Improving Explanatory Competence[1]

Kenneth S. Murray
murray@cs.utexas.edu
Department of Computer Sciences
University of Texas at Austin
Austin, Texas 78712

Abstract

Explanation plays an important role in acquiring knowledge, solving problems, and establishing the credibility of conclusions. One approach to gaining explanatory competence is to acquire proofs of the domain inference rules used during problem solving. Acquiring proofs enables a system to strengthen an imperfect theory by connecting unexplained rules to the underlying principles and tacit assumptions that justify their use. This paper formalizes the task of improving explanatory competence through acquiring proofs of domain inference rules and describes KI, a knowledge acquisition tool that discovers proofs of rules as it integrates new information into a knowledge base. KI's learning method includes techniques for controlling the search for proofs and evaluating multiple explanations of a proposition to determine when they can be transformed into proofs of domain inference rules.

1. Introduction

Knowledge-based systems must be capable of explaining their conclusions. One approach to gaining explanatory competence is to acquire proofs of the domain inference rules used during problem solving. Possessing proofs of rules used during problem solving has several advantages. First, a rule's proofs identify support for the rule, that is, the domain principles that justify the rule's correctness. Second, a rule's proofs explicate the tacit assumptions being made when the rule is used. By identifying the underlying principles and assumptions, proofs of inference rules enable the system to justify and qualify its conclusions to the user [SWAR83], guide knowledge refinement [SMIT85], and, in the case of default reasoning when assumptions are not met, improve problem solving [STAL77]. This paper formalizes the task of improving explanatory competence through acquiring proofs and describes a system that discovers proofs of domain inference rules as it integrates new information into a knowledge base.

Acquiring proofs strengthens an imperfect theory when new information enables proving previously unsupported rules. Initially, a knowledge base (or person) often includes tentative, default rules such as "birds can fly" or "leaves are green." However, as the knowledge base is extended, and competence in the domain improves, these default rules may be annotated with deeper causal support when explanations of the rules are discovered. Gagne [GAGN85] illustrates this behavior in people with the following example:

> A student is told *In vitro experiments show that Vitamin C increases the formation of white blood cells.* The student has prior knowledge that white blood cells destroy viruses, and intuitively knows that Vitamin C is taken to fight colds, which are caused by viruses. The student realizes that Vitamin C is capable of fighting colds *because* it stimulates the creation of white blood cells, which subsequently kill cold-causing viruses.

The student identified a causal explanation of an existing belief that was neither stated in the new information nor previously known. Having discovered this explanation the student possesses greater insight into why Vitamin C is taken to fight colds. For example, the student could now explain why Vitamin C is not taken in response to similar symptoms having causes unrelated to viruses (e.g., allergies).

[1] Support for this research is provided by the Air Force Human Resources Laboratory under RICIS grant ET. 14.

This paper describes research aimed at modeling this learning behavior. In Section 2, the learning task of proof acquisition is formulated, and a terminating, although incomplete, method of performing this task is developed in Section 3. Section 4 describes KI, a knowledge acquisition tool that implements this method to discover proofs of domain inference rules as it integrates new information into a knowledge base.

2. The Learning Task

Proofs of an inference rule identify sentences in a theory that ensure its truth. However, the contents of logical proofs are not restricted to sentences having explanatory (e.g., causal) significance. For example, a proof of a rule could include an arbitrary number of trivial tautologies (e.g., $p \Rightarrow p$) not relevant to the truth of the rule. Therefore, a distinction must be made between the set of all logical proofs and proofs acceptable as explanations. Let e be a predicate on proofs such that $e(p)$ is satisfied exactly when proof p is acceptable as an explanation. For example, e might restrict proof steps to be applications of modus ponens to a particular subset of rules. Let R be a domain theory and r_i be a sentence in R. The notation p_e denotes a proof that satisfies e, and $(R - r_i) \vdash_{pe} r_i$ denotes proof p_e is a derivation of r_i from the theory $(R - r_i)$.

