ENCODING PLANNING KNOWLEDGE FOR RECOGNITION, CONSTRUCTION, AND LEARNING

Charlie Dolan
Michael G. Dyer

April 1986
CSD-860074

Encoding Planning Knowledge for Recognition, Construction, and Learning*

Charlie Dolan
Artificial Intelligence Lab
Computer Science Department
UCLA
and
Hughes Aircraft AI Center

Michael Dyer
Artificial Intelligence Lab
Computer Science Department
UCLA

Topic Area: Learning, Knowledge Representation
Key Words: planning, constraints, chunking

Address:
UCLA
AI Lab
Boelter Hall
Los Angeles, CA, 90024
ARPA: cpd@UCLA-LOCUS.ARPA

Abstract

This paper discusses a method for representing thematic level structures, i.e. abstract plan/goal combinations. We make the case that the processes for recognition and construction of plans use the same memory structures. In particular, we are looking at the knowledge structures for recognizing and avoiding bad planning. The learning procedure we describe starts by observing the bad planning behavior of narrative characters and combines old descriptions of planning errors to create new abstract structures. The learning method discussed is a one-trial, schema acquisition method, which is similar to DeJong's [DeJong, 1983]. The method used involves taking schemas for planning situations that are found in an actual narrative situation, and using constraint-based causal reasoning to construct a new schema which better characterizes the situation. This work is part of the MORRIS project at UCLA [Dyer, 1983a]. The planning situations are represented using Thematic Abstraction Units (TAUs) [Dyer, 1983].

* This work was supported in part by grants from the Hughes Aircraft AI Center, the UCLA Center for Experimental Computer Science, and the Keck Foundation.

1 Introduction

In the real world, tasks cannot always be accomplished by using simple subgoal partitioning and recursive problem analysis. Both real world agents and narrative characters often must apply plans that require cooperation from other agents, adjust plans that conflict with an agent's concurrent goals, and manage plans which contribute simultaneously to more than one goal. A classification of real world plans is found in [Schank and Abelson, 1977], and a taxonomy of goal/plan interactions can be found in [Wilensky, 1978].

There are two reasons for looking at poor planning behavior. (1) Knowledge structures which encode poor planning can be used to critique plans and point-out possible weak points. In this way they are similar in function to Sussman's planning critics [Sussman, 1973]; however, they are much more varied than the fixed number of very general critics Sussman suggests, and as this paper shows, they are learnable. (2) It is important to represent planning errors, not only to critique plans, but also so that in counterplanning situations [Carbonell, 1979], a planner can try to trick another agent into making a planning error. Any situation which is bad for a planner is generally a good situation into which to force an opponent.

Successful real world planners often use adages to guide them in avoiding bad plans. Adages warn against both specific and general planning errors. *Poor Richard's Almanac* [Franklin, 1733-1758] gives many examples of adages, such as "A stitch in time saves nine" (error avoidance), and "Don't burn your bridges behind you." (error recovery).

Dyer [1983, 1981] showed how a class of planning errors could be represented by Thematic Abstraction Units (TAUs), and how these planning errors might be recognized in stories. This paper will present a representation for planning errors that also facilitates the combination of planning descriptions into new thematic structures and use of thematic structures in planning.

The combination method requires an example narrative situation that contains a new planning error. The input example is conceptually analyzed to discover whether known planning errors which are present in the example can be combined into new structures and then to establish the links between the component structures that make up the newly discovered planning error. TAUs are used in planning by treating an agent's plans as if they were a story and using the TAU recognition process to point out potential planning errors. Depending on the planning error detected, a particular set of heuristics is applied to try and fix the plan.

In addition to the representations for TAUs, goals, and plans mentioned above, the examples here also rely on Schank's Conceptual Dependency theory [Schank, 1972]. Other relevant work on memory organization includes [Schank, 1982], [Kolodner, 1980], and [Lebowitz, 1980].

