Appears in Case-Based Reasoning Integrations: Papers from the 1998 Workshop, AAAI Technical Report WS-98-15, pp. 61–66

Case-Based Seeding for an Interactive Crisis Response Assistant

Melinda T. Gervasio and **Wayne Iba** and **Pat Langley**
Institute for the Study of Learning and Expertise
2164 Staunton Court, Palo Alto, California 94306
{gervasio, iba, langley}@isle.org
`http://www.isle.org`

Abstract

In this paper, we present an interactive, case-based approach to crisis response that provides users with the ability to rapidly develop good responses while allowing them to retain ultimate control over the decision-making process. We have implemented this approach in INCA, an INteractive Crisis Assistant for planning and scheduling in crisis domains. INCA relies on case-based methods to seed the response development process with initial candidate solutions drawn from previous cases. The human user then interacts with INCA to adapt these solutions to the current situation. We will discuss this interactive approach to crisis response using an artificial hazardous materials domain, HAZMAT, that we developed for the purpose of evaluating candidate assistant mechanisms for crisis response.

Introduction

Crisis response has been the focus of a considerable amount of AI planning and scheduling research. While early systems were predominantly autonomous in nature, more recent systems provide interactive modes that allow human users to retain control of the problem-solving process (e.g., OPLAN-2 (Tate et al., 1994), SOCAP (Bienkowski, 1996)). Like these systems, the interactive crisis assistant we present in this paper allows a serendipitous utilization of human and intelligent systems' strengths. However, while these recent interactive systems support the generation of solutions from scratch, the need for speed in crisis situations, coupled with the availability of planned emergency response strategies, suggests a *case-based* approach to computational assistance.

The interactive, case-based approach we have developed uses a case library to provide initial candidate solutions that the human user can then interactively modify to suit the current situation. In this *slave-master* integration, the case-based component serves a subsidiary role to the subsequent interactive planning and scheduling processes. We explore this approach in the context of HAZMAT, a domain involving hazardous materials incidents, which we describe in the next section. We then present INCA, our INteractive Crisis Assistant that implements this approach in HAZMAT. Finally, we discuss related work and future work.

The Hazardous Materials Domain

A hazardous materials incident occurs when a spill of some material with hazardous properties poses a threat to humans, property, or the environment. We developed our artificial world, HAZMAT, using the North American Emergency Response Guidebook (NAERG) (Transport Canada et al., 1996), a handbook for first responders that describes the appropriate responses for different hazardous materials situations.

HAZMAT Problems

In HAZMAT, there are fifty different classes of hazardous materials, varying in form and in hazardous properties. A HAZMAT *incident* is a spill, and possibly a fire, involving one of these hazardous materials. There are four thousand different incident classes, differing in size, location, amount of material spilled, spill rate, amount of material on fire, and burn rate. In addition, incidents have associated fire and health hazards that are functions of the material, spill, and fire comprising an incident. These measure the probability of a fire starting and the level of danger to one's health, as well as their respective growth rates.

There are forty-nine different actions currently available in HAZMAT for addressing the spill, fire, and hazards presented by an incident. Different actions require different resources, of which there are twenty-five different types. For example, the action of `x-hose-water-manned-tanker` requires crew members, hoses, pumpers, and tankers; while the action of `absorb-with-dry-sand` requires crew members and dry sand.

A HAZMAT problem consists of one or more incidents and some number (possibly zero) of each type of resource. For simplicity, we consider single-incident problems. Given a particular type of hazardous material, NAERG recommends a set of actions to be used in developing a response. We call these actions the *legal* actions for an incident. In general, only a subset of these will be used in a response, as some will be alternatives to each other and some may have insufficient

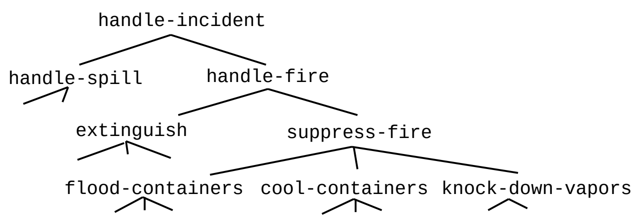

Figure 1: Hierarchical response plan.

resources available. The HazMat response task is to choose a subset of the legal actions for a problem and to schedule them on the available resources so that they can be executed to deal with the incident.

