

NETWORK TRUTH-MAINTENANCE FOR DEDUCTION AND MODELLING **

Alan M. Thompson
Information Systems Division
NASA Jet Propulsion Laboratory
4800 Oak Grove Drive
Pasadena, California 91103

Logical dependency based "Truth Maintenance" is applied to network representations of symbolic logic relations, resulting in reduced structural manipulation. Logical support is determined dynamically within existing structure by procedural specialists attached to the truth functional connectives which attempt to maintain consistency in the network. The use of context layers in addition to dependencies allows more accurate world-modelling than previous dependency-based modelling schemes. Focus of attention in deductive search and planning is accomplished by the propagation of control information through the network in addition to normal truth values.

1. The Development of Truth-Maintenance

1.1 Introduction

The problem of modelling the consequences of an action has been central to the development of automated symbolic problem solving. Selection of proper strategies for accomplishing goals may be done only when the machine has an accurate model of the state of the world at each decision point (or else has a model of its own uncertainty). Early systems required a programmer to associate with each action description a complete representation of the changes that should be made to the world model when the execution of the action was simulated. In highly coupled problem domains where single actions may alter large numbers of states,* this requirement can lead to a difficult programming problem. If an unrestricted hierarchy of definitions is allowed, the consequences of actions may become highly context dependent, and the problem of simply enumerating the affected states can easily become intractable.

In integrated problem-solving systems, in which planning and world-modelling are combined with automatic inference from logical relations, it becomes highly desirable for the modelling component to employ the same logical relationships used by the inference process. Thus, if some planned action changes a state that had been inferred from previous states, then its supporting antecedents must also be changed ("re-futed support"). Similarly, if the changed state had previously been used as the justifi-

cation for another state, then the dependent state may no longer be justifiably believed ("lost support"). Also, the new state may provide a justification for logically dependent states ("discovered support"). These concepts led to the notion of logical dependency as a means of maintaining the justifications of derived states and of a process of "Truth Maintenance" (TM) as a means of determining the state of belief or disbelief in expressions as a consequence of changes to the database.

1.2 Previous related work

The most complete recent systems employing TM techniques are those of Doyle [2] and London [5]. Although these systems are based on similar theoretical grounds, they differ widely in implementation and application. Doyle's system uses dependencies to determine the consequences of assumptions, to identify inconsistent sets of assumptions and to control backtracking. London's system is designed for modelling the consequences of proposed action sequences. Both of these systems allow knowledge to be expressed in localized "chunks" with the coupling to other chunks accomplished by the dependency network. Both systems depend upon a separate inference module to compute the initial justifications for states, which are then used by the TM process. In both cases, the dependencies must be explicitly declared by the inference process in clause form, i.e., as (the conjunction of) a list of the antecedents of the derived state.

A thorough background discussion on the use of logical dependencies may be found in the referenced works; this discussion is restricted to certain generalizations of the earlier methods and with the solutions to certain difficulties encountered in the use of dependencies for world-modelling. Only examples in propositional logic will be illustrated; the methods have been extended to first-order, but are not implemented as of this writing. (This is an abridged version of the original paper, which can be obtained from the author.)

1.3 Motivation and Features of the JPL System

The "ERIS"⁺ system being developed at JPL is designed to generalize and improve the TM and deductive processes in several respects.

⁺ ERIS was the Greek goddess of discord.

*The following definitions apply to this discussion: A statement is represented as an expression (normally a Predicate Calculus formula). Expressions are stored in an indexed (associative) database as nodes. A state is the truth state (represented by a truth value (TV), such as true, false, etc.) of a node (or its associated statement). axioms or rules are statements of known truth value whose state may not be changed. Statement connectives include the normal truth-functional connectives (AND, OR, etc.) plus connectives for representing causality.

**This paper presents the results of one phase of research carried out at the Jet Propulsion Laboratory, California Institute of Technology, under Contract No. NAS 7-100, sponsored by the National Aeronautics and Space Administration.

The propagation of logical support, one of the primary functions of truth maintenance (discovered support), is also required by deduction; as deduction pursues potential proofs, the discovery of an actual proof should result in marking the valid line of support. In addition, by removing the restriction that dependencies be explicitly declared in clause form, considerable structural manipulation could be avoided during TM processing. In ERIS, generalized TM procedures have been developed that are combined with the deductive process, resulting in a significant reduction in symbol/structure manipulation during deduction.