2 An Example Planning Situation

Here we will see a planning situation that contains three planning errors whose abstract descriptions are already known to the system. From this situation we will see how we can generate two specializations of planning errors, and one novel planning error construct. The situation is taken from Aesop's fables; the version below is taken from Bewick's collection [Bewick, 1973].

Other work dealing with Aesop's fables is TALESPIN [Meehan, 1979], a program which generates stories by simulating a character's planning and problem-solving behavior. TALESPIN lacked a theory of planning error analysis. As a result, although **The Fox and the Crow** story was generated, it was accomplished only in an ad hoc manner by using a pre-defined, non-extensible representation of that specific planning situation.

The Fox and the Crow

> The Crow was sitting in the tree with a piece of cheese in her mouth. The Fox walked up to the bottom of the tree and said to the Crow, "Crow what a beautiful voice you have; please sing for me." The Crow was very flattered and began to sing. When she did, the cheese dropped out of her mouth. The Fox grabbed the cheese and ran away laughing.

Note that this story can be looked at in two ways: (1) as an instance of bad planning on the part of the Crow and (2) as an instance of good planning on the part of the Fox. We will show how to recognize a planning error from its representation, how to learn new planning errors, and how to apply planning information to avoid goal failures.

The first of the three planning errors we will discuss is the most basic. When the Crow sings, she does not realize that she is already using her mouth to hold the cheese. This planning error is characterized at an abstract level by TAU-CONF-ENABL (confused enablement). The full representation for TAU-CONF-ENABL is given below.

In each TAU, the representation of a planning error situation consists of two parts: (1) the **binding-spec**: a list of conceptual patterns which occur in the story; (2) the **constraints**: a list of logical constraints among the patterns occurring in the **binding-spec** and other concepts from the story.

```
     TAU name TAU-CONF-ENABL
          binding-spec
            [?standing-goal (P-GOAL actor ?x
                                    obj (POSS-BY actor ?x
                                                 obj ?y)
                                    manner FAIL)]
-->         [?interfering-goal2 (GOAL actor ?x)]
            [?mistake (ACT actor ?x)]
          constraints
            intention(?interfering-goal,?plan),
            realization(?mistake,?plan),
            resulting(?mistake,?disabling-state),
            achievement(?desirable-state,?standing-goal1),
            disablement(?disabling-state,?desirable-state).
```

The abstract situation this structure characterizes is one where an agent has a goal, **?standing-goal**, which has failed and where the goal was to preserve possession of some object. The cause of the goal failure is an act, **?mistake**, which attempted to accomplish another goal, **?interfering-goal**.

This TAU serves as a description of a planning error and as such can be used as planning advice. Previous research has shown how constraints can be used for problem solving [Steele, 1980]. The constraints presented here are easily implemented as a logic program [Kowalski, 1979].

The processes of recognizing and indexing TAUs are covered more fully in [Dyer 1983] and [Dolan 1984]. Dolan [1984] gives a detailed explanation of the comprehension process and memory model which allows the recognition of TAUs in this format.

Briefly, the concept which is indicated with the **'-->'**, is the concept which triggers recognition of this TAU. In this case, the process for recognizing this TAU is indexed under a certain **GOAL** type which the program has observed to precede this type of planning error. Dyer [1983] gives some criteria for optimally locating this "indexing point" based on the importance of the goal and the frequency of the situation.

After this TAU is triggered, the recognition program looks for for the concepts in the **'binding-spec'** in the story. The concepts may be either explicitly mentioned or deduced from others. While searching for concepts which match the patterns, the program also ensures that the constraints listed under **'constraints'** are statisfied. The program also uses the ordering of the concepts in **'binding-spec'** as a heuristic to guide the search for concepts in the story.

