none of the cued sub-goals were missed, as the procedure is very routine in nature we would expect that the STs would be aware of a missed a cue after receiving the subsequent cue, which would seem sequentially inappropriate. Finally, if the system made an error, such as failing to give a cue, we would expect that the STs would be aware of this for the same reasons he would be aware of missing a cue (in fact one of the functions of the STs is to detect problems with the system).

To model this we assumed that the STs have knowledge of the sequence of sub-goals for the attitude maneuver. When an external interface representation causes a sub-goal to be pushed onto the stack, it is verified against this knowledge structure. When a sub-goal cued by the interface is in conflict with the ST's knowledge of what should be occurring it signals that a problem has occurred. At this point the STs would engage in problem solving behaviors (similar to Gray, Kirschenbaum and Ehret, 1997, this process could be modeled using a problem solving architecture, such as ACT-R or SOAR). Assuming no problem is detected, after the current sub-goal is verified the next sub-goal can be retrieved while work on the current goal is going on in parallel.

Levels of Analysis

As noted in the introduction, the unit of analysis for a GOMS model can vary. For example, the seven tasks comprising an attitude maneuver could easily be described without distinguishing between the computer systems and the STs. From a GOMS perspective, distributed cognitive systems, such as the satellite operations room, can be considered to possess expert knowledge in the same way that individual humans do. However, since we were interested in the HCI characteristics of the task, we differentiated between the STs and the computer systems, and found that the task knowledge is distributed across the STs and the computer systems, with some level of redundancy.

Another issue is the level of analysis with regard to the STs. In actuality the maneuver is performed in a room containing a supervisor and two operators, one to execute commands and the other to verify commands. The supervisor, in addition to supervising, also functions to communicate with other departments involved with various decisions pertaining to the maneuver. In terms of ST knowledge, our observations showed that, in addition to system cues and the knowledge stored in each ST's long term memory, knowledge was also drawn from interactions with job manuals, checklists, and the other STs.

Essentially, our model treats the STs, including their job manuals and checklists, as a single unit. The behavior of individual STs is never referred to. Treated in this way, the STs function as a distributed cognitive sub-system within the greater distributed system of the operations room. What this level of analysis leaves out is how the STs organize themselves and how they draw knowledge from sources other than the interface system. For example, our model assumes that the STs can retrieve the next sub-goal while work on the current sub-goal is going on in parallel. This could involve an individual ST retrieving the next sub-goal from memory while working on the current sub-goal in parallel, or requesting another ST to look up the next sub-goal on a check list while he or she attends to the current sub-goal. One of the next goals in this project is to model the flow of information between the individual STs, their manuals, checklists, and the computer interface system.

Therefore, to summarize, we can conceptualize this task at three different levels of analysis:

1) The room itself as a distributed cognitive system. This level focuses on a functional description of the task.
2) The computers as one distributed system and the STs (including their non-computerized reference material) as another distributed system. This level emphasizes the interaction between the STs and the computers.
3) The STs as individual systems interacting with their manuals, checklists, and the computer interface system. This level highlights the flow of information around the room.

Social Actions

The social interactions were highly constrained by the demands of the task, as well as organizational policies as to how the STs should interact during this task. In addition, the STs frequently perform this maneuver, as well as other maneuvers which are highly similar in terms of what is required from the STs. Thus the individual behaviors of the STs towards each other are very routine in nature, and as such can be captured using GOMS at fairly detailed level.

However, social interactions need not always be modeled at the level of individual behaviors. For example, the STs can decide who will "execute" and who will "verify" each time the task is performed. Currently we are modeling this decision process as an operator attached to the distributed cognitive system comprised of all three STs. That is, for our purposes we are not interested in the interactions involved in this decision, just the fact that it is made by the STs.

Social interactions between groups are also involved in this process. Specifically, the satellite operations department must interact with the control room (where instructions as to what type of maneuver to perform come from) and the orbital engineers (who calculate the parameters of the maneuver). These other departments are represented at the level of the department (i.e. as distributed cognitive entities without reference to the human and computer agents that make them up), and are modeled only in terms of their relevance to the satellite room.

Conclusions

The thesis of this paper is that GOMS can be used to model large, interactive systems such as organizations or institutions, by treating them as distributed cognitive systems. The example from our ongoing research on satellite operations is highly specific, but this is the point of GOMS modeling - to uncover a knowledge level description of how a particular task is performed. In contrast, work on social interactions and the behavior of organizations has tended to focus on finding general psychological mechanisms (e.g. social psychology), or general principles or patterns of interactions (e.g. sociology, anthropology). GOMS, therefore, offers an alternative perspective which we believe would compliment work in these areas. In addition, by considering the organizational context in which an HCI task occurs we gain a broader picture of the task. This is particularly relevant as organizations are increasingly employing computer networks, instead of isolated PCs.

References

Anderson, J. R. (1993). *Rules of the Mind*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Card, S. K., Moran, T. P. and Newell, A. (1983). *The*

psychology of human-computer interaction. Lawrence Erlbaum Associates, Hillsdale, N.J.
Damasio, A. R. (1994). *Decartes Error*. Avon Books, N.Y.
Fishbein, M. & Ajzen, I. (1974). Attitudes towards objects as predictors of single and multiple behavioral criteria. *Psychological Review, 81*, 59-74.
Gray, W. D., John, B. E., & Atwood, M. E. (1993) Project Ernestine: A validation of GOMS for prediction and explanation of real-world task performance, *Human-computer interaction*, 8 (3), pp. 237-309.
Gray, W. D., Kirschenbaum, S. S., & Ehret, B. D. (1997). *Subgoaling and subschemas for submariners: Cognitive models of situation assessment* (GMU-ARCH 97-01-16): George Mason University.
Hutchins, E. (1994). *Cognition in the wild*. Cambridge, MA: The MIT Press.
John, B. E. (1995). Why GOMS? *Interactions. October, 1995*.
John, B. E. & Kieras, K. E. (1994). The GOMS family of analysis techniques: tools for design and evaluation. *In Human-computer interaction institute technical report CMU-HCII-94-106.*
John, B. E., & Vera, A. H. (1992). A GOMS analysis for a graphic, machine paced, highly interactive task. Behavior and Information Technology, 13, 4, 255-276.
Kahneman, D., & Tversky, A. (1982). The simulation heuristic. In D. Kahneman, P. Slovic, & A. Tversky (Eds.), *Judgement under uncertainty: Heuristics and biases* (pp. 201-208). New York: Cambridge University Press.
Mantovani, G. (1996). Social context in HCI: A new framework for mental models, cooperation, and communication. *Cognitive Science*, 20 (2), 237-269.
Newell, A. (1990). *Unified theories of Cognition.* Cambridge, Mass: Harvard University Press.
Olson, J. R., & Olson, G. M. (1990). The growth of cognitive modeling in humn-computer interaction since GOMS. *Human-Computer Interaction*, 5, 221-265
Vera, A. H. & Simon, H. A. (1993). Situated action: A symbolic interpretation. *Cognitive Science*, 17 (5), 7-48.