HAZMAT Solutions

There are two parts to an incident response: a plan and a schedule. Plans in Inca are hierarchical in nature and are represented by a tree (Figure 1). The root node of every plan is the abstract action `handle-incident`. A node at one level expands to a set of nodes at the next lower level. In contrast to traditional hierarchical task networks, these nodes are not conjunctive. They do not all have to be executed but more than one may be executed—thus, both a null plan and a plan including all legal actions are valid solutions. The difference is in their impact on the incident and in their respective costs. There are also no causal supports between the actions of a plan—that is, an action does not establish preconditions for any other action.[1]

The leaves of a plan constitute the actions or jobs to be scheduled on the available resources. Only *primitive* actions (i.e., actions with no further expansions) may be scheduled; thus, unexpanded nodes cannot be scheduled. Scheduling a job involves four decisions: determining the amount of resources to allocate, selecting specific resources, and assigning a start time and duration. Thus, a schedule associates every job with a set of simultaneous time intervals on a set of resources.

Each resource is associated with a *capacity*, representing the maximum number of jobs that may be scheduled simultaneously on the resource, and a *quantity*, representing the total amount available for consumption. A feasible schedule must not violate the capacity or quantity constraints of any resource.

Evaluation of HAZMAT Responses

In most planning domains, the goodness of a plan is a binary proposition—a plan either achieves or does not achieve its goals. However, this does not hold for HazMat plans. Because HazMat response occurs in real time, even the best possible response will be unable to prevent all undesirable effects. Some material will always be spilled, some harmful fumes released into the environment, and so on. The severity of the situation,

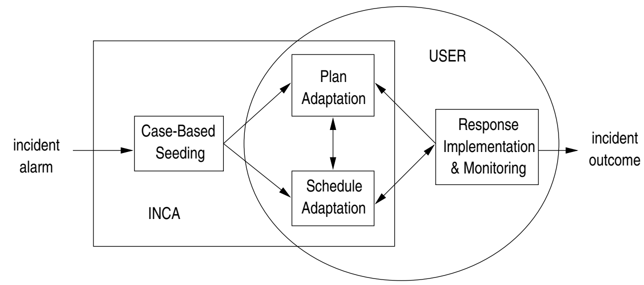

Figure 2: Interactive crisis response with Inca.

combined with the constraints placed by the available resources, also limits the space of attainable situations.

To evaluate various responses (including no response), HazMat uses a simulator. The simulator maintains processes that track and update the dynamic characteristics of the domain for a given incident. The state of the world is defined by numeric variables corresponding to the nominal features of HazMat incidents. Every action affects some of these variables and the state of the world is thus influenced by the specifics of the given incident and the actions initiated by the problem solver.

A Crisis Response Assistant

Rapid response in HazMat is desirable as delays typically result in more severe consequences—more material spilled, larger fires, higher health hazard levels, etc. The INteractive Crisis Assistant (Inca) uses case-based methods to seed the interactive response development process with initial candidate solutions drawn from previous cases. Preliminary experiments with Inca support the claim that case-based seeding leads to more rapid development of high-quality solutions (Gervasio et al., 1998a). Here, we focus on a description of the crisis response process with Inca.

System Overview

Figure 2 depicts the interactive crisis response process with Inca. Given a HazMat incident, Inca retrieves an appropriate case from the case library of previous solutions and performs some initial adaptation. The user then interactively adapts this response with Inca. This interaction is carried out through a menu-based, point-and-click graphical user interface that provides separate screens for planning and scheduling. Figure 3 shows the scheduling screen in the middle of an adaptation operation.

The graphical interface also serves as an interface to the HazMat simulator, letting the user implement and monitor responses. At any time, the user may *post* the schedule to begin execution of scheduled actions. The user may also request situation updates and continue interactively adapting the response with Inca. The response process ends when the crisis situation reaches a

[1] We expect these to change as HazMat and Inca are extended to more complex problem-solving scenarios.

