The Intelligent Classroom: Providing Competent Assistance

David Franklin and **Kristian Hammond**
Intelligent Information Laboratory
Northwestern University
Evanston Illinois USA
{franklin, hammond}@infolab.nwu.edu

Abstract

In the software industry, designers are forever trying to "improve" their products by adding ever more features to them, producing bloated software systems that are capable of doing just about anything. However, these systems often make it increasingly difficult for their users to perform their tasks as they are forced to wade through a mess of unwanted features to find the few that they actually need. We believe that a fruitful area of research is in building intelligent systems for particular tasks and then having the systems actively try to assist their users in performing these tasks. Such a system knows the plans (and the problems associated with those plans) that their users are likely to pursue. These *competent assistants* can use their expertise on their particular tasks to guide their users through their tasks, provide better help, or perhaps even volunteer to take over some part of the task. Such an assistive agent is able to helpful because, through its task knowledge, it is able to limit what it needs to consider in cooperating with its user. In this paper we look at the implementation of a competent assistant that functions in a physical domain. The Intelligent Classroom is a prototype automated lecture facility that serves as its own audio/visual assistant. We focus on the representations and algorithms to use task knowledge to produce cooperative behavior, arguing these techniques could easily to be extended for use in a wide range of domains (i.e. both physical and purely electronic domains).

1 Introduction

How do we make software and software systems more powerful? In recent history, the answer has almost always been to add ever more features. As a result, when puzzling over how to accomplish some task with your word processor, you are faced with the frustrating knowledge that, while the software almost certainly has a feature for that particular task, it will be terribly time consuming to find it in the vast maze of menus and dialog boxes. The difficulty lies in the way people are expected to interact with their computers. All the effort lies with the users, who must first decide what they want to do and then figure out how they can do it.

Instead, we propose striving to build *competent assistants* that work within the bounds of quite specific tasks. These assistants do not attempt to be experts at everything, but rather they are familiar with all of the various processes that make up their task. Such an agent knows the steps involved in a task, but does not necessarily know the reasons behind them. For appropriate tasks, this limited knowledge is adequate and allows the competent assistant to be very helpful.

Interactive software agents (see (Rich & Sidner 1997) for a promising collaborative framework) have been built along these lines. In this paper, we discuss an agent we have built to function in a physical domain. The Intelligent Classroom is a prototype automated lecture facility that serves as its own audio/visual assistant. As its user, the speaker, goes about his presentation, the Classroom watches and listens to him and, when appropriate, assists him. For example, the Classroom films the presentations that occur within it, producing a video feed that would be suitable for distance learning or for storing in presentation archives. To produce a useful video feed (i.e. to be a good assistant) the Classroom must capture all the activities in the Classroom that the speaker would wish his viewers to see.

So, when the speaker is writing on the board, the Classroom must make sure that it focuses its presentation camera on the words and figures that the speaker is writing. It must zoom the camera in tight enough that the words can be read, but also capture the surrounding words so that people viewing the video feed see the words in their greater context. Also, when the speaker is standing still and lecturing, the Classroom should zoom the camera in on his face so that the viewers can see his face expressions. To produce a reasonable video feed of a presentation, the Classroom must utilize a wealth of camera framing techniques, and in order to determine which technique is appropriate, the Classroom must have at least a superficial understanding of what the speaker is doing at any given moment.

When the speaker is lecturing from a set of slides, the Classroom must listen to the speaker so it can tell when it is time to switch slides. The speaker might indicate that he wishes to go on by issuing a command–saying, "Next slide, please" or touching the next slide "button" on the display screen. Or, as the Classroom follows along with what the speaker is saying, it can infer that the speaker wishes to go on to the next slide when he has finished talking about the current slide and has begun discussing a new one.

When the speaker wants the Classroom to play a particular video segment, the Classroom must cue up the video on the VCR, set the display to accept the VCR output, start the video, and begin listening for the various commands that are associated with playing videos (i.e. "pause", "skip forward" and "go back"). The speaker might express his desire to play the video by explicitly telling the Classroom to play the video segment, or, if the speaker provides a script of the presentation to the Classroom, it will know when it is the appropriate time to play the video.

In these examples, we hope to provide a sense of what it means to be a competent assistant. To be useful, an assistant must have an understanding of the tasks it is to be used in, and how it is to assist in accomplishing these tasks. Ideally, such an assistant is able to accurately observe what is going on around it, and always infer how it can best assist. But since this ideal is not always achieved, the assistant should also know the sorts of things that it may be asked to do and always be listening for directions. In all of these examples, the assistant is the most useful when it is actively cooperating. When it sees a way that it can help, the assistant just does it.