Each proof of a rule identifies a set of underlying principles and assumptions that justify the rule's use; different proofs may elucidate different principles and assumptions. Therefore, the goal of acquiring proofs of rules for explanation includes identifying every p_e for each rule rather than any single p_e. This learning task may be characterized as the following information processing task:

Given: a rule set R
a predicate e on proofs

Find: for each $r_i \in R$
the proof set $P_i = \{p_e \mid (R - r_i) \vdash_{pe} r_i\}$

Unfortunately, when the language used to encode rules is as expressive as first order logic (FOL) this task is not solvable. However, this task becomes decidable when the following restrictions are adopted:

Let R_e be the subset of R including only rules that may be represented as horn clauses without functions.

1) U_e, the universe of discourse for R_e, is a finite subset of U, the universe of discourse for R.

2) e admits only proofs containing non-cyclic applications of rules in R_e.

These restrictions enable the existence of a terminating method by sacrificing completeness. Intuitively, decidability is achieved by restricting the universe of discourse to a finite set. A theory equivalent to R_e can be constructed by replacing each $r_i \in R_e$ with the set of ground implications that includes every possible binding of the variables in r_i to elements of U_e. The finiteness of U_e makes such a construction possible. The result is a propositional theory and is therefore decidable.

The restricted learning task may be characterized as:

Given: a rule set R
a predicate e on proofs

Find: for each $r_i \in R$
the proof set $P_i = \{p_e \mid (R_e - r_i) \vdash_{pe} r_i\}$

While this task is guaranteed to have a terminating solution, the restrictions cannot guarantee tractability. In practice, solutions cannot be expected to discover all proofs enabled by the restricted theory. Therefore, solutions must include some mechanism to bias their search for proofs. The next section describes a method of guiding search for restricted proof acquisition.

3. Discovering Proofs through Hypothetical Reasoning

One approach to guiding deduction involves separating inference from the process of instantiating quantified formulae [McAL80]. Inference is then limited to computing the entailment of a small but "representative" set of ground propositions. This sections describes a ***generate and test*** search procedure that manipulates a set of ground propositions and implications to guide search for proofs of rules in R. This search is summarized by the following cycle:

1) generate a hypothetical context: a set of propositions over hypothetical instances of some small subset variables referenced by rules in R

2) generate all ground explanations (i.e., sequences of deductions) enabled by repeatedly applying rules in R_e to the propositions

3) determine if any resulting explanation can be generalized into a proof of some rule in R

4) extend the context with propositions over new hypothetical instances of variables referenced by rules in R; goto step 2.

To initiate the search, select a very restricted set of rules called *Training*; the search will be biased towards discovering proofs that make use of these rules.[2] Let *Context* be some small set of propositions that satisfy every sentence in *Training*. For example, if *Training* = {[*isa*(*x Person*) & *location*(*x Austin*) ⇒ *location*(*x Texas*)]}, then the proposition set {*isa*($Person_1$ *Person*), *location*($Person_1$ *Austin*), *location*($Person_1$ *Texas*)} satisfies *Training*.

Second, generate ground explanations enabled by the propositions in *Context* and the rules in R_e. This involves computing all possible deductions by repeatedly applying rules in R_e to the propositions in *Context* (i.e., by exhaustive forward-chaining). While termination is guaranteed, exhaustive forward-chaining has the potential for exponential combinatorics. However, the rules are chaining on a very restricted set of instances, so, in practice, restrictions on proof construction will typically be sufficient to prevent intractable chaining.[3]

Finally, determine if any resulting explanation can be generalized into a proof of some rule. This is accomplished using techniques developed for explanation-based learning to generalize and compile explanations (e.g., [MOON88]). To continue the search for proofs, extend *Context* with additional hypothetical instances and add some set of propositions on these new instances. Then repeat the generation and evaluation of explanations as before.

