Using Guidelines to Constrain Interactive Case-Based HTN Planning

Héctor Muñoz-Avila†, Daniel C. McFarlane†, David W. Aha†,
Len Breslow‡, James A. Ballas‡, & Dana Nau‡
† Department of Computer Science
University of Maryland
College Park, MD 20742-3255
lastname@cs.umd.edu
(301) 405-2684 | FAX: 405-6707
‡ Navy Center for Applied Research in AI
Naval Research Laboratory
Washington, DC 20375
lastname@aic.nrl.navy.mil
(202) 404-4940 | FAX: 767-3172

Abstract

This paper describes HICAP, a general purpose and interactive case-based planning architecture. HICAP is a decision support tool for planning a hierarchical course of action. It integrates a hierarchical task editor, HTE, with a conversational case-based planner, NaCoDAE/HTN. HTE maintains a task hierarchy representing guidelines that constrain the final plan. HTE also encodes the hierarchical organization responsible for these tasks. This supports bookkeeping, which is crucial for real-world large-scale planning tasks. HTE can be used to activate NaCoDAE/HTN to interactively refine user-selected guideline tasks into a concrete plan. Our application of HICAP to the task of noncombatant evacuation operations inspired its architecture. In this application, our empirical evaluation with ModSAF simulations confirms that the plans output by HICAP outperform those generated using alternative approaches on three dimensions.

1 Introduction

Planning a course of action is difficult, especially for large hierarchical organizations (e.g., the US Navy) that assign tasks to elements (i.e., groups or individuals) and constrain plans with guidelines (e.g., doctrine). In this context, a concrete plan must adhere to guidelines but should also exploit organizational knowledge (e.g., standard procedures for solving tasks, previous experiences when reacting to unanticipated situations) where appropriate. Case-based reasoning (CBR) can be used to capture this knowledge.

†NCARAI Technical Note AIC-99-004
In large planning environments, automatic plan generation is neither feasible nor desirable because users must observe and control plan generation. We argue that, rather than relying on an automatic plan generator, users prefer and can greatly benefit from the assistance of an intelligent plan formulation tool with the following characteristics:

- **Guidelines-driven**: Uses guidelines to constrain plan generation.
- **Interactive**: Allows users to edit any detail of the plan.
- **Provide Case Access**: Indexes plan segments from previous problem-solving experiences, and retrieves them for users if warranted by the current planning scenario.
- **Perform Bookkeeping**: Maintains information on the status of and relations between task responsibilities and individuals in the organizational hierarchy.

This paper describes HICAP, a general-purpose plan formulation tool that we designed to meet these characteristics.² HICAP (Hierarchical Interactive Case-Based Architecture for Planning) integrates a task decomposition editor, HTE (Hierarchical Task Editor) (Muñoz-Avila et al., 1998), with a conversational case-based planner (CCBP), NaCoDAE/HTN. The former allows users to edit and select guidelines for refinement, while the latter allows users to interactively refine HTN plans. Refinements use knowledge of previous operations, represented as cases, to augment or replace standard procedures.

The following sections describe the application task, HICAP’s knowledge representation, its architecture, an empirical evaluation, and a discussion of related work.

## 2 Planning Noncombatant Evacuation Operations

Noncombatant evacuation operations (NEOs) are conducted to assist the USA Department of State (DoS) in evacuating noncombatants, nonessential military personnel, selected host-nation citizens, and third country nationals whose lives are in danger from locations in a host foreign nation to an appropriate safe haven. They usually involve a swift insertion of a force, temporary occupation of an objective (e.g., a USA Embassy), and a planned withdrawal after

---

²Implemented in Java 2, the HICAP applet can be run from www.aic.nrl.navy.mil/~munoz/hicap. HICAP was introduced in (Muñoz-Avila et al., 1999), which did not include the evaluation described here.
mission completion. NEOs are usually planned and operated by a Joint task force (JTF), a hierarchical military organization, and conducted under an Ambassador’s authority. Force sizes can range into the hundreds and involve all branches of the armed services, while the evacuees can number into the thousands. More than ten NEOs were conducted within the past decade. Publications describe NEO doctrine (DoD, 1994), case studies (Siegel, 1991; 1995), and more general analyses (e.g., Lambert, 1992).³

The decision making process for a NEO is made at three increasingly-specific levels: strategic, operational and tactical. The strategic level involves global and political considerations such as whether to perform the NEO. The operational level involves considerations such as determining the size and composition of the force executing the NEO. The tactical level is the concrete level, which assigns specific resources to specific tasks.