2 Building a Competent Assistant

This paper focuses on the components of the Intelligent Classroom that allow it to serve as a competent assistant. Essentially, this involves looking at the way the Classroom is embedded in its world: how it views what is going on and how it acts in response. In the Classroom domain, these questions are phrased as follows:

- How does the Classroom reason about the speaker's actions? The Classroom attempts to understand everything it observes the speaker do in terms of a high-level explanation of what he is doing. These explanations do not need to be particularly deep–for its plan recognition and monitoring, the Classroom just needs to know what it should be able to observe. The Classroom maintains a set of activities going on in the Classroom: the speaker's current activity (e.g. lecturing at the podium) and any activities that the Classroom is performing (e.g. playing a video tape). This is the Classroom's understanding of what is going on in the presentation. When something happens that is not explained by any of the processes in its understanding, the Classroom must revise this activity set to include something that would explain the strange observation. For example, while the speaker is playing a video, the Classroom does not expect him to go over to the board and start writing. If he does, the Classroom will have to revise its understanding of what he is doing.

- How does the Classroom cooperate with the speaker? Each plan in the Classroom's plan library represents a common understanding of how a speaker and an audio/visual assistant might interact. The plan contains the speaker's actions as well as what the Classroom should do. The idea is that the speaker, in doing his part of the plan, is implicitly assuming that the Classroom will do its part. For a given interaction (like having the lights turned on) there may be many several different ways the speaker would do his part of the plan. Rather than having the speaker memorize that particular command for each thing he wants to do, we want the Classroom to watch for whatever the speaker may do. The ultimate goal is for whatever works with a human A/V assistant to also work with the Classroom.

2.1 Representing what is going on in the Classroom

The Classroom models all activities as sets of running continuous actions punctuated by discrete events. Basically, an agent sets some continuous actions in motion and then watches them run. The agent watches for certain events to occur, prompting it to set some different continuous actions in motion. Each of these sets of continuous actions can be viewed as a step in doing the task. We use the term *process* to refer to

```
(define-process (move-to-board-and-write)
  (main-actor
   (person ?speaker))
  (roles
   (markerboard ?board)
   (marker ?marker))
  (steps
   (_go (do _move (move-to ?board))
        (track _delta (track-floor-distance
                         ?speaker ?board))
        (wait-for (_move :done) _2)
        (ensure _delta (:decreasing)))
   (_do (do _choose (select-marker ?marker))
        (wait-for (_choose :done) _3))
   (_3 (do _pickup (pick-up-from
                      ?marker ?board))
       (wait-for (holding ?speaker ?marker)
                 _4))
   (_4 (do _write (write-on-board ?board))
       (wait-for (_write :done) :done)))
  ...)
```

Figure 1: A process definition for the speaker going to the chalkboard and writing

these sequences of steps. The steps in theses processes may be totally ordered (with one step always leading directly to the next) or they may be only partially ordered (with the results of each step determining which step will follow). These processes can refer equally well to the Classroom's activities and those of the speaker. In the case of the speaker's activities, the Classroom is observing the speaker's actions in order to follow along with him in his process.

Figure 1 shows a simplified version of the process representation the Classroom uses to model what the speaker does in going over to the board and writing. The first parts of the process definition give the process a name and list the important process roles such as agents, locations and props. Then the steps describe what to do in executing the process.

The first step (labelled `_go`) invokes another process that will handle the actual movement towards the marker board and starts monitoring the distance to the board. The process and monitor are the continuous actions for this step. The events that the step is waiting for are either an arrival at the board (in which case the process advances to the next step) or a lack of progress towards the board (in which case the process fails). Figure 2 shows the state of the Process Manager (the component in the Classroom that keeps track of processes) as the Classroom observes the speaker performing the first step in the process.

The other three steps in the process involve the speaker selecting a marker from the tray on the marker board (the Classroom keeps track of the locations of the markers and eraser), picking it up and finally actually writing with it. As would be expected, when the writing is done, the process itself is done.