Under this strategy, the search for proofs is controlled by the extensions made to *Context*. One way to generate hypothetical instances is to instantiate Skolem functions appearing in the rule set R. For example if *isa*(*Fred Person*) is a proposition in *Context*, and [*isa*(*x Person*) ⇒ *mother*(*x* $fn_1(x)$)] is a rule in R, $Person_1$ could be a new hypothetical instance and *mother*(*Fred* $Person_1$) a new proposition for *Context*. Now R_e can be extended with the ground implication [*isa*(*Fred Person*) ⇒ *mother*(*Fred* $Person_1$)]. This enables limited representation in R_e of rules from R that involve predicates on functions (e.g., Skolem functions). The next section describes a program that implements this method and illustrates it with an example.

---

[2] To model learning by discovery, *Training* can be any subset of R. Alternatively, when existing knowledge is being extended with new information, it is natural to prefer discovering explanations enabled by the new information. In this context, *Training* is the set of axioms being added to R.

[3] To guarantee tractability, additional restrictions can be imposed on proof construction, such as a bound on the execution time allotted to compute explanations.

Figure 1: New information and the initial context

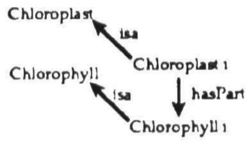
1a) New Information Provided to KI

$[\forall$ x isa(x Chloroplast)

$\Rightarrow$

$\exists$ y isa(y Chlorophyll) & hasPart(x y)]

1b) Propositions satisfying the new information



4. Acquiring Proofs During Knowledge Integration

KI is an interactive knowledge-acquisition tool being developed to help knowledge engineers integrate new information into the Botany Knowledge Base [MURR88, PORT88]. This knowledge base currently contains over four thousand frames representing plant anatomy, physiology, and development; it has been constructed in collaboration with MCC's CYC project [LENA89].

This section describes an implemented example of KI extending the Botany Knowledge Base with new information relating chloroplasts and chlorophyll. The knowledge base already contains extensive partonomic knowledge of plants and some knowledge of photosynthetic pigments, such as chlorophyll. A knowledge engineer wishes to extend the knowledge base to represent the fact that chlorophyll is a constituent part of chloroplasts (see Figure 1a). The task of KI is, in general, to identify interesting consequences of this new information and, in particular, to identify how this new information can explain existing beliefs. KI's model of knowledge integration comprises three prominent activities:

1) Recognition: identifying the knowledge relevant to new information
2) Elaboration: applying the expectations provided by relevant knowledge to determine the consequences of the new information
3) Adaptation: modifying the knowledge base to accommodate the elaborated information

4.1 Recognition

During recognition KI identifies concepts in the knowledge base that are relevant to the new information. This involves maintaining a *learning context* – a set of propositions about hypothetical instances of concepts deemed relevant to the new information. When presented with new information, KI initializes the context with propositions that satisfy the new rules (e.g., Figure 1b). To extend the learning context, KI uses *views* to determine which concepts in the knowledge base, beyond those explicitly referenced in the context, are relevant.

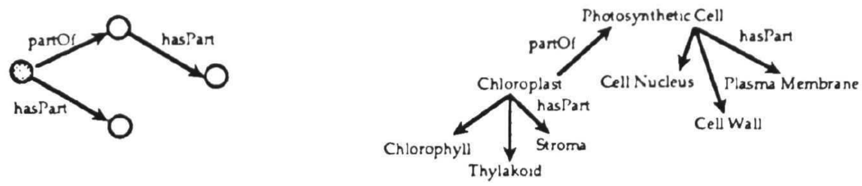
Views are sets of propositions that interact in some significant way and should therefore be considered together. Views are created by applying a generic *view type* to a domain concept. Each view type is a parameterized semantic net, represented as a set of paths emanating from a root node and used during knowledge integration as a reminding schema. Applying a view type to a concept involves binding the concept to the root node and instantiating each path. Figures 2a and b present an example view type and the view created by applying it to chloroplast.

To extend the learning context, KI identifies the views defined for concepts already contained in the learning context. Each candidate view is scored with a heuristic measure of relevance: the percentage of concepts contained in the view that are also contained in the learning context. KI presents the list of candidate views, ordered by their relevance score, to the knowledge engineer, who selects one for use.[4] The set of propositions contained in the selected view are added to the learning context. This results in a learning context comprising those concepts in the knowledge base considered most relevant to the new information.