JTF commanders plan NEOs in the context of doctrine (DoD, 1994), which defines general guidelines (e.g., chain of command, task agenda) for designing strategic and operational plans; tactical considerations are only partly addressed. Doctrine is abstract; it cannot account for characteristics of specific NEOs. Thus, JTF commanders must always adapt doctrine to a NEO’s specific needs, and does so in two ways. First, they dynamically modify doctrinal guidance by eliminating irrelevant planning tasks and adding others, depending on the operation’s needs, resource availabilities, and relevant past experiences. For example, although NEO doctrine states that a forward command element must be inserted into the evacuation area prior to the primary evacuation elements, this is not always feasible (e.g., Siegel, 1991). Second, they employ experiences from previous NEOs, which complement doctrine by suggesting tactical refinements suitable for the current NEO. For example, they could draw upon their previous experiences to identify whether it is appropriate to concentrate the evacuees in the embassy or to plan on multiple evacuation sites.

3 Knowledge Representation

Because HTNs (hierarchical task networks) (Erol et al., 1994) are expressive representations for plans, we used a variant of them in HICAP. An HTN is a set of tasks and their ordering relations, denoted as $N = \langle \{T_1, \ldots, T_m\}, \prec \rangle$ ($m \geq 0$). The relation $\prec$ has the form $T_i \prec$…

³See www.aic.nrl.navy.mil/~aha/neos for more information on NEOs.
Problem solving with HTNs occurs by applying methods to decompose tasks into subtasks. Each method has the form $M = \langle l, T, N, P \rangle$, where $l$ is a label, $T$ is a task, $N$ is an HTN, and $P = \langle p_1, \ldots, p_k \rangle$ is a set of preconditions for applying $M$. When $P$ is satisfied, $M$ can be applied to a task $T$ to yield $N$.

HICAP’s HTN consists of three task types. First, non-decomposable tasks are concrete actions and can occur only at a network’s leaves. Next, uniquely decomposable tasks correspond to guideline tasks (e.g., doctrine), and are solved by unconditional methods ($k = 0$). Finally, multiply decomposable tasks must be solved in a specific problem-solving context.

Methods for problem-specific tasks are represented as cases, which encode preconditions as a set of question-answer pairs. Cases are obtained from manuals describing standard procedures or reports detailing previous problem-solving episodes. When solving a task $T$, HICAP retrieves all cases that can decompose $T$. If all the preconditions of a standard procedure are met, then it should be used to decompose $T$. Otherwise, a case corresponding to the most similar episode should be used. For example, standard NEO planning procedures state that the evacuees must be concentrated in the embassy prior to troop deployment. This is not always possible; escorted transports were organized after the joint task force was deployed in Eastern Exit (Siegel, 1991), and the evacuees in Sharp Edge (Sachtleben, 1991) were concentrated in several places, forcing multiple separate evacuations.

4 HICAP: An Interactive Case-Based Planner

HICAP (Figure 1), which integrates HTE with NaCoDAE/HTN, inputs an HTN describing the guidelines for an application (i.e., with all uniquely decomposable tasks expanded and displayed), along with a set of cases for each subtask that can be decomposed in multiple ways. Under user control, it outputs an edited HTN whose leaves are concrete actions as specified by case applications. In this way, HICAP satisfies the requirements stated in Section 1. First, all plans formulated using HICAP are in accordance with the guidelines or user modifications. Second, HICAP interactively supports task editing, triggering conversations for tasks that can be decomposed by case application. Third, it incorporates knowledge from previous problem solving episodes as cases, which serve as task decomposition alternatives. Finally, it allows users to visually check that all tasks are assigned to JTF elements.
4.1 Hierarchical Task Editor

In complex environments where dozens of tasks must be performed by many people, tracking the completion status for each task can be challenging. For example, during the NEO Operation Eastern Exit, the task to inspect evacuees prior to embarkation was not assigned (Siegel, 1991). One of the evacuees produced a weapon during a helicopter evacuation flight. Although it was immediately confiscated, this oversight could have resulted in tragedy and illustrates the difficulties with planning NEOs manually.