The representation for a particular process can be used from two perspectives. From the perspective of an actor (the speaker), it tells what to do. From the perspective of an observer (the Classroom), it tells what it looks like when someone else is doing it. Both the actor and the observer utilize the progress conditions, and, for both, the mechanisms for dealing with the hierarchical decomposition of the process are very similar. In looking at how this dual nature of processes is used in the Intelligent Classroom, it is important to state that we do not actually believe that speakers literally represent their plans in this way and execute them using these mechanisms. Rather, we believe that speakers have process representations that are analogous to these and that the Classroom can observe them as if they actually were using the processes it represents.

The process representation relies upon a lower-level system (described in (Franklin & Flachsbart 1998)) that handles the actual interaction with the physical world. This system is important to this discussion because physical sensors do not produce the discrete events that the higher-level components (i.e. those discussed in this paper) require and the discrete commands produced by the higher-level do not map neatly into actuator settings. The two-layer architecture we use provides a convenient layer of abstraction, making the messy continuous world into a somewhat neater and more discrete one. For sensing, the higher-level system configures the lower-level system to tell it what to look for and to suggest techniques that might be more effective in the current Classroom context. In this paper, *speaker events* refer to the discrete events produced by this low-level system. A *speaker action* generally refers to the discrete event that indicates the speaker has began that action (e.g. "the speaker has started moving" or "the speaker has assumed a writing pose at the board").

2.2 Plan recognition

The Process Manager provides a mechanism for following along with processes that the speaker is executing. An important part of this mechanism is making sure that a process is progressing well. A process can be considered to be progressing poorly if a monitored quantity does not behave as expected, if a step takes too long (steps may provide a range specifying how

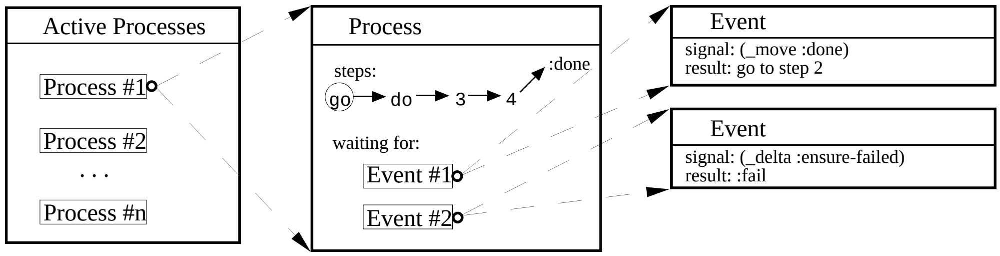

Figure 2: Important structures in the Process Manager

long it should take to complete), or if the speaker takes an action that is inconsistent with him performing the process (process steps specify what speaker actions can be expected, and what to do as a result).

This ability to detect when a process is not progressing well is central to the Classroom's plan recognition ability. The basic idea is that the Process Manager carefully monitors a single top-level process that serves as its understanding of what the speaker is doing at the moment. But, if that process fails or if the speaker takes an action that is inconsistent with the process, the Classroom must come up with a new explanation for what the speaker is doing. First, it looks at the speaker's most recent actions and determines what possible speaker processes are consistent with those actions. Then, as future speaker actions contradict the presence of some of these candidate explanations for the speaker's actions, the set of processes is eventually reduced to a single process: the Classroom's explanation for what the speaker is currently doing.

So, when faced with an unexplained speaker action, the Classroom finds all the processes in its library that: (1) can have the speaker as their main actors, (2) have the unexplained action as a part of one of their steps, and (3) are consistent with the speaker's previous actions. By looking through the process definitions and looking for the appropriate information, the Classroom can easily verify the first two conditions. The third condition requires a little more work. If the unexplained action is in the process' first step, this condition trivially holds because there are not any actions before the step that could be inconsistent with what actually was observed. For other cases, the Classroom must simulate the running of the process to be sure that everything the process would expect to happen before the unexplained action, did happen. To facilitate this, the Classroom maintains a history of the speaker's last five actions (i.e. the events reported by the lower-level system). For each of the (four) possible sequences of actions leading up to the unexplained speaker action, the Classroom simulates the running of the process. If the process runs successfully in simulation, explaining each of the speaker's actions, the process is considered a candidate explanation for the speaker's action–it has met all three conditions.