[4] Alternatively, an autonomous version of KI selects the view having the highest relevance score.

Figure 2: An example view and view type

2a) Qua Component **2b) Chloroplast Qua Component**



2c) [∀ s isa(s Chloroplast) ⇒ ∃ tuvwxyz isa(t Chlorophyll) & hasPart(s t) & isa(u Thylakoid) & hasPart(s u) & isa(v Stroma) & hasPart(s v) & isa(w Photosynthetic-cell) & partOf(s w) & isa(x Cell-nucleus) & hasPart(w x) & isa(y Cell-wall) & hasPart(w y) & isa(z Plasma-membrane) & hasPart(w z)]

View type *Qua Component* identifies two paths emanating from a concept relevant to its role as a part of a physical structure (the shaded node designates the root concept). Applying this view type to chloroplast identifies the segment of the knowledge base representing chloroplast as a part of a photosynthetic cell. The path variables may have multiple bindings (e.g., the chloroplast parts include chlorophyll, stroma, and thylakoid). 2c) expresses in FOL the inference this view supports.

In addition to adding propositions contained in the selected view to the learning context, KI adds the implications that characterize the conditions under which these propositions are assumed to be true. Whenever the preconditions of a view are satisfied, the propositions contained in the view are assumed to hold. For example, when the proposition *partOf(chloroplast$_1$ photosynthetic-cell$_1$)* is added, the following implications are also added:

1) *isa(chloroplast$_1$ chloroplast)* ⇒ *partOf(chloroplast$_1$ photosynthetic-cell$_1$)*
2) *isa(photosynthetic-cell$_1$ photosynthetic-cell)* ⇒ *partOf(chloroplast$_1$ photosynthetic-cell$_1$)*

The first implication follows from view *ChloroplastQuaComponent* (see Figure 2c); the second follows from *PhotosyntheticCellQuaStructure* (see Rule 6 of Figure 6). Since there is high overlap among views, many such implications are added to the learning context. This enables limited representation in R_e of rules from R that involve predicates on functions (e.g., Skolem functions). The use of these implications is often essential for completing proofs of domain inference rules.

4.2 Elaboration

During elaboration KI determines how the new information interacts with the existing knowledge within the learning context. Rules in R_e are allowed to exhaustively forward-chain, propagating the consequences of the training throughout the context. For example, one consequence of chloroplasts having chlorophyll is that their color is green. Some of the domain inference rules applicable to this example are listed in Figure 3a, and the resulting conclusions are presented in Figure 3b.

KI enters a cycle of recognition (i.e., selecting views) and elaboration (i.e., applying inference rules) that explicates the consequences of the training while searching for new proofs of rules. This cycle continues until the user intervenes or the relevance scores of all candidate views fall below a progressive threshold. Figure 4 illustrates the second round of this cycle. The recognition phase extends the context of Figure 3b with the set of propositions relevant to a photosynthetic cell in its role as a producer during cell photosynthesis. The elaboration phase propagates the consequences of the new information throughout the extended context.

4.3 Adaptation

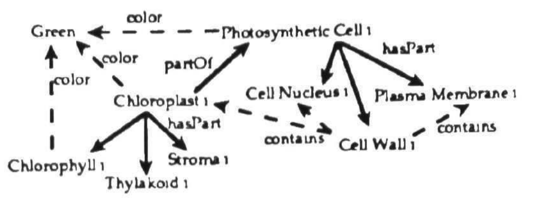
During adaptation KI determines if elaboration has revealed any new proofs of inference rules. An interesting prerequisite of discovering proofs of rules is that multiple ground explanations for some proposition must exist. When this occurs, KI determines whether any explanation can be generalized into a proof of some inference rule.