The Hierarchical Task Editor (HTE) (Muñoz-Avila et al., 1998) serves HICAP as a bookkeeping tool to track the status of each task. HTE inputs a knowledge base consisting of an HTN task agenda, its ordering relations, the organization’s command hierarchy, and an assignment of tasks to command elements. It allows users to edit the knowledge base and select tasks to refine by invoking NaCoDAE/HTN, thus tailoring the knowledge base to the particular circumstances of the current NEO.

For our NEO application, we encoded an HTN to capture critical planning doctrine (DoD, 1994), yielding 200+ tasks and their ordering relations. Next, we elicited the JTF command hierarchy commonly used in NEO operations. Finally, we elicited relations between tasks
Prepare safe haven
Organize and process evacuees
Perform evacuation
Insert JTF main body
Insert forward command element
Prepare ISB

JTF Commander
Safe Haven Force
ISB Force
JTF Main Body
Forward Command Element
JTF Support Force

Figure 2: Top level NEO tasks and their assignment to JTF command elements (double arrows denote assignments; arrows denote task orderings; ISB = intermediate stage base).

and the JTF elements responsible for them. The mapping of tasks to command elements is many-to-one. Figure 2 displays (left) the top level tasks that, according to doctrine, must be performed during a NEO and (right) the elements in the JTF responsible for them.

4.2 Conversational Task Decomposer

NaCoDAE/HTN, an extension of the NaCoDAE conversational case retrieval tool (Aha & Breslow, 1997; Breslow & Aha, 1997), supports HTN planning by allowing users to refine selected tasks into concrete actions. When given a task $T$ to refine by HTE, NaCoDAE/HTN uses $T$ as an index for case retrieval and conducts an interactive conversation, which ends when the user selects a case $C = \langle l, T, N, P \rangle$. Network $N$ is then used to decompose $T$. Sub-tasks of $N$ might themselves be decomposable, but non-decomposable tasks corresponding to concrete actions will eventually be reached. Task expansions are displayed by HTE.

During conversations, NaCoDAE/HTN displays the labels of cases that can decompose the selected node and the questions from these cases whose answers are not yet known for the
current situation. The user can select and answer any displayed question; question-answer pairs are used to compute the similarity of the current task with its potential decomposition methods. Cases are ranked according to their similarity (Aha & Breslow, 1997), while questions are ranked according to their frequency among the displayed cases. Answering a question modifies the case rankings and the displays. A conversation ends when the user selects a case for decomposing the current task.

Some of the displayed cases are standard procedures; they can only be selected to decompose a task after all of their questions have been answered and match the current planning scenario. That is, preconditions of the standard procedures must match before they can be applied. In contrast, cases based on previous experiences can be selected even if some of their questions have not been answered, or if the user’s answers differ. Thus, they support partial matching between their preconditions and the current planning scenario.

5 Example: NEO Planning

During NEO planning, the user views the top level tasks first, revising them as needed. They can decompose any task and view its decomposition. In Figure 3, the user has selected the
Figure 4: NaCoDAE/HTN: Before (left) and after (right) answering a question. The top window lists possible answers to selected questions, while the lower windows display the ranked questions and cases.

*Select assembly areas for evacuation & ECC (Evacuation Control Center) sites* task, which is highlighted together with the command element responsible for it.

Standard procedure dictates that the embassy is the ideal assembly area. However, it is not always possible to concentrate the evacuees in the embassy. Alternative methods can be considered for decomposing this task. When the military planner selects this task, HICAP displays the alternatives and initiates a NaCoDAE/HTN conversation (see Figure 4 (left)).

If the user answers *Are there any hostiles between the embassy and the evacuees?* with *uncertain*, a perfect match occurs with the second ranked case (Figure 4 (right)). Figure 5 shows the decomposition when selecting it to decompose this task; two new subtasks are displayed, corresponding to this case’s decomposition network. *Send Unmanned Air Vehicle to …* is a non-decomposable concrete action. If the user tells HICAP to decompose *Determine if hostiles are present*, HICAP will initiate a new NaCoDAE/HTN dialogue (Figure 5).