Once the Classroom has a set of candidate explanations for the speaker's action, it must reduce the set to the single correct explanation. This is not immediately possible because the Classroom simply does not have enough information to determine which process the speaker is performing. So, the Classroom eliminates the incorrect processes as the speaker performs more actions. There are two ways that the Classroom is able to eliminate the incorrect processes. For the first, the Classroom eliminates processes that are not progressing properly. This means either that one of the time constraints was broken or that one of the physically measurable quantities is not behaving properly. For instance, if the Classroom has hypothesized that the speaker is going over to write on the board, it expects the speaker to arrive there within a few seconds and to be moving generally towards the board. If the speaker stops before getting to the board or starts moving away from the board, the Classroom will be able to eliminate the `move-to-board-and-write` process from consideration. In both cases, the temporal constraint will be violated and the measurable quantity (the distance between the speaker and the board) will not be behaving properly.

The second way that a process can be eliminated is if the speaker performs actions that are inconsistent with him executing the process. Since the Classroom assumes that the speaker is only doing one thing at a time, the Classroom expects all the speaker's actions to be explained by the speaker's process. Therefore, if the speaker takes an action that is not a part of a candidate process, that process can be eliminated from consideration. This is much like the situation that prompted the plan recognition in the first place: the Classroom thought the speaker was doing one thing, but the speaker's actions indicated that he was doing something else.

In the actual implementation, the Process Manager provides the functionality for keeping track of all the candidate explanations and eliminating them as future speaker actions contradict them. When the Classroom observes the speaker taking an action, it tells the Process Manager to deal with the event of the action occurring. Then, the Process Manager tries to use this event to advance the processes in its process set. In doing so, it also makes sure that the top-level speaker process (or one of its sub-processes) explained the speaker event. If the speaker event is not explained, then the Classroom will need to find a new explanation for what the speaker is doing. It gathers a set of candidate explanations and adds these processes to the Process Manager, marking them as explanations.

As the Process Manager runs, it checks that each candidate process is progressing normally, and eliminates those that are not. When the Classroom observes the speaker taking another action, the Process Manager will advance the candidate explanations using the event associated with this action. Any of the processes that do not explain the action are removed from the Process Manager. When only one candidate explanation remains, it is accepted as the correct explanation and its mark as a candidate explanation is removed.

2.3 Cooperation

Given its understanding of what is going on, how does the Intelligent Classroom decide what to do? First, given the top-level process the speaker is executing, the Classroom selects the (unique) plan from its plan library that is associated with it. Then, for activities where the speaker is expecting the Classroom to cooperate with his intentions, the Classroom must execute its process(es) in the speaker's plan. The important thing here is to be sure not only to do the right thing, but do it at the right time. The plans define "at the right time" by stating how steps in different processes need to be synchronized. For example, a plan might essentially say that the Classroom should change the camera mode immediately after the speaker enters the vicinity of the marker board.

Formally, we define a *plan* as a set of processes (often to be executed by a number of different agents) that when run together successfully, accomplish some goal. In the Intelligent Classroom, many plans have some processes executed by the speaker and other processes executed by the Classroom. It is important to note that this is not really a new definition of plan–any plan that has a step of the form "wait for this event to happen" is implicitly representing processes external to its main actor. This definition makes explicit the presence of other agents or exogenous events.

```
(define-plan (move-to-board-and-write)
  (main-actor
   (person ?speaker))
  (roles
   (intelligent-classroom ?classroom))
  (accomplishes
   (?speaker
    (do (move-to-board-and-write))))
  (processes
   (_p1 ?speaker
        (move-to-board-and-write))
   (_p2 ?classroom
        (film-move-to-board-and-write
           ?speaker)))
  (synchronization
   (equals (_p1 _go) (_p2 _1))
   (starts (_p1 _do) (_p2 _2))
   (equals (_p1 _4) (_p2 _3))))
```

Figure 3: A plan definition for the speaker going to the chalkboard and writing

Figure 3 shows a portion of the plan representation the Classroom uses when the speaker walks over to the board and writes. The plan uses the previously described process for what the speaker does (walking and writing) and a process which details how the Classroom should film the speaker as he walks and writes. Finally, the plan definition itself ties these processes to a particular goal and specifies how the processes fit together. In the plan shown in Figure 3, the synchronization clauses tell the Classroom which camera techniques to use while the speaker executes his part of the plan. While the speaker is walking, the Classroom zooms out and tracks the speaker's motion. When the speaker arrives at the board, the Classroom uses a camera technique that shows the contents of the board while keeping the speaker in the camera frame. Finally, once the speaker starts writing, the Classroom zooms in on what he is writing. The Classroom keeps track of the particular region of the board that this piece of writing spans, and shows the whole region so that people watching the video feed can understand the writing in its context.