Figure 3: Example rules and inferences

3a) Example Inference Rules

1. Rule: @partOf-Preserves-Color
   Color may be inferred from the color of the parts
   [∀ xyz partOf(x y) & color(x z) ⇒ color(y z)]
2. Rule: @Chlorophyll_color
   Inheritance rule: Chlorophyll is inherently green
   [∀ x isa(x Chlorophyll) ⇒ color(x Green)]
3. Rule: @Leaf_color
   Inheritance rule: Leaves are assumed to be green
   [∀ x isa(x Leaf) ⇒ color(x Green)]

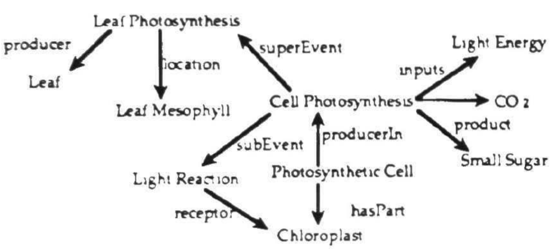
3b) The Elaborated Context



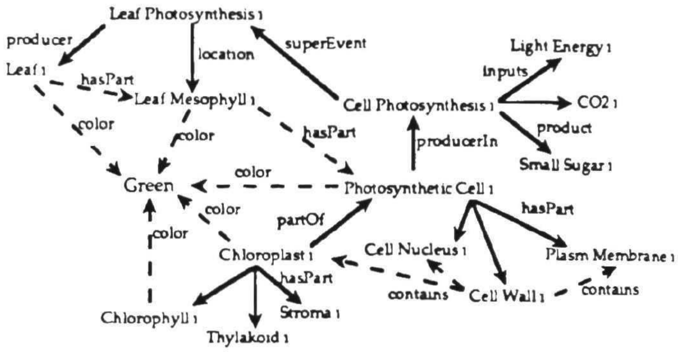
The dashed arrows indicate propositions inferred during elaboration; subscripts denote category instances (e.g., $isa(Chloroplast_1\ Chloroplast)$).

Figure 4: Recognition and elaboration during cycle 2

4a) Photosynthetic Cell Qua Producer



4b) The Elaborated Context



Let E be the set of explanations of some proposition, and let r_i be the last rule applied in some explanation $e_i \in E$. KI evaluates each alternative explanation in E to determine if it can be transformed into a proof of r_i. First KI uses explanation-based generalization to compute the maximal generalization of each $e_j \in E\text{-}e_i$. Let ge_j be the generalization of explanation e_j. Then KI compares the consequence of ge_j to the consequence of r_i. When the consequence of ge_j is equivalent to (or subsumes) the consequence of r_i, KI searches ge_j for sub-explanations whose weakest preconditions entail the consequence of ge_j and are equivalent to (or subsume) the preconditions of r_i; each such sub-explanation constitutes a proof of r_i.

In the example, KI's elaboration of the chloroplast training produces several explanations of the proposition $color(Leaf_1\ Green)$, two of which are presented in Figure 5. Explanation e_1 involves a single application of rule r_3 whose precondition is $isa(x\ Leaf)$. Since this is identical to the preconditions of the generalization of explanation e_2, the generalization of e_2 is a new proof of *@Leaf_color*. Note the importance of implications that explain propositions arising from views. As shown in Figure 6, the views *LeafQuaStructure*, *MesophyllQuaStructure*, *PhotosyntheticCellQuaStructure* and *ChloroplastQuaStructure* all chain together to demonstrate that leaves contain chlorophyll.

This example shows how KI discovers a proof for the existing rule ***leaves are green*** while integrating the new information ***chloroplasts have chlorophyll.*** This proof improves the system's explanatory competence by revealing the tacit assumptions (e.g., ***mesophyll contains photosynthetic cells***) and domain principles (e.g., ***an object's color is determined by the color of it's parts***) that justify the rule's use. For example, the proof provides an answer the query ***why are leaves green?***

