A TASK PLANNING, SCHEDULING AND SEQUENCING ORTHOSIS FOR THE COGNITIVELY IMPAIRED

Richard Simpson[1], Deborah Schreckenghost[2] and Ned Kirsch[3]
[1]Department of Rehabilitation Science and Technology, University of Pittsburgh
[2]TRACLabs, Metrica, Inc.
[3]Department of Physical Medicine and Rehabilitation, University of Michigan

ABSTRACT

We are using planning, scheduling and task sequencing software originally developed at NASA JSC to produce a *cognitive orthosis* for individuals who have difficulty planning, scheduling and carrying out tasks. The cognitive orthosis decouples the user interface from the underlying planning, scheduling and sequencing software to (1) support a variety of user interfaces and (2) allow simultaneous access to the schedule by both the client and one or more caregivers.

INTRODUCTION

Neurological diseases and trauma (e.g., traumatic brain injury, stroke, infectious and toxic encephalopathies) are often characterized by alterations of behavioral style, dependence on others for care, and reduced vocational capabilities. These changes represent a significant burden on health care system, communities and families. As an example, for traumatic brain injury alone, Frankowski, Annegers and Whitman [1] have reported a national incidence rate of 200 cases per 100,000 per year, with acute medical costs as high as $165, 294 [2]. The National Head Injury Foundation has estimated the total cost of lifetime care for a severely brain injured person to be as high as $4 million [3].

There are also intangible costs associated with the care of persons having non-progressive but disabling neurocognitive disorders. For example, realignment of family roles and responsibilities may be required as family members are forced to assume an increased burden of care. Clearly, interventions are desirable that facilitate independent performance of functional activities for such individuals, thereby reducing the degree to which attendant and supervisory care are required.

We are currently developing a new interactive activity guidance system based on planning, scheduling and task guidance research performed at NASA's Johnson Space Center [4, 5, 6, 7]. The envisioned system will interact with users over wireless systems and provide web-enabled access to management of plans, schedules and messages for activity guidance. This "cognitive orthosis" will offer a number of innovations. First, it will provide for interaction between a remote planning, scheduling and task guidance system and a wireless device, used by the client[1]. Unlike existing wireless therapeutic cueing systems that only present alphanumeric messages, the system we propose will permit clients to communicate interactively and responsively with the remote system, through web-page based content. Second, based on client responses, the remote system will be able to: a) monitor the progress of activity performance at every stage; b) provide additional assistance when an error occurs, c) manage the interaction between activities and d) provide mechanisms for intervention from third-parties (e.g., a caregiver[2]) when required. Third, these features will permit us to develop clinical interventions for a broad range of functional activities, including those in the community and at work sites.

PRELIMINARY RESEARCH

NASA builds *crew activity plans* prior to a mission. These plans are tightly scheduled. Consequently, crew workload is often high and is performed in a distracting environment. As a result, it is necessary to remind the crew of task deadlines and pending tasks, and to monitor for crew errors. Because operations do not always happen as planned, this plan has to be updated as conditions change and the crew must be notified when the plan changes.

The crew also requires assistance in executing planned tasks. *Crew procedures* describe the actions the crew must take to accomplish a task (e.g., repair a broken gas analyzer). Prior to a mission, the crew practices executing these procedures using simulated mission situations. Even with this pre-mission training, there are so many different procedures and procedures can be so complex that the crew needs to be reminded about specific steps of a procedure at the time it is executed.

ACTIVITY PLANNING

The software used for crew activity is the Adversarial Planner (AP) [8], a non-linear state-based hierarchical task net planner developed for use in military planning. This software automatically generates an activity plan, monitors the execution of this plan by comparing expected effects of activities to actual effects, and dynamically replans when a task fails to achieve the expected effect. It develops plans for multiple agents, both human and software agents, that ensure agents are dedicated to a task. In addition to these features, new capability required for crew activity planning has been added to AP [9]. This new capability includes

[1] The term *client* will be used to refer to the individual with the cognitive impairment who is using the cognitive orthosis.

[2] The term *caregiver* will be used to refer to the person that is responsible for maintaining the client's schedule. This may be a spouse, relative, clinician or therapist.

optimization for temporal and resource constraints (i.e., scheduling) and consideration of user preferences (e.g., preferred start time for an activity) when building plans. It also monitors the plan and notifies the user of pending activities, deadlines, and when a replan occurs.