2.4 Generalization

One tricky bit: the Classroom cooperates based on its understanding of what the speaker is doing, and

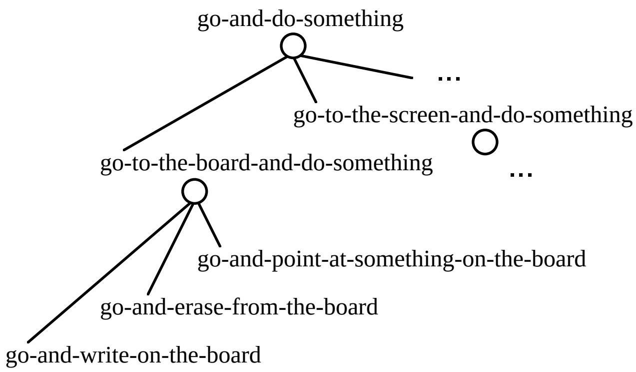

Figure 4: A portion of the explanation hierarchy used in the Intelligent Classroom

often must act before it has eliminated the candidate explanations. To deal with this, we have organized all the speaker processes into a single ISA hierarchy, and the Classroom always acts upon a generalization of its set of candidate processes. This generalization process is the most specific process that is an ancestor of all of the candidate processes. As candidates are removed, this generalization process can be made increasingly specific.

The leaves of the hierarchy are the "real" processes. For example, the Classroom has processes for going to the board to write, going to the board to revise something, and going to the board to point at something. A generalization of these three processes is a process for going the board to do something. A further generalization is a process for going somewhere to do something. For any set of candidate processes, there is a unique most-specific generalization process.

Figure 4 shows the hierarchical structure of a subset of the speaker processes used by the Classroom. The generalization (interior) processes capture the common structure of their children. The various "marker board" processes are broken into "go to the board" and "do something" parts. As the generalization process is changed (i.e. as candidate explanations are rejected), the common structure allows the Process Manager to start the more specific generalization process at the right point.

Whenever the Classroom tries to explain a speaker's action, it adds a generalization process to the Process Manager, along with the candidate explanations. This generalization process is considered to be what the speaker is actually doing. Initially, it will be an overly vague notion of what he is doing, but at all times, it is as precise as the evidence warrants. Whenever the Classroom further specifies this generalization process, the Process Manager replaces the process with a more specific one. The new process is started at the same step as the old process was in, and all of the appropriate variable bindings are copied over as well. This allows the new process to pick up as close as possible to where the old one left off.

The Intelligent Classroom has a very particular use for plan recognition: figure out what the speaker is doing so that the Classroom knows what it ought to do to cooperate. The speaker might frequently change what he is doing, so the Classroom must act quickly and often has little time to determine precisely what he is doing. Fortunately, in the plans that the Classroom needs to recognize, it is nearly always appropriate to choose actions based on an overly general notion of what the speaker is doing. The Classroom can be more helpful with a precise understanding, but it is still helpful with a more general one.

2.5 Issues

This section (Section 2) discusses the general framework used by the Intelligent Classroom to interact with speakers giving presentations. The algorithms and forms of representation define the kinds of interaction that are possible; any competent assistant will need to have analogous components (i.e. systems that sense user actions, build an explanation for those actions, and decide what to do to cooperate). The domain-dependent process and plan definitions define the actual interaction. This is the task knowledge the agent uses to be helpful. Section 3 details two ways in which the Classroom interacts with a speaker in his presentation task.

Our choices of algorithms and representation do limit the sorts of interactions that are possible, but, for the Classroom domain, we feel that these limitations are acceptable. First, the Classroom only reasons about a single speaker plan. This means that the Classroom does not directly reason about interleaved or suspended activities. If a speaker interleaves actions from two activities during a presentation, the Classroom will explain the resulting sequence of actions as series of short activities. When the activities are things like writing on the board or viewing a brief segment of a video tape, such overly simplistic explanations are sufficient to produce cooperative behavior. However, it is not clear whether it will remain sufficient as we build more complicated behavior into the Intelligent Classroom.

Second, the Classroom does not directly reason about the speaker's goals; the Classroom's interactions rely on the speaker's activities, not the underly-

Speaker:	Classroom:
1) "Play the video segment"	
	A) Cue the tape
	B) Set video source
	C) Start the VCR
2) "Pause"	
	D) Pause the tape
3) "Rewind a bit"	
	E) Slowly review
4) "Play"	
	F) Resume playing

Figure 5: A time-line of events occurring while the speaker presents a video segment

ing goals. Again, for the Classroom's interactions, this light goal representation is sufficient for now, but to produce more interesting cooperative behavior, goal-based reasoning will almost certainly be necessary.