As we mentioned above, **The Fox and the Crow** instantiates two other TAUs: (1) TAU-VANITY is the planning error of allowing personal vanity to dictate plan choice; (2) TAU-ULTERIOR is the planning error of not considering another agent's possible motives before acting. Both of these TAUs display an important characteristic of TAUs as planning heuristics; not only do TAUs provide admonitions against bad planning, but they can also be turned around and used as plans to try and force other agents into situations where they will make mistakes.

These TAUs can be combined to form new planning heuristics. There are two key problems in TAU acquisition:

1. How does a program know which TAUs to select and examine for combination attempts?

2. Once selected, how are TAUs actually combined to form new planning and indexing structures?

Both 1. and 2. are non-trivial. A sophisticated planner will have numerous stories indexed by multiple TAUs in memory. Attempting to combine TAUs arbitrarily would lead to combinatoric problems. Fortunately, memorable stories (such as Aesop's fables) are designed to give novel planning advice through illustrating planning errors. Thus, TAU selection can be governed by the following strategy:

WHENEVER two TAUs share concepts in an observed
planning situation,
TRY to combine them to form a novel planning construct

This heuristic can only be applied after reading a story. The comprehension of the story thus makes available TAUs for combination and indicates which concepts are shared. Even when two TAUs share concepts, however, they may not be able to be combined.

There are two ways to combine TAUs based on the way they share concepts:

(1) specialization,
(2) combination (chunking).

Recent work in specialization learning includes [DeJong, 1983], [Lebowitz, 1980], and [Kolodner, 1980]. Both workers formulate methods for creating new planning knowledge through specialization, but do not have a method extensible to chunking. Most research in learning by chunking has been in domains where there is no counter-planning [Laird, 1984].

A detailed discussion of the learning process for TAUs can be found in [Dolan and Dyer, 1985]. Here we will simply outline the procedure and highlight the features of the representation which facilitate learning.

3 Creating New TAUs through Specialization

TAU-ULTERIOR this TAU represents the situation in which **?y** tells **?x** information that motivates **?x** to perform an act which results in the disablement of one of **?x**'s goals, while at the same time achieving a goal of **?y**'s which **?x** did not forsee.

```
    TAU name TAU-ULTERIOR
1       binding-spec
2         [?standing-goal (GOAL actor ?x
3                               status FAILED)]
4         [?desirable-state (STATE)]
5         [?mtrans (MTRANS actor ?y
6                          to ?x
7                          obj ?z)]
8         [?mistake (ACT actor ?x)]
9       constraints
10        achievement(?desirable-state,?standing-goal),
11        motivation(?z,?interfering-goal),
12        achievement(?mistake,?interfering-goal),
13        disables(?mistake,?desirable-state),
14        goal-conflict(?goal,(GOAL actor ?y
15                                  obj ?w)),
16        achievement(?mistake,(GOAL actor ?y
17                                   obj ?w)),
18        not-obvious-result(?mtrans,?mistake).
```

From the description of TAU-ULTERIOR we can see that in some sense is contains TAU-CONF-ENABL because all of the concepts from the **'binding-spec'** of TAU-CONF-ENABL are present in TAU-ULTERIOR. The only concept which TAU-ULTERIOR has that TAU-CONF-ENABLE doesn't is lines 5-7, "**?y** tells **?x** ...". If we examine TAU-CONF-ENABL, we see from the way **The Fox and the Crow** instantiates both this TAU and TAU-ULTERIOR that they share the following concepts:

?standing-goal --> to keep the cheese
?interfering-goal --> to show-off
?mistake --> singing.

The other concept in the **binding-spec** of TAU-ULTERIOR is,

?mtrans --> the Fox telling the Crow she has a nice voice

All of the components from TAU-CONF-ENABL are shared with TAU-ULTERIOR, but TAU-CONF-ENABL has additional constraints. we say the TAU-ULTERIOR **contains** TAU-CONF-ENABL. The containment relation allows us to create a new TAU, a **specialization** of TAU-ULTERIOR. The new TAU is formed by taking the extra constraints from TAU-CONF-ENABL ("extra" meaning those not already in TAU-ULTERIOR) and conjoining them with those in TAU-ULTERIOR. This TAU, which we will call TAU-MISDIRECTED-ENABL is used to characterize situations where **?x** says something to **?y** to deflect **?y**'s attention away from an an enablement condition on a current goal. This TAU is also used to process stories such as the following one,

Joey's Waffle

Joey was sitting at the breakfast table about to eat his waffle. His sister Mary said, "Look Joey, Haley's comet!". When Joey looked up Mary grabbed the waffle and started eating it.