A unique feature of the NASA technology is that it combines automated task planning and automated task scheduling. When planning, the user specifies a high-level task and the planning software automatically decomposes it into a sequence of detailed tasks. This decomposition can include alternative ways to accomplish a high-level task, based on starting conditions. Placement of tasks in time is opportunistic (i.e., when conditions permit). When scheduling, the user specifies a set of tasks and the scheduling software automatically selects times when these tasks are executed. This time placement includes selecting a time period when adequate resources are available. We have combined planning and scheduling such that the planner instead of the user specifies the set of tasks for the scheduler. For our software, the user only specifies the high-level tasks. The user may also control time placement by specifying preferred times for tasks, such as time of an appointment, and by specifying task priorities.

The combination of planning and scheduling capabilities is needed when planning client activities. Consider an example where the client's schedule include the following tasks:

1. do weekly housekeeping
2. go to doctor's appointment
3. take daily medication
4. water the lawn
5. wash the car
6. bake cookies

Use of Planning Features: Hierarchical task decomposition provided by the planner will automatically decompose the first task "do weekly housekeeping" into a sequence of detailed tasks like "sweep the carpet", "sweep the floor", "mop the floor". This permits the person building the schedule to quickly build complex plans by specifying high-level tasks. The planner also ensures the temporal sequencing of detailed tasks is correct. Thus, it will place the task "sweep the floor" before "mop the floor". And it will request the client to check if the sweeper bag needs to be changed before sweeping the carpet. The planner also provides alternative ways to accomplish a task, depending upon the conditions when the plan is built. For example, if the client drives and the car works, the client will be directed to drive himself to the doctor. If the car is in for repairs, however, the client will be directed to find someone else to drive him. Finally, the planner can build plans for more than one person (called multi-agent planning), which permits coordinating the schedules of the caregiver and the client, when necessary.

Use of Scheduling Features: Using the scheduling capability, the person building the client's schedule can select a specific time for doctor appointment. Before a task is scheduled, the software will verify that adequate resources are available for a task. When a task like watering the lawn can be done concurrently with another task, the scheduling software ensures that the concurrent task does not require resources already in use. In this case, it would not schedule washing the car at the same time as watering the lawn. Finally, the scheduling software permits specifying some tasks as more important (higher priority) than others. Thus, taking daily medication would never be delayed while baking cookies might.

A key component of the activity planning software is the storage of plans in a database. This database can be accessed remotely by both the client and the caregiver via a web interface. Since handheld computers can be readily web-enabled, the client can view the schedule from their personal assistant. The figure below shows a web view of a crew daily schedule.

JOHN - nominal_dec_11_2000

Mon Dec 11	09:00 weekly_medical_lab_op	3 hr
	13:00 eva_base_camp_op	4 hr
Tue Dec 12	09:00 science_show_op	3 hr
	13:00 process_food_op	3 hr
Wed Dec 13	09:00 lab_op	3 hr
	13:00 harvest_crop_op	4 hr
Thu Dec 14	09:00 science_show_op	3 hr
	13:00 lab_op	4 hr
Fri Dec 15	09:00 inspect_op	8 hr
Sat Dec 16	09:00 plant_crop_op	3 hr
	13:00 off_duty_op	4 hr
Sun Dec 17	09:00 off_duty_op	3 hr
	13:00 off_duty_op	4 hr

agent: JOHN Show New Schedule

[top]

Procedure Execution

The software used for aiding crew execution of procedures is the Reactive Action Package System (RAPS) [10], a reactive planning system that dynamically selects and sequences low level task steps based on environmental conditions. Like the activity planner in the previous section, RAPs generates a task plan by breaking down a high-level task into more details steps. It also monitors the execution of that plan. If a plan fails, it dynamically builds a new plan, which may include the use of alternative methods for accomplishing the same task. The fundamental technical difference between this procedure executor and the activity planner is that the procedure executor is *reactive* and the activity planner is *deliberative*. This means that the activity planner plans all tasks before any task is executed (i.e., it is predictive), while the procedure executor plans tasks one at a time, waiting for the first task to complete before selecting the next one (i.e., it is reactive). Thus, the procedure executor can schedule tasks with knowledge of the actual effects of the previous task, making it easily adaptive to unexpected conditions that invalidate the plan.