The plan recognition techniques employed by the Intelligent Classroom are those of "keyhole" recognition (i.e. the sort of plan recognition one could perform while observing someone through a door's keyhole). But, unlike the typical actors in pure keyhole recognition, the Classroom's speaker is aware that the Classroom is trying to understand his actions and, furthermore, he wishes for the Classroom to successfully understand. The Classroom, however, does not rely on "intended" recognition techniques, because we do not wish to require the speaker to think about how to communicate his intentions. Rather, we want the speaker to be able to just go about his presentation, trusting the Classroom to just do its job.

3 Examples

In this section, we look at how the plan representation discussed in this paper is interpreted by the Intelligent Classroom. We step through the execution of two plans that are used in the Classroom, looking at how the plan and process pieces fit together. Each example starts at the moment that the Classroom realizes it needs a new explanation for what the speaker is doing and ends when the plan has reached its end.

3.1 Playing a video segment

Figure 5 shows a time-line representing the events that occurred during one instance of the speaker playing a video segment as a part of a presentation. The numbered events on the left refer to actions that the speaker took and the lettered events on the right refer to actions that the Classroom took in cooperation. Throughout this section, phrases such as "Event 1" and "Event A" refer to the corresponding events in this time-line.

```
(define-process (present-video-segment)
  (main-actor
   (person ?speaker))
  (set-up
   (require vcr-command-set))
  (steps
   (_1 (wait-for (speak-command i-start-vcr)
                 _play))
   (_play (wait-for (speak-command i-pause)
                    _pause))
         ...
         (wait-for (speak-command i-stop)
                    :done))
   ...))
```

Figure 6: A partial process definition for the speaker presenting a video segment

For this example, the speaker already placed a videotape in the Classroom's VCR and furthermore, he used a presentation script (a special process that a speaker can use to outline the key events in his planned presentation) to specify what video segment is appropriate at this point in the presentation.

Figure 6 shows the process making up the speaker's part in showing a video segment as a part of a presentation. The first step requires the speaker to command that the video be played and the other steps correspond to various states that the VCR can be in. The plan for showing a video segment includes this process and a process for the Classroom that directly controls the VCR. Each step in the speaker's process corresponds to a step in the Classroom's process that puts the VCR in the appropriate state. (There is also a Classroom process that is responsible for making the video feed of the presentation, but it will be ignored in this example.)

Referring back to the time-line in Figure 5, when the Classroom hears the speaker say, "Play the video segment,"[1] (Event 1) it recognizes this as a command

[1] Speech input is handled by a dedicated speech recognition computer running IBM ViaVoice. The Intelligent Classroom is always listening for a set of high-level commands and, based on the Classroom context, may also listen for lower-level context-dependent commands. In its slide-advancing task, the Classroom actually enables the dictation grammar as well. ViaVoice allows the Classroom to dynamically enable and disable command and dictation grammars. Recognized commands and utterances are reported as speaker events.

Speaker:	**Classroom:**
1) Leave the podium and start towards board	
	A) Propose candidate explanations
2) Arrive at board	
	B) Refine to "go to the board to do X"
3) Pick up marker	
	C) Refine to "go to the board to write"
4) Write	

Figure 7: A time-line of events occurring while the speaker goes to the marker board and writes

to go on to the next step in its presentation script and starts up the plan for showing a video segment. In starting its process in this plan, the Classroom makes sure the tape is cued properly (Event A), sets the display to accept the VCR output (Event B), and starts the videotape (Event C). From the speaker's process, the Classroom knows to start listening for the various voice commands involved in controlling a video presentation. (The fact that a video is playing provides a context in which to understand otherwise ambiguous commands like "stop" and "pause".)

Now both processes are in the steps corresponding to simply playing a videotape segment. The Classroom expects either that the speaker to say one of the VCR commands or that the segment will play to completion. When the speaker says, "Pause," (Event 2) the speaker's process is advanced to the "pausing" step and, since the active plan requires that this step in the speaker's process be run at the same time as the corresponding step in the Classroom's process, the Classroom's process is also advanced to its "pausing" step, and the Classroom pauses the tape (Event D). In dealing with the remaining speaker events, the Classroom keeps its process synchronized with the speaker's, allowing the speaker to control the VCR.