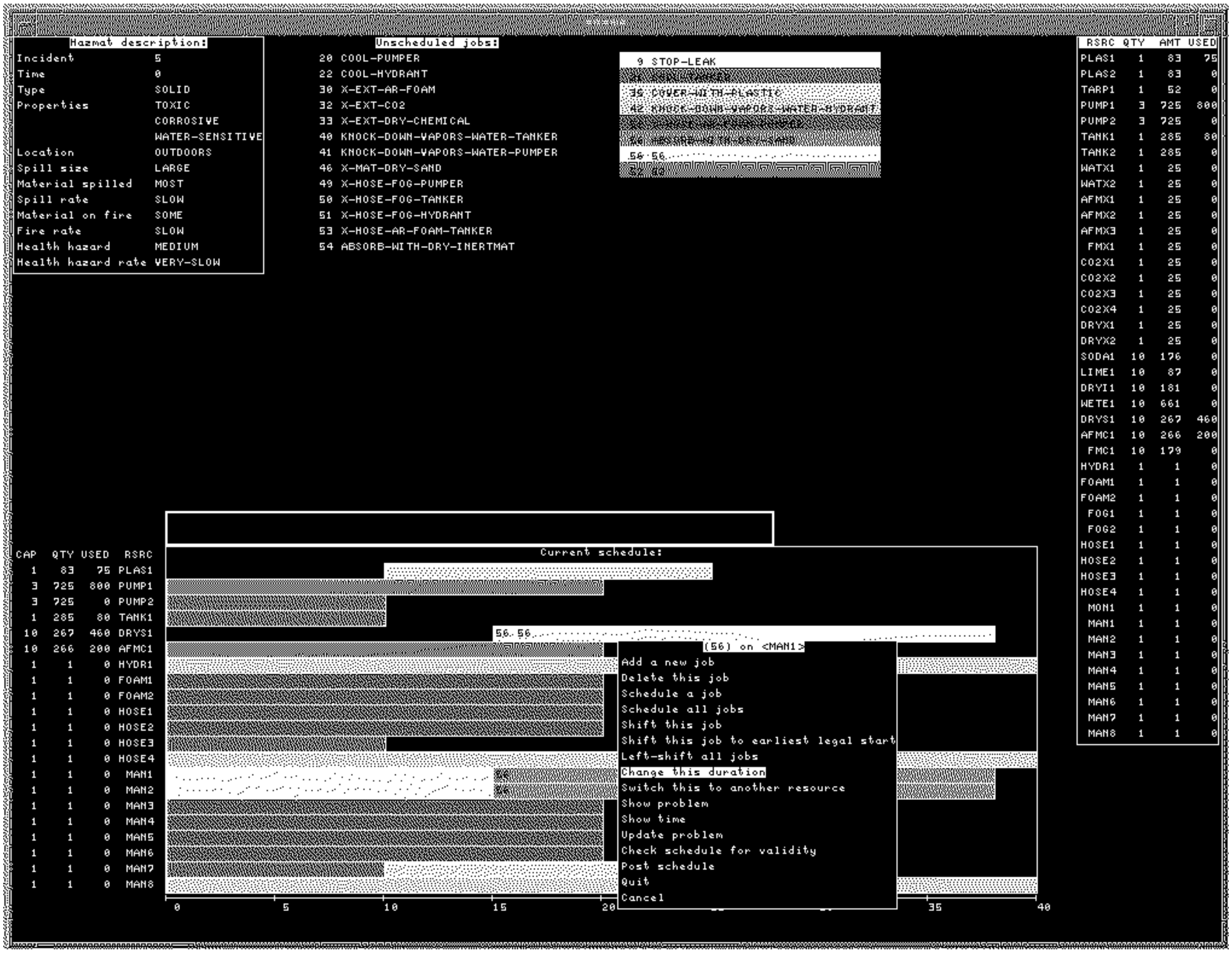

Figure 3: INCA scheduling screen, showing user about to change the duration of the `ABSORB-WITH-DRY-SAND` action, which currently has the dry sand (`DRYS`) resource overallocated.

stable point—either when the execution of all the scheduled actions successfully stops the spill and extinguishes the fire, or all the material is spilled and burned.

Case Retrieval

INCA is responsible for finding an applicable case from the case library and for performing an initial adaptation of the solution from the retrieved case. A *case* consists of a problem, a set of resources, a set of legal actions, a plan, and a schedule. An interesting characteristic of HAZMAT solutions is the significant overlap between the sets of legal actions for different types of hazardous material. Thus, different HAZMAT responses are, in general, more alike than they are different.