The user can prompt another dialogue by selecting the *The UAV detects hostiles* alternative and decomposing its subtasks. This cycle, in which HICAP displays alternatives and the user answers questions and selects an alternative, continues until non-decomposable tasks (i.e., concrete actions) are reached, which form part of the final plan.
6 Empirical Validation

An experiment was run to test HICAP’s effectiveness in choosing successful plans for an example NEO subtask. Two researchers performed the experiment: one operated a military simulator (MCSF) while the other operated HICAP. A strict blind was imposed to ensure that the HICAP user had no knowledge concerning the simulated hostile forces; this tests HICAP’s utility for planning under realistic situations where decision makers have uncertain information about the state of the world. We hypothesized that HICAP would allow users to choose a relatively successful plan from among known tactical options. The relative success of HICAP’s selected plan was evaluated relative to three other methods for selecting plans: *random choice*, *heuristic choice*, and choice by the *most frequently used plan* in past NEOs.

6.1 The ModSAF Simulation System

We used Marine Corps SAF (MCSF), a variant of ModSAF (Modular Semi-Automated Forces), to evaluate the quality of NEO plans elicited using HICAP. ModSAF, developed by the USA Army to inject simulated auxiliary forces into training exercises, has been deployed to simulate real-world military scenarios (Ceranowicz, 1994). It is a finite state simulation with modular components that represent individual entities and parts of entities. For example, a simulated tank would have physical components such as a turret. It would
also have behavior components that represent its nominal tasks such as move, attack, target, react to fire, etc. Certain 3D aspects are also represented (e.g., terrain elevation, tree and vegetation, rivers, oceans, atmospheric conditions), which can affect sensory and movement behavior. The realism of ModSAF/MCSF simulations is sufficient for training exercises.

Figure 6’s MCSF snapshot displays a simulated USA Embassy, a host country government compound, and some simulated objects. For example, a simulated transport helicopter is positioned at the heliport within the Embassy site.

MCSF is a nondeterministic simulator that models multiple sources of stochastic variation. Some events are determined by a random number generator; others are highly sensitive to the initial startup conditions. MCSF simulates the behavior of military units in context as they follow given tactical orders. Therefore, MCSF can simulate simplified NEO subtasks in which a single planning decision determines tactical orders.
6.2 Experimental Setup

We created a NEO subtask for this evaluation concerning how to move 64 evacuees from a meeting site to a US embassy. The meeting site was at a crossroads in an uninhabited area outside of a city. Evacuees had to be transported (8 per vehicle) through this undeveloped area, which had heavy tree cover, and out through a city to the US embassy. Evacuees had to pass near a local government complex to enter the US embassy grounds. This NEO context requires only a single tactical plan decision with four distinct planning solutions:

1. Land evacuation using 8 armored trucks
2. Land evacuation using 8 armored trucks with an 8 tank escort
3. Air evacuation using 8 transport helicopters
4. Air evacuation using 8 transport helicopters with an 8 attack helicopter escort

The kind of military units used in the simulation are typical of those available to the Marine Expeditionary Units (MEU’s) that frequently perform NEO’s. A detailed terrain database of the Camp Lejeune (North Carolina, USA) area was chosen to simulate the environment. We chose this location because real MEU’s train there for NEOs.

Two scenarios were defined that were identical except for the type of hostile forces. All hostiles were two-person dismounted infantry teams. Hostile teams in both scenarios were armed with two automatic rifles and a portable missile launcher. Each scenario included only one type of missile for hostile teams (i.e., either anti-tank missiles or anti-air missiles, but not both). These types of infantry teams positioned in an urban environment are typical of the kinds of hostile forces encountered in real NEO’s. The positions of the hostile teams were the same for both scenarios and selected to ensure that the opposing forces will meet.

All four plan options were simulated ten separate times for each of the two scenarios. This resulted in 80 (2 × 4 × 10) total MCSF simulations. Each of the eight plan-and-scenario combinations was repeated ten times because MCSF is nondeterministic. For example, slight differences produced by MCSF’s stochastic movement models resulted in very different formations of friendly units when they first encountered the hostile teams. These differences often lead to drastically different simulated battle outcomes.

The HICAP user had no knowledge of the scenarios being tested; scenario information was gradually extracted through the questions prompted by NaCoDAE/HTN. That is, case-based planning was done with incomplete information about the world. Furthermore, the
Table 1: Summaries of casualties (mean & standard deviation) from 80 MCSF simulations.