Rule-based systems often have the restriction that knowledge be expressed in clause form, with inference (and therefore logical support) directed only from the antecedent to the consequent of implicative rules. Since this syntax restriction can often prove burdensome, especially to non-programmers, a more general representation was desired, in which domain axioms could be expressed using any combination of relational connectives. The SRI "SNIFFER" system employed such a representation, but required "extraction" rules to place the term to be proved in the consequent of an implication [3]. In ERIS, by defining TM behavior for each statement connective the need for applying rewrite rules to reformulate domain axioms as implications during deduction has been eliminated by a graph labelling procedure that determines logical support. Also, there is no need to conduct a separate search for a proof of a statement's negation, since the TM process may result in the discovery of a line of support for either result.

Previous work has demonstrated the advantages of dependency methods over linear (chronological) context layer schemes for the control of backtracking (see [4], [5], [2]), but context switching for purposes of hypothetical reasoning or multi-world modelling requires considerable computation using dependencies alone, as compared to typical "context-list" schemes. Although ERIS uses dependencies to control backtracking (by identifying the particular assumptions responsible for conflicts), the use of context layers is available for these other activities. Furthermore, by allowing the definition of a particular sub-context for modelling purposes, the combined dependency/context scheme permits the same TM procedures to be used for both deduction and modelling, even though different behavior results. This method has resulted in greater accuracy in world-modelling than London's dependency-based technique, upon which it is based.

When the truth value of a statement is not known, the deductive process may have to examine all applicable axioms for a derivation. Focus of attention during the search is the key factor in determining the effectiveness of a deduction method when the number of domain axioms is large. In a manner analogous to the propagation of logical support for known truth values, focus of attention during deduction is propagated through the network of expressions using "meta" truth values that contain control information, such as the type of search (deduction, planning, etc.), priority, strategy advice, etc. The meta-TV's allow efficient, uniform treatment of all the propagation rules associated with statement connectives, and, by initial propagation through the domain axioms to possible antecedents of relevant nodes, they

provide a way of delimiting the subdomain within which the forward (antecedent) reasoning of TM behavior is allowed. This restriction of forward inference is essential in unbounded domains to suppress the generation of irrelevant assertions. Moreover, when domain-dependent deductive strategy information is available (as a supplement to generalized syntactical strategy rules), the control information in the database can be explicitly accessed by modal operators that may be used in strategy advice expressions. Similar use of modal operators for control information was available in the MIT AMORD/TMS system [2, 1].

2. The Method of Network Truth Maintenance

2.1 The Definition of Statement Connectives

The truth-functional connectives of symbolic logic are defined by associating with the name of the connective an encoded set of rules that govern the behavior of expressions formed with the connective. The rules determine the ways in which the expression can provide logical support to the subexpressions it connects (and vice versa) and are responsible for propagating the consequences of changes in support to the affected expressions. For efficiency, the rules are embedded in a procedural specialist for the connective. When a new TV is asserted for an argument of a connective expression or an entire expression, the specialist examines the TV's of the expression and its arguments. Knowledge of the truth table for a connective is embedded in its specialist and is used to find a consistent labelling of the expression and its arguments. If the new TV was assigned to the expression, then the TV's of the arguments may have to be changed; if the new TV was assigned to one of the arguments, then the TV of the expression may change unless the expression is acting as an axiom, in which case the TV of some other argument may have to change. The specialists are also responsible for propagating control information during deduction or planning. Connectives used to represent knowledge of causality are defined to the system similarly, by providing an appropriate procedural specialist.

2.2 The Notion of a "Consistency Context"

Any connective expression may behave as an axiom by having "current" support for its truth value. Truth values are context sensitive; an expression may have one truth value in a given context and a different one in another context or in an earlier "layer" of the given context. A special subcontext, called the "Consistency Context" (CC), defines the subcontext within which logical support for one truth value may not be refuted by a contradictory line of support. Normal TM guarantees that TV's of true or false are well-founded, that is, that they do not depend upon a circular justification [2]. During deduction the CC is identical with the entire context, well-founded TV's may not change, and contradictions result in retracting the offending assumptions (as in [5]). During modelling, however, the CC contains only the current set of axioms and those changes made SINCE the (modelled) execution of the most recent action. Lines of support within the CC may override other support, permitting the network to change so that it reflects the world state after the execution of the action. Truth values in the global plan

context are assumed not to change unless so forced by new support. Axioms behave as such by the fact that their TV is in the CC, so that even subexpressions may behave as axioms if their TV has well-founded support within the CC. For example, given the axiom (IMPLIES A (OR B C)), support for the truth of A will cause the disjunction to behave as an axiom, i.e., if B is known to be false, then C will be labelled true.