This has two main implications for the CBR component. First, it makes it difficult to find a set of indices that partition the cases into meaningful groups. We are currently using sufficiently small libraries (less than one hundred cases) that allow us to essentially avoid the indexing issue by performing similarity assessment over the whole library to find the most similar case. However, based on informal observations during our development of similarity metrics, we expect to eventually develop an indexing mechanism that primarily considers the set of legal actions for an incident.

Second, each incident matches many previous cases that provide reasonably good solutions. This allows us to achieve good performance even with simple matching mechanisms. But, in combination with the difficulty of finding good indices, this also makes it difficult to find the best case for a problem. However, we contend that providing a reasonably good initial solution rather than the best initial solution is sufficient, as subsequent phases allow the user to perform additional adaptations.

The components of a case may each be viewed as a feature vector, and similarity is determined using a simple count of matching features from a predefined subset. In the pilot study described in (Gervasio et al., 1998a), INCA retrieved the case sharing the most legal actions with the current problem. However, all the components of a case are available for matching.

Initial Case Adaptation

After the most similar case is retrieved, INCA performs some initial adaptation. Initial plan adaptation involves two operations: the removal of actions that were legal in the case problem but are illegal in the current problem, and the addition of actions that were illegal in the case problem but are legal in the current problem. This ensures that no illegal actions are executed and that no legal actions are excluded without the user's knowledge. This initial adaptation preserves the expansion of the case plan; any courses of action not pursued in the case problem remain so in the adapted plan. Newly legal actions remain unexpanded as well.

Initial schedule adaptation involves two steps: matching one-to-one the case resources and the resources for the current problem, and removing previously scheduled actions that have no corresponding resources in the current problem. Actions with no corresponding resources and newly legal actions are left unscheduled.

This approach to adaptation takes advantage of two aspects of the domain. First, because there are no causal supports between the actions of a plan, every action can be scheduled independently of other actions. Second, the resources are naturally grouped into pools, the members of which are completely substitutable. Thus, a resource of a particular type in one problem is just as good as another resource of the same type in another problem.

Interactive Plan Adaptation

Interactive adaptation allows the user to modify the initially adapted case plan in two ways. The first adaptation involves the expansion of an unexpanded node. This corresponds to exploring additional courses of action and has the effect of adding to the set of jobs available for scheduling. The second adaptation involves the deletion of any subtree of the plan. This has the effect of removing both scheduled and unscheduled jobs.

In expanding a node, INCA applies rules to filter out any illegal actions. These are the same rules used during initial case adaptation to remove any actions that were legal in the previous incident but are illegal in the current incident. These deleted actions are never shown to the user, and thus cannot be included in a response. INCA also has a heuristic planning mode in which it applies heuristics to filter out less useful subtrees. The user may override these deletions. Using these heuristics, INCA can also expand all nodes to autonomously generate a complete plan.

Although we discuss planning and scheduling separately, there is not a clear division of responsibilities. In contrast to the traditional planning and scheduling framework, the actions or jobs selected in the planning phase do not all have to be allocated to resources in the scheduling phase. Decisions about which actions to schedule (i.e., planning) can also be made during scheduling. However, the planning component can delete large groups of actions, thus limiting the size of the set of jobs to be considered by the scheduler. For example, if there already is a fire, removing the high-level node `prevent-fire` limits the scheduler's attention to the more relevant `handle-fire` actions.

Interactive Schedule Adaptation

As discussed earlier, scheduling a job involves choosing the number of resources to allocate to the job, the specific resources to allocate, a start time, and a duration. This is much less constrained than the traditional scheduling task, where the duration and the number of resources, and often the specific resources as well, are fixed. However, it is reflective of the HAZMAT domain, where variable amounts of resources may be allocated to a job, and where the desired effects of an action will be realized more quickly with more resources and with other simultaneous jobs affecting the same parameters.

Interactive adaptation lets the user modify the initially adapted case schedule in five ways. The user may add a job to the schedule, delete a job from the schedule, shift the start time of a scheduled job, change the duration of a scheduled job, or replace one of the specific resources assigned to a job. INCA participates in this adaptation by taking the user through the sequence of decisions necessary for performing the schedule repair. For example, in changing the duration of a job, INCA first asks whether the user wants to increase or decrease the duration, and then the amount of the change. INCA allows the user to consider oversubscribed or overallocated schedules—that is, schedules that violate a capacity or quantity constraint—during the development of a response. However, INCA will prevent the user from posting (executing) such infeasible schedules. Figure 3 shows the user about to repair an infeasible schedule by reducing the duration of an action that is currently using more than the available amount of dry sand.