NASA used the RAPs software to develop an assistant to guide crew through procedures using the Space Shuttle robotic arm [11]. This reactive planning approach is well-suited to providing task instructions for clients as well, because it selects the next instruction based on the results of the current instruction. This permits detecting client errors or misunderstandings, and using this information to select the most helpful next instruction. For example, if the client doesn't understand an instruction, an alternative wording or more detailed instruction can be provided instead of proceeding to the subsequent step. RAPs also permits specifying both sequential and concurrent tasks, enabling more complex instructions like turn on the oven and while it is warming up butter the pan and then put cookie dough on the pan. RAPs provides alternative methods that are only used if the preferred method fails. For example, if the client is still confused after the detailed instruction, the caregiver might be paged to provide help. The RAPs system also provides the ability to automatically execute a task when specific conditions are met or after an elapsed time interval. This permits interrupting the instruction sequence when a specific amount of time has elapsed since a previous instruction or when conditions have changed (such as a client error). For example, when baking cookies the client needs to be instructed to take them out of the oven after a specific amount of time has elapsed.

COMBINING PLANNING AND TASK EXECUTION

NASA has integrated activity planning and procedure execution together to provide automated control of life support systems. The Three Tier (3T) Control Architecture [Bonasso97a] consists of multiple tiers of control processing that operate concurrently. The top tier of this architecture is the Planning tier, implemented using the Adversarial Planner (AP). It predicts the control tasks required to achieve control objectives. The middle tier of this architecture is the Sequencing tier, implemented using the Reactive Action Package System (RAPS). When used together, this software can schedule tasks, monitor the execution of these tasks, and suggest procedures for accomplishing the tasks in this plan. This combination of planning and procedure execution could be the basis of an integrated tool that both manages the client's schedule and automatically instructs the client in how to perform a task when the scheduled time of the task arrives. For example, the planner would notify the client when it is time to bake cookies and the procedure executor would guide the client through the steps of cookie baking.

Cognitive Orthosis Testbed	Commercial Products
Constant, fast internet connection	Sporadic, slow internet connection
Personal Digital Assistant	Multiple hardware types (PDA, pager, cell phone, etc.)
Unlimited computational power and memory.	Limited computational power and memory.

PROPOSED DESIGN

We are currently developing a "testbed" that will serve as a rapid prototyping platform for use in development and evaluation of the cognitive orthosis. The testbed will allow investigators to experiment with the functionality of the system, but will not have the same bandwidth, storage, or computational limitations as existing wireless telecommunication and handheld computing technologies. This will allow investigators to:

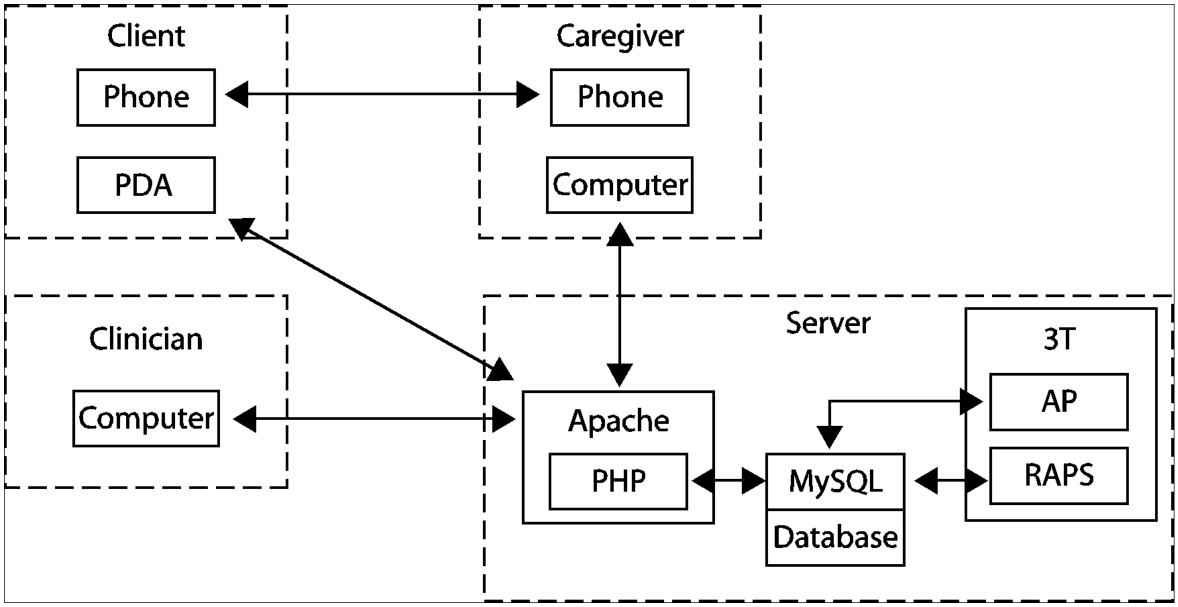
- explore trade-offs between function, bandwidth, storage and computational power;
- experiment with a variety of interface styles;
- perform user trials of the system within controlled environments.