Figure 5: Two explanations of $color(leaf_1\ Green)$

explanation e_1

- color(Leaf$_1$ Green)
- $\Leftarrow_{r3}$ isa(Leaf$_1$ Leaf)

explanation e_2

- color(Leaf$_1$ Green)
- $\Leftarrow_{r1}$ partOf(Mesophyll$_1$ Leaf$_1$)
  - $\Leftarrow_{r4}$ isa(Leaf$_1$ Leaf)
  - color(Mesophyll$_1$ Green)
  - $\Leftarrow_{r1}$ partOf(Photosynthetic-cell$_1$ Mesophyll$_1$)
    - $\Leftarrow_{r5}$ isa(Mesophyll$_1$ Mesophyll)
      - $\Leftarrow_{r4}$ isa(Leaf$_1$ Leaf)
    - color(Photosynthetic-cell$_1$ Green)
    - $\Leftarrow_{r1}$ partOf(Chloroplast$_1$ Photosynthetic-cell$_1$)
      - $\Leftarrow_{r6}$ isa(Photosynthetic-cell$_1$ Photosynthetic-cell)
        - $\Leftarrow_{r5}$ isa(Mesophyll$_1$ Mesophyll)
          - $\Leftarrow_{r4}$ isa(Leaf$_1$ Leaf)
      - color(Chloroplast$_1$ Green)
      - $\Leftarrow_{r1}$ partOf(Chlorophyll$_1$ Chloroplast$_1$)
        - $\Leftarrow_{r7}$ isa(Chloroplast$_1$ Chloroplast)
          - $\Leftarrow_{r6}$ isa(Photosynthetic-cell$_1$ Photosynthetic-cell)
            - $\Leftarrow_{r5}$ isa(Mesophyll$_1$ Mesophyll)
              - $\Leftarrow_{r4}$ isa(Leaf$_1$ Leaf)
        - color(Chlorophyll$_1$ Green)
        - $\Leftarrow_{r2}$ isa(Chlorophyll$_1$ Chlorophyll)
          - $\Leftarrow_{r7}$ isa(Chloroplast$_1$ Chloroplast)
            - $\Leftarrow_{r6}$ isa(Photosynthetic-cell$_1$ Photosynthetic-cell)
              - $\Leftarrow_{r5}$ isa(Mesophyll$_1$ Mesophyll)
                - $\Leftarrow_{r4}$ isa(Leaf$_1$ Leaf)

The notation $p \Leftarrow_i q$ denotes p is infered from q by rule i (see Figures 3 and 6).

Figure 6: Inferences enabled by views and required for the proof

Rule 4) *LeafQuaStructure*
[∀ w isa(w Leaf) ⇒ (∃ xyz isa(x Mesophyll) & partOf(x w) & isa(y Epidermis) & partOf(y w) & isa(z Vascular-network) & partOf(z w))]

Rule 5) *MesophyllQuaStructure*
[∀ x isa(x Mesophyll) ⇒ (∃ y isa(x Photosynthetic-cell) & partOf(y x))]

Rule 6) *PhotosyntheticCellQuaStructure*
[∀ v isa(v Photosynthetic-cell) ⇒ (∃ wxyz isa(w Cell-nucleus) & partOf(w v) & isa(x Cell-wall) & partOf(x v) & isa(y Chloroplast) & partOf(y v) & isa(z Plasma-membrane) & partOf(z v))]

Rule 7) *ChloroplastQuaStructure*
[∀ w isa(w Chloroplast) ⇒ (∃ xyz isa(x Chlorophyll) & partOf(x w) & isa(y Thylakoid) & partOf(y w) & isa(z Stroma) & partOf(z w))]

(e.g., *leaves are green because they contain chlorophyll*). Alternatively, the proof guides dependency-directed backchaining to identify assumptions that explain why a particular leaf is not green (e.g., *the leaf's mesophyll has no photosynthetic cells*).