As with planning, INCA has a heuristic scheduling mode in which it uses heuristics to suggest default values for resources, start times, and durations. These defaults are included in the menus of choices, so the user may accept or ignore them as desired. This mode also provides additional schedule repair operations such as shifting an infeasibly scheduled job to its earliest legal start time and automatically scheduling a job. Again, INCA can use these operations and heuristics to autonomously generate a schedule.

Related Work

Many crisis planning and scheduling systems today, including OPLAN-2 (Tate et al., 1994) and SOCAP (Bienkowski, 1996), have mixed-initiative modes that allow users to control the problem-solving process. While these interactive systems aid users in developing solutions from scratch, INCA aids users in adapting solutions from previous cases. CLAVIER (Hinkle & Toomey, 1994) is an advisory system for autoclave loading that, like INCA, retrieves previous cases for a user who then interacts with the system to perform any additional modifications. INCA differs in its domain and consequent focus on planning and scheduling.

DIAL (Leake, 1995) and CHARADE (Perini & Ricci, 1995) are examples of case-based systems for crisis domains. DIAL is an autonomous case-based planner for disaster response. DIAL's adaptation is fully automated, but for the purposes of crisis response, we believe that maintaining user input is crucial. CHARADE is a case-based system for developing first intervention plans for controlling forest fires. Like INCA, CHARADE uses a case library to seed subsequent planning and scheduling processes, which have autonomous as well as interactive components. While CHARADE is focused on the development of the initial response, however, we are designing INCA for the continuous development and evaluation of a response to an evolving crisis. In addition, automatic adaptation to different users, using machine learning techniques, will be an integral part of INCA.

CABINS (Miyashita & Sycara, 1995) is an interactive assistant that uses case-based methods to learn user preferences for job-shop scheduling. CABINS uses a heuristic scheduler to seed the scheduling process and employs case-based methods to learn individual preferences in the form of *repair cases*. In addition to the difference in application, INCA uses case-based reasoning as a seeding mechanism, analogous to the way in which organizations respond to crises using emergency response handbooks, and other inductive learning methods to learn user preferences. The hybrid planner for JMCAP (desJardins et al., 1998) uses cases both for plan seeding and subsequent plan adaptation. JMCAP is grounded in a maritime crisis domain and the planner has been extended to a distributed context. The planner's case library comprises a model of user preferences for plan generation and adaptation, whereas INCA's user model is distributed between its case library for schedule generation and the mechanism it learns for predicting user repair operations.

Future Work and Conclusions

We plan on extending INCA in various directions. INCA currently relies on a case-based reasoning component to seed a collaborative planning and scheduling process with the human user (Figure 2). Preliminary results support the utility of this approach (Gervasio et al., 1998a), and we plan to carry out more extensive experimentation to evaluate the case-base component. In the near future, we also plan to extend the interactive nature of INCA to the case retrieval process, which is currently entirely automated, to let the user's expertise and preferences influence the choice of a case seed.

Our ultimate goal is an *adaptive*, interactive crisis response assistant. INCA's interactions with the user provide a ready source of data regarding user preferences. Through the use of inductive learning techniques, INCA can learn user models and adapt its behavior to individual users to further improve efficiency. In (Gervasio et al., 1998b), we show that learning can be used to successfully predict user adaptation operations. We are currently investigating various other learning tasks for the assistant mechanism. In the near future, we plan to fully integrate learning into INCA and evaluate its adaptive capabilities.

We also hope to expand our software to support coordination among multiple crisis managers. This will involve detecting resource conflicts among different users' schedules and recommending steps to resolve those conflicts while still meeting each user's goals. Traces of such conflicts and their resolutions will again provide data for learning, which should let the system improve its ability to recommend resolutions that are likely to work for particular groups of users. This is a natural extension to the adaptive, interactive approach we have taken with individual crisis response, an approach to computational assistance that we believe will greatly facilitate the generation of efficient and effective crisis response.