The above table lists several of the differences between the testbed and the commercial products that are available today. It should be noted that several of the differences are due to limitations in existing technology. We expect the resulting cognitive orthosis to be well-positioned to take advantage of future advances in this area, such as the advent of G3 cell phones with integrated PDAs. The figure below presents a schematic of the testbed, which will consist of four components, each of which is described below.

CLIENT

The cognitive orthosis testbed will allow the client to interact with the planning/scheduling software through a web-based interface running on a Personal Digital Assistant (PDA). The PDA will be used to present instructions and solicit information from the client through interactive web pages displayed by the PocketIE web browser. The individual web pages will make use of Javascript and/or Java to present a dynamic interface to the user. Information will be presented in the form of text, pictures and sounds, all of which can be downloaded to the PDA on demand (as opposed to being stored in the PDA) to reduce the amount of memory required by the PDA.

A Compaq iPaq PDA will be used in the testbed to simulate a variety of device form factors. The display of the iPaq is touch sensitive, and can be programmed to simulate interaction with a cell phone, a one-way pager, or a two-way pager (e.g., Blackberry), in addition to a PDA. The iPaq will be connected to a local area network (LAN) by a wireless connection (either using an 801.11b card or a



Bluetooth connection). Communication from the server to the iPaq will be handled using a "Virtual Push" method. This will be accomplished either by continuously polling the server through a Java applet or by using HTML code to cause frequent refreshing of the web page.

The actions that a client will be able to perform using the PDA include:

- Selecting and executing an instruction set;
- Requesting additional information about the current step;
- Providing additional information to the server about the current state of the task environment;
- Indicating that the current step within a task was (or was not) completed successfully;
- Requesting outside intervention from a caregiver.

The cognitive orthosis testbed will also allow direct client/caregiver interaction using a telephone. Ultimately, this is expected to be implemented using an integrated PDA/cell phone, such as the Kyocera 6035 SmartPhone, but will be simulated in the testbed using a separate telephone. Direct communication (i.e., by telephone) between the client and caregiver will be used when the cognitive orthosis testbed and client have exhausted all other means of accomplishing a task.

CAREGIVER

In order for the clinician to provide effective assistance to the client during direct interaction, the cognitive orthosis will have to provide the clinician with as much information about the client's environment as possible. In particular, allowing the clinician to "see what the client sees" will be very important. Ultimately, this is expected to be implemented by integrating a video camera into the unit carried by the client, which can provide a video feed to the caregiver. However, in the cognitive orthosis testbed, this will be simulated by using a video phone or separate video camera.

In addition to directly interacting with the client *during* task execution, the cognitive orthosis testbed will allow a caregiver to interact with the client's schedule *prior* to task execution through a web browser running on a desktop computer. The types of changes a caregiver will be able to make to a client's schedule (either off-line or while the client is actively following the schedule) include:

- Add a task;
- Delete a task from the current day's schedule;
- Reschedule a task for a later time in the day or on another day;
- Interrupt a task with a second task;
- Change information about a task;
- Change a step within a task.

CLINICIAN

The cognitive orthosis will also provide clinicians (and therapists) with access to the client's schedule. Some clinicians may even be provided with the same direct interaction capabilities as the primary caregiver (or may, in fact, *be* the primary caregiver). However, it is expected that clinicians will typically play a less active daily role, and will be more interested in making occasional changes to the client's schedule or task descriptions.

SERVER

Information about each task, and the steps required to accomplish each task, will be stored in a MySQL database on the server. This information will be used by the server to lead the client through each stage in the task and to identify when tasks cannot be completed as scheduled. `A caregiver (or clinician) will interact with the database through a web browser running on their computer. The browser will present forms to the caregiver for querying and modifying data in the database.