Here the goal which failed for Joey was to maintain possession of the waffle. The enablement condition on that goal was to attend his eyes to the waffle and guard it. The condition was removed when he looked up.

In some sense when one TAU contains another as with TAU-ULTERIOR and TAU-CONF-ENABL, the contained TAU mediates the application of the containing TAU. An exmaple of how another TAU can mediate for TAU-ULTERIOR is,

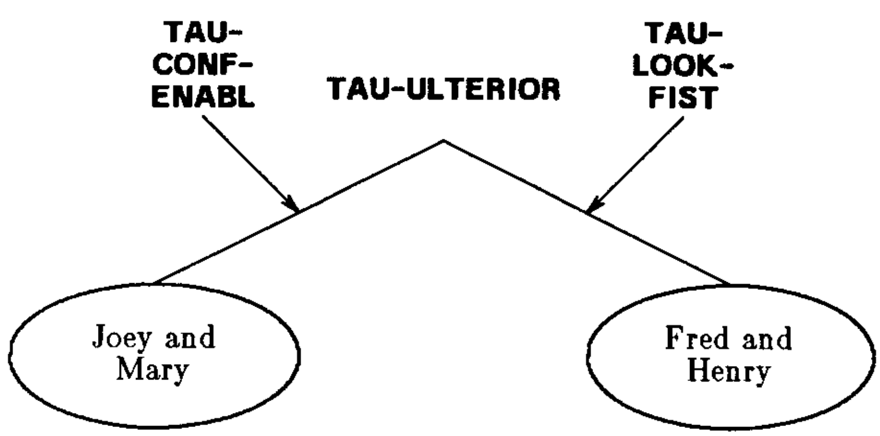
Bogus Job Advice

Fred was a post-doc working at the college. He told his boss, Henry, that there was a faculty position open at the university across town. Henry decided to give his notice at the college and apply for the job. When Henry left, Fred took his position at the college. As it turned out, Henry was not quite right to the new position and didn't get the job.

In this case, TAU-UTLERIOR is mediated by the TAU which encodes the adage, "Look before you leap." TAU-MISDIRECTED-ENABL, shown below, is planning advice specific to possession goals and ulterior motives.

```
TAU name TAU-MISDIRECTED-ENABL
    binding-spec
      [?standing-goal (GOAL actor ?x
                            status FAILED)]
      [?desirable-state (STATE)]
      [?mtrans (MTRANS actor ?y
                       to ?x
                       obj ?z)]
      [?mistake (ACT actor ?x)]
    constraints
      achievement(?desirable-state,?standing-goal),
      motivation(?z,?interfering-goal),
      intention(?interfering-goal,?plan),
      realization(?mistake,?plan),
      achievement(?mistake,?interfering-goal),
      disables(?mistake,?desirable-state),
      goal-conflict(?goal,(GOAL actor ?y
                                obj ?w)),
      achievement(?mistake,(GOAL actor ?y
                                 obj ?w)),
      unforseen(?mtrans,?mistake).
```

The way the two other TAU mediate TAU-ULTERIOR is show in the diagram below,



We see that this organization of TAUs points out the differences and similarities of the two stories on a thematic level.