2.3 Operation During World-Modelling

There are several problems associated with the use of dependencies for world-modelling, relating to the requirement for refuting all logical support for a changed state. These problems are: local uncertainty, global uncertainty, and the need for adequate domain knowledge. Local uncertainty arises in cases where support derives from multiple antecedents, so there is more than one possible labelling that refutes the support should the consequent be changed by an action. Suppose that C was originally proved true by the truth of A, B, and the axiom (IMPLIES (AND A B) C). If C is made false by a planned action, it is necessary to refute the conjunction. Since both A and B were true (before the action), it is indeterminate whether one or both should be changed. London outlines two approaches to the problem of local indeterminacy: the introduction of uncertainty into the network by allowing the propagation of a TV for "unknown" to refute the support for both A and B, or else by requiring domain knowledge to be sufficiently complete and, moreover, suitably expressed to determine with certainty all consequences of a state change (by forward (antecedent) inference). These alternatives are generally unacceptable, but the method discussed here offers less sensitivity to the structure of domain axioms and avoids the introduction of uncertainty in most cases.

London pointed out that, in the case of refuting conjunctive support, the spread of uncertainty can in some cases be mitigated by new lines of support emanating from the state originally changed, as would occur in the case of "exhaustive support" with complete domain knowledge [4]. Since with incomplete knowledge such new support is not guaranteed to propagate to the refuted conjunction, some uncertainty may still remain in the net. By using a multiple-level agenda mechanism, the ERIS system defers the propagation of uncertainty to a queue with lower priority than that for known truth values, so that a line of support may be found that eliminates the uncertainty without having to mark nodes as "unknown" unnecessarily.

Furthermore, unlike London's system the CC method provides a way to undo the labelling of nodes marked "unknown" in the event that uncertainty propagates to a well-founded consequence of the original state change or if uncertainty was spread to nodes in a conjunctive antecedent and subsequently one of the nodes was found to have a known refutation. This is accomplished in part by allowing well-founded TV's to override the "unknown" TV even within the CC and also by retracting the spread of uncertainty when needed to reconstruct previous support. The CC method has the additional ability to detect instances where multiple antecedents are not linked by the same connective, but in which the network is globally indeterminate following a change.

The remaining difficulty encountered in modelling is the problem of capturing all the consequences of a state change. The above discussion pertains to instantiated structure with initially known truth values, but it is often necessary to invoke deduction to determine whether nodes that could possibly have been providing support before the state change (due to unresolved uncertainty) were in fact providing support. If such nodes were supporting a state that is changed by an action, then their support must also be refuted. Failure to determine the support status of indeterminate nodes with respect to a changed state may cause some of the consequences of an action to be ignored. This is the problem of exhaustive support, as described by London [4], and is a major problem in dependency-based modelling. It is related both to the completeness of the domain knowledge and to deductive search, since the basic problem is that of proving a universal, namely the completeness of the set of consequences (or the non-existence of a line of support from the action consequent to any supposedly unaffected state). The problem of the completeness of domain knowledge is outside the scope of the system (although indeterminacy following a state change may allow it to detect incompleteness). Deducing the TV's of previously unknown states to determine their support status may place large demands on the deductive component.

3. Concluding Remarks

By allowing logical support to be determined within existing structure, the restriction that dependencies be explicitly declared in clause form is eliminated. This is accomplished by connective specialists that use graph labelling to maintain consistency in the network, greatly reducing the overhead for symbol manipulation required by the deductive process, as rewrite rules are also eliminated (except for the initial simplification of expressions). The creation of new structure is limited to the instantiation of rules during deduction. By avoiding the structural manipulation of expressions, the network becomes, in effect, a "compiled" knowledge representation, upon which the procedural specialists operate merely by the propagation of normal and "meta" truth value "signals".

REFERENCES

- [1] J. DeKleer, et al., "Explicit Control of Reasoning", MIT AI Lab Memo 427, June 1977.
- [2] Jon Doyle, "Truth Maintenance Systems for Problem Solving", A.I. Laboratory, MIT. AI-TR-419.
- [3] Richard Fikes and Gary Hendrix, "A Network-Based Knowledge Representation and its Natural Deduction System", Proc. IJCAI-77, MIT, Cambridge, Mass., August 1977.
- [4] Philip E. London, "Dependency Networks as a Representation for Modelling in General Problem Solvers", Dept. of Computer Science, U. Maryland. TR-698.
- [5] Richard Stallman and Gerald Sussman, "Forward Reasoning and Dependency-Directed Backtracking in a System for Computer-Aided Circuit Analysis", MIT AI Memo 380, also in *Artif. Intell.*, 9:2 (October 1977).