A client's schedule will consist of a sequence of tasks - such as "do the laundry," "take medicine," and "eat dinner." Each task will be broken into individual steps (or stages) - such as "add 1 cup of bleach" or "open the bottle of medicine" - that will be presented sequentially to the user. After the user completes a stage of a task, he or she will be asked to provide a response to the cognitive orthosis. This response from the client may take several forms, including:

- an acknowledgement that the stage was completed and the client is awaiting instructions for the next stage of the task;
- some information requested by the cognitive orthosis (e.g., the temperature setting of the thermostat);
- a notification that an event has occurred (e.g., the oven has finished pre-heating);
- an indication that an error has occurred.

The server will then choose what subsequent action to take based on the response provided by the client. Options available to the server include:

- Presenting the most appropriate of several different possible steps to further the task;
- Switching to an "error correction" routine;
- Connecting the client directly with the caregiver.

In addition to the sequence of stages required to complete a task, the caregiver will be able to specify additional information about each task that will be used to schedule each task, including:

- Earliest start time;
- Latest start time;
- Expected Duration (Earliest/Latest finish time);
- Task Priority;
- Resource constraints (e.g., equipment needed, lighting conditions, etc.)

This additional information will be used by the server to manage the flow of control *between* tasks. The actions the server will be able to take automatically include:

- Add a task;
- Extend the duration of a task
- Delete a task from the current day's schedule;
- Reschedule a task for a later time in the day or on another day;
- Interrupt a task with a second task;
- Change information about a task;
- Change a step within a task.

DISCUSSION

The cognitive orthosis testbed will be used to design and evaluate a fully-functional cognitive orthosis. The resulting system will require advances in wireless communication and hand-held computing technology in order to be feasible as a real-world product. However, the behavior and "look and feel" of the cognitive orthosis can be designed now in anticipation of these advances.

REFERENCES

[1] Frankowski RF, Annegers JF, Whitman, S. Epidemiology of head trauma in the United States. In D.P. becker & J.T. Powlishock (Eds.). Central Nervous System Trauma Report, Washington: Neurological and Communicative Disorders and Stroke. 1985; 33-44.

[2] Klauber MR, Barrett-Conner E, Marshall LF, Bowers SA. The epidemiology of head injury: a prospective study of an entire community - San Diego County, California, 1978. Am J Epidemiol. 1981; 5:500-509.

[3] National Head Injury Foundation. Facts About Traumatic Brain Injury. Interagency Head Injury Task Force Report. Bethesda: National Institute of Neurological Disorders and Stroke.

[4] Bonasso, P., Kortenkamp, D., and Whitney, T. 1997. Using a Robot Control Architecture to Automate Shuttle Procedures. In *Proceedings of 9th Conference on Innovative Applications of Artificial Intelligence*, 949-956 Providence, R.I.

[5] Schreckenghost, D.; D. Ryan; C. Thronesbery; P. Bonasso; and D. Poirot. (1998, July). Intelligent Control of Life Support Systems for Space Habitats. *AAAI Innovative Applications of AI 1998*.

[6] Schreckenghost, D., Bonasso, P., Hudson, M.B., and Kortenkamp, D. (2000) "Activity Planning for Long Duration Space Missions". Workshop on Representational Issues on Real-World Planning Systems. *Annual Meeting of American Association of Artificial Intelligence*.

[7] Kortenkamp, D., Schreckenghost, D., and Bonasso, R. P. Adjustable Control Autonomy for Manned Space Flight. *IEEE Aerospace Conference.* Mar 2000.

[8] Elsaesser, C., & MacMillan, T.R. Representation and Algorithms for Multiagent Adversarial Planning. Technical Report MTR-91W000207. MITRE, Wash. D.C. 1991.

[9] Schreckenghost, D., and M. B. Hudson. "Automation in Context: Planning Manned Space Exploration Activities". *Proceedings of ISAIRAS-2001.* Jun 2001.

[10] Firby, J. Adaptive Execution in Complex Dynamic Domains. Ph.D. diss., Yale University., 1989

[11] Bonasso, P, J.Firby, E.Gat, D.Kortenkamp, D.Miller, & M. Slack. Experiences with an architecture for intelligent, reactive agents *Journal of Experimental Theory of AI* 9(97)