3.2 Writing on the board

In this example, we focus on how the Classroom revises its understanding of what the speaker is doing over the course of several actions. We track how the Classroom builds a set of candidate explanations for a speaker's unexpected action and how the Classroom eventually discovers that the speaker is going over to the marker board to write something.

Figure 7 shows a time-line representing the events that occurred during one instance of the speaker going over to the board and writing. In this example, the Classroom is not explicitly expecting the speaker to write on the board; there is no presentation script and the speaker is currently speaking at the podium. When the speaker leaves the podium (Event 1), the Classroom is not expecting it and simply recognizes that this is an unexplained speaker action, requiring a new explanation for what the speaker is doing.

Unfortunately, in proposing candidate explanations (Event A), the Classroom finds that the speaker's single action does not restrict the speaker's possible plans very much. All the Classroom can infer is that the speaker is going somewhere to do something. The Classroom adopts this overly general explanation (Event B) and starts up its process in this plan (which films the speaker as he walks about).

However, as the speaker moves towards the marker board, he is simultaneously moving away from the other potential destinations, causing plans that deal with the display screen or classroom props to be rejected. So, the Classroom infers that the speaker is going to do something at the board (Event B) and adjusts its presentation camera to focus on the board, even before the speaker arrives there (Event 2).

Once the speaker is at the board, the Classroom is considering only three explanations for what he is doing: that he intends to write something, erase something, or point out something that he previously wrote. When he picks up a marker (Event 3), the Classroom finally recognizes precisely what he is doing (Event C) and begins to watch for writing motions. Figure 8 shows how the Classroom films the speaker as he writes. When the speaker starts writing, the Classroom keeps track of the region of the board that he writes on and focuses the camera there.

4 Related work

The research discussed in this paper is the result of a line of research aimed to integrate simple plan recognition into the behaviors of an autonomous mobile robot. Some of the first efforts in this direction involved having the robot recognize gestures to indicate an object to pick up. In (Franklin *et al.* 1996), a human "restaurant patron" could point at a soda can on the floor which the robot would then go over to, pick up and then hand to the patron, asking him to enjoy the beverage. (Firby *et al.* 1995) describes the layered system architecture that was used in that task, allowing a symbolic reactive planner to closely control the closed control loops that actually moved the robot. This architecture serves as the basis for the architecture of the Intelligent Classroom and the plan representations used with the robot (Firby 1994) have been adapted to become the process representations described in this paper. An early

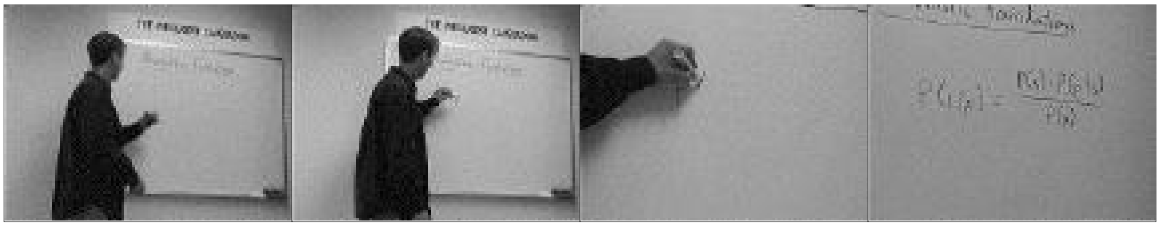

Figure 8: As the speaker writes, the Classroom zooms in on the writing

technique of looking at processes from both the perspective of execution and monitoring is described in (Earl & Firby 1997).

The Intelligent Classroom as a domain fits mostly closely into the realm of Intelligent Environments, physical spaces that observe and listen to their users, often utilizing computation in areas where people would not expect. For instance, in (Coen 1998), the author describes Intelligent Environment research in an automated conference room that tracks the motions of participants, listens for commands and displays results on projection screens. It also talks about the development of an intelligent office that observes activity, plays music, and will even serve as an alarm clock for a napping office-dweller. In (Lucente, Zwart, & George 1998), the authors describe a multi-modal system that allows a human user to manipulate the objects in its display through a combination of speech and gesture. The user can add, remove, relocate and resize any number of graphical objects. The Intelligent Classroom differs from these Intelligent Environments in that it has a plan recognition component; these systems essentially provide a multi-modal command interface to their functionality.