4 Creating New TAUs through Combination

As we saw above, we can get a non-trivial specialization of a TAU by discovering containment in a particular situation. In general TAU combination, however, neither TAU contains the other. In these cases we must examine the relationships or constraints among the concepts not shared between the two TAUs. For an example, consider

TAU-VANITY. This TAU represents a situation in which **?x** believes he has a special skill and is thus motivated to have a goal (of "showing off" in the Fox and Crow story) which will interfere with pre-existing goals.

```
    TAU name TAU-VANITY
1       binding-spec
2         [?belief (KNOW actor ?x
3                        obj (APPRAISAL obj (BODY-PART owner ?x)
4                                        value GOOD))]
5         [?standing-goal (GOAL actor ?x
6                               status FAILED)]
7         [?interfering-goal (GOAL actor ?x)]
8         [?mistake (ACT actor ?x)]
9       constraints
10        intention(?interfering-goal,?plan),
11        realization(?plan,?mistake),
12        enablement((APPRAISAL obj (BODY-PART actor ?x)
13                              value GOOD),
14                   ?mistake),
15        thwarting(?mistake,?standing-goal).
```

TAU-ULTERIOR shares a number of concepts with TAU-VANITY. The concepts which the two TAUs share in **The Fox and the Crow** are:

?standing-goal,
?mistake, and
?interfering-goal

The concepts which they do not share are **?mtrans** from TAU-ULTERIOR and the **?belief** in TAU-VANITY, lines 2-4[*].

The key to combining TAUs is finding the relationship between concepts which are not common to two TAUs which are being combined, here **?mtrans** and **?belief**. The TAU the we get by combining TAU-VANITY and TAU-ULTERIOR is TAU-SUCKERED. A detailed example of this process is given in [Dolan and Dyer, 1985]. This TAU embodies the planning failure of allowing someone else to take advantage of your dormant goals by providing one of the missing enablement conditions on that goal. In the case of **The Fox and the Crow** the dormant goal is the Crow's goal to show off. The missing enablement condition is a receptive audience. The Fox provides that audience and so tricks the Crow into defeating her standing goal of keeping the cheese. In this case **?mtrans** from TAU-ULTERIOR provides the enablement condition for **?belief** from TAU-VANITY.

TAU-SUCKERED is a new counterplanning technique which can be used by a planning program in situations where the appropriate constraints have been met. The represen-

---

[*] The representation for **?x**'s vain self-appraisal is simplified in this representation.

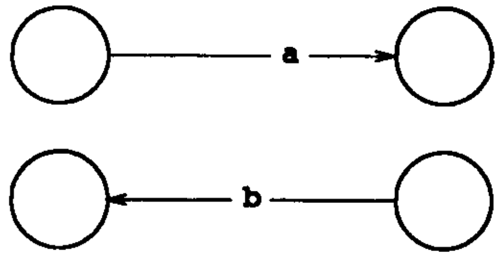
tation for TAU-SUCKERED is,

```
    TAU name TAU-SUCKERED
        binding-spec
          [?standing-goal (GOAL actor ?x
                                status FAILED)]
          [?interfering-goal (GOAL actor ?x)]
          [?act (ACT actor ?y)]
          [?mistake (ACT actor ?x)]
        constraints
          intention(?interfering-goal,?plan),
          realization(?plan,?mistake),
          thwarting(?mistake,?standing-goal),
          achievement(?mistake,?interfering-goal),
          enablement(?plan,?sub-goal),
          resulting(?act,?state),
          achievement(?state,?sub-goal),
          goal-conflict(?standing-goal,(GOAL actor ?y
                                            obj ?w))
          achievement(?mistake,(GOAL actor ?y
                                     obj ?w))
          not-obvious-result(?act,?mistake).
```

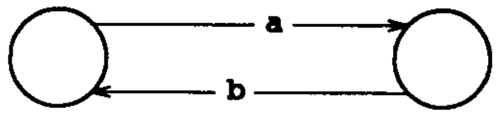
It is intuitively obvious to people that this TAU-SUCKERED represents the proper lesson to be learned by a reader who knows about vanity and physical enablements, but who has never seen this kind of trickery before.