Acknowledgments This work was partially supported by the Office of Naval Research under Grant N000014-96-1-1221. We would also like to thank Michael Fehling and Gregg Courand for many useful discussions on crisis; Stephanie Sage for implementing INCA's GUI; and Greg Trafton and David Aha for many useful discussions on INCA and HAZMAT.

References

Bienkowski, M. 1996. SOCAP: System for Operations Crisis Action Planning. In *Advanced Planning Technology*, Tate, A., ed. Menlo Park: AAAI Press.

desJardins, M.; Francis, A.; and Wolverton, M. 1998. Hybrid Planning: An Approach to Integrating Generative and Case-Based Planning. In *Working Notes of the AAAI-98 Workshop on Case-Based Reasoning Integrations.*

Gervasio, M.; Iba, W.; Langley, P.; and Sage, S. 1998a. Interactive Adaptation for Crisis Response. In *Working Notes of the AIPS-98 Workshop on Interactive and Collaborative Planning.*

Gervasio, M. T.; Iba, W.; and Langley, P. 1998b. Learning to Predict User Operations for Adaptive Scheduling. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence.*

Hinkle, D. and Toomey, C. 1994. CLAVIER: Applying Case-Based Reasoning to Composite Part Fabrication. In *Proceedings of the 6th Conference on Innovative Applications of Artificial Intelligence.*

Leake, D. B. 1995. Combining Rules and Cases to Learn Case Adaptation In *Proceedings of the 17th Annual Conference of the Cognitive Science Society*.

Miyashita, K. and Sycara, K. 1995. CABINS: A Framework of Knowledge Acquisition and Iterative Revision for Schedule Improvement and Reactive Repair. *Artificial Intelligence* 76: 337–426.

Perini, A. and Ricci, F. 1995. An Interactive Planning Architecture In *Proceedings of the European Workshop on Scheduling and Planning*.

Tate, A.; Drabble, B.; and Kirby, R. B. 1994. O-Plan2: an Open Architecture for Command Planning and Control. In *Intelligent Scheduling*, Zweben, M. and Fox, M. S. (eds.). San Francisco: Morgan Kaufmann.

Transport Canada, the U.S. Department of Transportation, and the Secretariat of Communications and Transportation of Mexico. *1996 North American Emergency Response Guidebook*.

Appendix

1. **Integration Name/Category**:

   INCA

2. **Performance Task**:

   Interactive generation of crisis response plans and schedules for hazardous materials incidents.

3. **Integration Objective**:

   Increased efficiency and increased solution quality. Efficiency is measured in terms of total response generation time (real time seconds). Solution quality is measured as the simulated improvement in the final situation using the constructed response relative to the final situation with no response. Situation is characterized in terms of parameters such as amount of material spilled, amount burned, and health hazard levels.

4. **Reasoning Components**:

   Hierarchical planner employing heuristic rules for filtering out unnecessary or dangerous actions, heuristic scheduler for choosing jobs (actions) and allocating resources, machine learning component for learning user preferences.

5. **Control Architecture**:

   CBR as slave, sequential. CBR supports and precedes planning and scheduling.

6. **CBR Cycle Step(s) Supported**:

   Retrieval, reuse.

7. **Representations**:

   HAZMAT *problems* formulated as feature vectors; *plans* as a degenerate form of hierarchical task networks (currently no variable and precedence constraints—semantically, an OR tree); and *schedules* as a set of resources, associated quantity and capacity constraints, and job reservation blocks corresponding to fixed-length time intervals for jobs on the resource. All three are used by the CBR, planning, and scheduling components. A *case* consists of a problem, plan, and schedule.

8. **Additional Components**:

   Human user, who has ultimate control over the response generation process and may thus accept or override any system suggestions.

9. **Integration Status**:

   CBR, planning, and scheduling components integrated and applied to synthetic hazardous material incidents domain (HAZMAT). Preliminary empirical evaluation performed on utility of case-based schedule seeding on HAZMAT response. Utility measured in terms of efficiency and solution quality (see integration objectives above).

10. **Priority Future Work**:

    More thorough experimentation with larger subject pool to replicate preliminary findings. Extension of interactive approach to case-based reasoning component. Integration of learning component into INCA. Development of learning techniques for constructing cases from reactive planning and scheduling episodes.