The eClassroom project (formerly Classroom 2000 (Abowd *et al.* 1996)) is the Intelligent Environment that is outwardly most similar to the Classroom (i.e. it is a classroom). However, the eClassroom serves mostly as a information gathering tool, setting up a web page through which students can review the lecture and everything that was written, said, or viewed in it.

The Intelligent Classroom relies on plan recognition in a way that, to my knowledge, is different than any existing system, in that it is sensing and acting in a physical domain (Franklin 1998). However, many of the plan recognition problems and techniques have been addressed to a degree by other researchers. The hierarchical recognition technique we use is inspired by (Kautz & Allen 1986) which allows a recognizer to always have a working theory. Also, in (Tambe & Rosenbloom 1995), a simulated fighter pilots observe other agents in order infer what they are doing and then react (for or against as appropriate). The plans described in this paper utilize a single-step backtracking technique to efficiently deal with plan revisions.

The temporal representations used in the Classroom keep track of events in a presentation, are similar to those used in (André & Rist 1996) to control a multimedia presentation (being made by an animated agent). Another, virtual agent, Steve (Rickel & Johnson 1997) serves as a tutor to help students to learn procedural tasks in a virtual environment. As a student performs a task, Steve follows along with its representation of the task, making sure that the student performs the task properly. Steve's plan representations include the reasons for each task step, so he can explain why each is important. Steve uses his plans in a very different way than the Classroom uses its, but the sorts of step preconditions, postconditions and causal links that Steve uses could be used by the Classroom to better understand user actions and to speed up the plan recognition by eliminating nonsensical plans.

5 Conclusions

In this paper we introduce the notion of a competent assistant which knows just enough about a task domain to be very helpful. Further, we discuss the Intelligent Classroom as an implemented example of such an agent. The underlying implementation of the Classroom was designed with cooperation in mind and we believe that the representations and algorithms could easily be applied to a wide range of domains to provide similar cooperative behavior. The reason for this is simple: if an agent knows its domain and what goes on in it, the reasoning needed to understand what it observes and to cooperate with other agents is made relatively simple.

References

Abowd, G. D.; Atkeson, C.; Feinstein, A.; Hmelo, C.; Kooper, R.; Long, S.; Sawhney, N. N.; and Tani, M. 1996. Teaching and learning as multimedia authoring: The classroom 2000 project. In *ACM Multimedia'96 Conference*.

André, E., and Rist, T. 1996. Coping with temporal constraints in multimedia presentation planning. In *American Association for Artificial Intelligence*.

Coen, M. H. 1998. Design principles for intelligent environments. In *15th National Conference on Artificial Intelligence*.

Earl, C., and Firby, R. J. 1997. Combined execution and monitoring for control of autonomous agents. In *First International Conference on Autonomous Agents*.

Firby, R. J.; Kahn, R. E.; Prokopowicz, P. N.; and Swain, M. J. 1995. An architecture for vision and action. In *Fourteenth International Joint Conference on Artificial Intelligence*.

Firby, R. J. 1994. Task networks for controlling continuous processes. In *Second International Conference on AI Planning Systems*.

Franklin, D., and Flachsbart, J. 1998. All gadget and no representation makes jack a dull environment. In *AAAI 1998 Spring Symposium on Intelligent Environments. AAAI TR SS-98-02*.

Franklin, D.; Swain, M. J.; Kahn, R. E.; and Firby, R. J. 1996. Happy patrons make better tippers: Creating a robot waiter using perseus and the animate agent architecture. In *International Conference on Automatic Face and Gesture Recognition*.

Franklin, D. 1998. Cooperating with people: The intelligent classroom. In *15th National Conference on Artificial Intelligence*.

Kautz, H. A., and Allen, J. F. 1986. Generalized plan recognition. In *National Conference on Artificial Intelligence*.

Lucente, M.; Zwart, G.-J.; and George, A. D. 1998. Visualization space: A testbed for deviceless multimodal user interface. In *AAAI 1998 Spring Symposium on Intelligent Environments. AAAI TR SS-98-02*.

Rich, C., and Sidner, C. 1997. COLLAGEN: When agents collaborate with people. In *First International Conference on Autonomous Agents*.

Rickel, J., and Johnson, W. L. 1997. Integrating pedagogical capapbilities in a virtual environment agent. In *First International Conference on Autonomous Agents*.

Tambe, M., and Rosenbloom, P. S. 1995. RESC: An approach for real-time, dynamic agent tracking. In *Fourteenth International Joint Conference on Artificial Intelligence*.