In both cases of TAU learning we have examined the representation played a central role in guiding the learning. In specialization, the separation of the constraint from the concept patterns allowed us to form a new TAU by forming the conjuction of constraints. In the case of combination this separation of constraints from patterns showed use which concepts were not common to the two TAUs. Finding the relationship between theses two concepts allowed us to construct a new TAU, TAU-SUCKERED.

The problem of learning a new TAU from two old ones, either by specialization or by combination, can be thought of as a graph union problem. The elements of the **'binding-spec'** are node and the **'constraints'** are arcs. The union of the graphs represents the intersection of the situations which conform to the constraints. In matching abstract graphs however we have a problem. Say that we want to take the union of two graphs as below,



to



We have to make sure that the constraints **'a'** and **'b'** don't conflict with each other, making their union an inconsistent graph. That problem is solved here because we have a situation, found in the story, which has all the constaints among the components of the story. Therefore we know that the new structure is a possible combination. This is why learning from examples is important because the examples provide a model of what is actually possible.

5 Using TAUs in Planning

Using TAUs in planning requires two phases: (1) discovering a potential planning error, and (2) taking corrective action. The first phase is accomplished in the same manner as TAU recongition in the task of comprehending stories. As the planner examines the state of the world and formulates various plans, the concepts are "played back" as if they were occuring in a story. When a TAU trigger, such as line 6 of TAU-CONF-ENABL is encountered we have a potential planning error. Possible outcomes of the situation are examined to see if they meet the constraints of the TAU.

In these cases, where some outcome of the current situation might result in a goal failure, corrective action needs to be determined. In informal protocols taken from **The Fox and the Crow**, people seem to think that the Crow could have done two things: (1) put the cheese down, and (2) eaten the cheese. Along with the structure present in TAU-CONF-ENABL two simple heuristics yield these two solutions. The first is,

IF a new goal might cause a standing goal to fail
TRY removing a possible resource contention
between plans for the two goals

the standing goal is to maintain control over the cheese; The new goal is the goal to sing; the common resource is the Crow's mouth. Since there is no way to achieve the new goal without using the Crow's mouth, the system would choose to find another way to support the cheese.

The second solution, that of eating the cheese on the spot, is actually an instance of removing the need for the standing goal. In that case the guiding heuristic is,

IF a new goal might cause a standing goal to fail
TRY relieving the need for the standing goal

Here the system would see that the support for having the goal of controlling the cheese was to satisfy a hunger goal. If the hunger goal is satisfied, then the need to control the cheese goes away.

The function of TAUs like TAU-SUCKERED is to point out goals which need to be dropped. The goal conflict shown in lines 15-16 of TAU-SUCKERED points to the goal we need to examine. The heuristic,

IF a potential goal conflict is found,
LOOK for a a current plan which would thwart
  the conflicting goal and reconsider the need
  support for the current plan

directs the program to look at **'thwarting'** constraints (such as line 10 of TAU-SUCKERED) for the ACT which could defeat the goal. On the other hand, the heuristic,

IF a potential goal conflict is found,
LOOK for current plan which would achieve
  the conflicting goal for the other actor and
  reconsider the support for the current plan.

directs the program to look at **'achievement'** constraints (such as lines 17-18 of TAU-SUCKERED) for the ACT which could defeat the goal. In the case of TAU-SUCKERED, both these heuristics give the same result.

Our hypothesis is that the number of these heuristics is small, but that a program with a large number of TAUs could correct most of its planning errors using a few heuristics and the knowledge contained in TAUs.

6 Progress and Future Work

A program, CRAM, is under development as part of this research. Currently, CRAM is able to understand stories that are input as unconnected Conceptual Dependency [Schank 1972] structures. CRAM finds the planning errors in each story and characterizes them in terms of one or more TAUs. These TAUs are then used to index the story in memory for later retrieval. Based on an analysis of 17 Aesop's fables, we have encoded 12 TAUs by hand for CRAM. In addition to **The Fox and the Crow**, CRAM has been tested on 2 other stories: (1) another story on deception, **The Fox and the Bear**, and (2) a variation of **The Dog and his Shadow**.