4.4 Strengths, Limitations, and Future Work

KI's approach to knowledge integration involves creating a *hypothetical model* comprising concepts relevant to the new information, and then using the model to derive the consequences of the new information for concepts represented in the model. Reasoning with a single, propositional model (e.g., a model of a hypothetical leaf), rather than reasoning about entire classes of objects (e.g., models of all possible leaves) provides greater focus and tractability. However, this prevents KI from discovering many proofs that alternative models would reveal. Furthermore, KI is currently not capable of exploring all the alternative, and often mutually-inconsistent, behaviors of a model that frequently arise during qualitative simulations [KUIP87]. This prevents KI from discovering many proofs that a single model may be capable of revealing under varying assumptions. Future work should develop methods for guiding the exploration of alternative models and the *possible worlds* for a single model.

The inferences completed with the model are not explicitly selected: rules exhaustively forward-chain. This type of reasoning corresponds to what Johnson-Laird calls *implicit inference* – the automatic, seemingly effortless inferences humans make during mundane tasks, such as discourse comprehension [JOHN83]. The complement of implicit inference is *explicit inference* – the intentional and conscious reasoning humans perform during problem solving. Currently, KI is not capable of demonstrating this kind of goal-directed elaboration. Future research must address developing methods for interleaving these two types of inference.

5. Summary

Explanation plays an important role in a system's ability to acquire knowledge, solve problems, and establish the credibility of its conclusions. One approach to gaining explanatory competence is acquiring proofs of the inference rules used during problem solving. Acquiring proofs enables a system to strengthen an imperfect theory as previously unexplained rules are connected to the underlying principles and tacit assumptions that justify their use.

KI is a knowledge acquisition tool that strengthens an existing domain theory by discovering new proofs of inference rules. When new information is provided, KI actively searches for proofs of existing beliefs that are enabled by the new information. This requires methods for restricting both the universe of discourse and the use of inference rules that include predicates on functions.

KI exploits a type of domain knowledge called *views* to precisely manage a context comprising ground propositions used during the search for proofs. Views are knowledge-base segments composed of interrelated propositions that should be considered collectively. Each view embodies the use of functions to create entities over which propositions are asserted. Separating the use of functions to create entities from the problem of proving theorems enables KI to guide its search for proofs of domain inference rules.

References

[GAGN85] Gagne, E.D. *The Cognitive Psychology of School Learning*, Boston: Little, Brown and Company, 1985.

[JOHN83] Johnson-Laird, P.N. *Mental Models*, Harvard University Press, 1983.

[KUIP87] Kuipers, B.J., and Chiu, C. Taming Intractable Branching in Qualitative Simulation. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence* (1987).

[LENA89] Lenat, D.B., and Guha, R.V. *Building Large Knowledge-Based Systems: Representation and Inference in the CYC Project*, Addison-Wesley, 1989.

[MCAL80] McAllester, D.A. An Outlook on Truth Maintenance. AI Memo No. 551, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1980.

[MOON88] Mooney, R.J. *A General Explanation-Based Learning Mechanism and Its Application to Narritive Understanding*, PhD Dissertation, Computer Science Department, University of Illinois at Urbana-Champaign, 1988.

[MURR88] Murray, K.S. KI: An Experiment in Automating Knowledge Integration. Technical Report AI-88-90, Artificial Intelligence Laboratory, Department of Computer Sciences, University of Texas at Austin, 1988.

[SMIT85] Smith, R.G., Winston, H.A., Mitchell, T.M., Buchanan, B.G. Representation and Use of Explicit Justification for Knowledge Base Refinement. *Proceedings of the Ninth International Joint Conference on Artificial Intelligence* (1985).

[STAL77] Stallman, R.M., and Sussman, G.J. Forward Reasoning and Dependency Directed Backtracking in a System for Computer-Aided Circuit Analysis. *Artificial Intelligence 9* (1977).

[SWAR83] Swartout, W.R. XPLAIN: A System for Creating and Explaining Expert Consulting Programs. *Artificial Intelligence 21* (1983).

[PORT88] Porter, B., Lester, J., Murray, K., Pittman, K., Souther, A., Acker, L., and Jones, T. AI Research in the Context of a Multifunctional Knowledge Base: The Botany Knowledge Base Project. Technical Report AI-88-88, Department of Computer Sciences, University of Texas at Austin, 1988.