<table>
<thead>
<tr>
<th>Tactical Plans</th>
<th>Scenario 1</th>
<th>Scenario 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Evacuees</td>
<td>Friends</td>
</tr>
<tr>
<td>Land</td>
<td>6.4</td>
<td>5.1</td>
</tr>
<tr>
<td>Land/Escort</td>
<td>3.2</td>
<td>10.1</td>
</tr>
<tr>
<td>Air</td>
<td>56.0</td>
<td>9.2</td>
</tr>
<tr>
<td>Air/Escort</td>
<td>0</td>
<td>0.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6.3</td>
</tr>
</tbody>
</table>

Effects of actions were uncertain; the only way to learn the effects of an action was to actually execute it. This contrasts with traditional planning approaches that assume an action’s effects are known a priori (Fikes and Nilsson, 1971).

### 6.3 Results

Table 6.3 summarizes the casualty results for the 80 total simulations, which each required approximately 15 minutes to run. The success measures were taken from the US Navy’s Measures of Effectiveness (MOE’s) published in the Universal Naval Task List. Recommended MOEs are specified for evaluating each kind of military operation. There are several MOE’s for the tactical aspects of NEO’s; three were chosen as most important for evaluating the results of this experiment: (1) number of evacuees safely moved, (2) number of casualties to friendly forces, and (3) number of casualties to hostile forces.

HICAP did not choose the same tactical plan for both scenarios. For the first (anti-tank) scenario, it chose to move the evacuees by helicopter with an attack helicopter escort. For the second (anti-air) scenario, it chose to move evacuees by armored truck with a tank escort. HICAP’s CCBP method was evaluated by comparing the success of its chosen plans to plans chosen by the other three plan selection methods. The random choice method simply averaged the results of all four plans. The heuristic choice method always sent an escort, and its results were the average of the two escort plans. The most frequently used plan method for this subtask in recent NEOs was to move evacuees using escorted land vehicles.

Figure 7 compares the effectiveness of the four selection methods. Overall, HICAP’s
Figure 7: Comparison of plan selection methods using Navy MOEs for NEOs.
CCBP method selected plans of higher quality than the other methods because its plan selection decisions are tailored to the characteristics of each scenario.

7 Related Research

Case-based planning (CBP) has been the subject of extensive research (Bergmann et al., 1998). Our research is closely related to studies on hierarchical CBP (Kambhampati, 1993; Bergmann & Wilke, 1994; Branting & Aha, 1995). HICAP differs from these other approaches in that it includes the user in its problem solving loop. This is particularly important for applications like NEO planning, where automatic tools are unacceptable.

Some researchers have used CBP with HTNs for military tasks. For example, Mitchell (1997) used integrated CBP to select tasks for a tactical response planner. NEO planning requires that each task be addressed - no choice is involved - and we use CBP to instead choose how to perform a task. MI-CBP (Veloso et al., 1997) uses rationale-directed CBP to suggest plan modifications, but does not perform doctrine-driven task decomposition. HICAP’s interactions instead focus on retrieval rather than plan adaptation and learning.

8 Future Work and Conclusion

The HICAP case-based planner helps users to formulate a course of action for hierarchical tasks. It is the first tool to combine a task guideline decomposition process with CBR to
support interactive plan formulation. It yields plans that benefit from previous experiences and are sound according to predefined guidelines. HICAP also supports experience sharing, thus allowing planners to exploit knowledge from other planning experts. These design characteristics enhance HICAP’s acceptance by military planning personnel.

We are currently integrating HICAP with a generative HTN planner that can evaluate numeric expressions, which is particularly important for NEOs because decisions often depend on resource capability and availability (i.e., determining whether a helicopter requires in-flight refueling for a given mission). HICAP will serve as the plan formulation component for SPAWAR’s Interactive Decision Support (IDS) system. When completed, IDS will perform distributed NEO plan formulation, execution, monitoring, and replanning.

Acknowledgements

This research was supported by grants from the Naval Research Laboratory, the Army Research Laboratory, and the Office of Naval Research.

References


Kambhampati, S. (1994). Exploiting causal structure to control retrieval and refitting during plan reuse. Computational Intelligence, 10, 213–244.