The next step in the on-going development of the theory presented here is to give CRAM enough real-world knowledge so that it can understand all 17 stories used in our analysis; then remove several of TAUs from its knowledge-base and replace them with more primitive TAUs to have it learn the more complex ones.

In addition, CRAM will be able to give advice to correct the character's planning errors. Also planned for CRAM are a natural language parser and generator so that CRAM can take in stories as verbatim input and later generate English explanation of new TAUs it has discovered.

7 Conclusions

The approach presented here allows both specialization learning and chunking learning of planning errors in multiple planning agent domains. The structures learned can be used both for critiquing plans and also for generating counterplanning advice.

References

Bewick, T., Illustrator, *Treasury of Aesop's Fables*, Avenel Books, New York, 1973.

Carbonell, J. G., *Subjective Understanding: a Computer Model of Belief Systems*, Yale Ph.D Dissertation, Technical Report 150, 1979.

DeJong, G., "Acquiring Schema Through Understanding and Generalizing Plans", in *Proceedings of the Eighth International Joint Conference on Aritifical Intelligence*, 1983.

Dolan, C. P., *Memory Based Processing for Cross Contextual Reminding: Reminding and Analogy Using Thematic Structures*, UCLA Computer Science Department, M.S. Thesis, 1984.

Dolan, C. P. and Dyer, M. G., "Learning Planning Heuristics through Observation", Submitted for publication, 1985.

Dyer, M. G., *In-Depth Understanding: A Computer Model of Integrated Processing for Narrative Comprehension*, The MIT Press, Cambridge, Mass., 1983.

Dyer, M. G., "Understanding Stories through Morals and Remindings", in *Proceeding of the Eighth International Joint Conference on Artificial Intelligence*, 1983.

Dyer, M. G., "The Role of TAUs in Narrative Processing", in *The Proceeding of the Third Anual Meeting of the Cognitive Science Society*, 1981.

Franklin, B., *Poor Richard: The Almanacks for the Years 1733-1758*, Heritage Press, New York, 1964.

Kolodner, J. L., *Retrieval and Organization Strategies in Conceptual Memory: A Computer Model*, Technical Report 187, Yale University, Department of Computer Science, Ph.D. Disseration, 1980.

Kowalski, R., *Logic for Problem Solving*, North Holland, 1979.

Laird, J. E., Rosenbloom, P. S., and Newell, A., "Towards Chunking as a General Learning Mechanism", in *Proceeding of the National Conference on Artificial Intelligence*, 1984.

Lebowitz, M., *Generalization and Memory in an Integrated Understanding System*, Technical Report 186, Yale University, Department of Computer Science, Ph.D. Dissertation, 1980.

Meehan, J., *The Metanovel: Writing Stories by Computer*, Technical Report 74, Yale University, Computer Science Department, Ph.D. Dissertation, 1979.

Schank, R. C., "Conceptual Dependency: A Theory of Natural Language Understanding", *Cognitive Psychology*, Vol 3, No 4, 1972.

Schank, R. C. and Abelson, R. P., *Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures*, Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1977.

Schank, R. C., *Dynamic Memory: A Theory of Reminding and Learning in Computers and People*, Cambridge University Press, Cambridge, 1982.

Steele, G. L., *The Definition and Implementation of a Computer Language Based on Constraints*, MIT, Ph.D Dissertation, 1980.

Sussman G. J., *A Computational Model of Skill Aquisition*, AI Technical Report 297, AI Laboratory, MIT, Ph.D. Dissertation, 1973.

Wilensky, R. *Understanding Goal-Based Stories*, Technical Report 140, Yale University, Department of Computer Science, 1978, Ph.D. Dissertation.