AUTOMATED TEMPORAL EXPLANATION WITH THE MODAL LOGIC Z

David E. Leasure
Texas A&M University-Corpus Christi
leasure@tamucc.edu

Abstract

The method of temporal explanation by minimization of miracles is presented in the modal logic Z. Because we are able to represent problems propositionally, and because propositional Z is decidable, temporal explanation in Z has been automated. Four examples are presented.

INTRODUCTION

Temporal explanation is the ability to reason both backwards and forwards in time. A typical temporal explanation problem is, "given a partial description of the problem, fill in the missing actions, preconditions, and consequences of actions." In the extreme, when only the initial and final states are provided, temporal explanation is equivalent to goal oriented planning. Temporal explanation is more general than planning because in addition to choosing actions to perform, it also reasons about existing plans and supplies missing facts that explain the observed consequences of executing a given plan, even when those consequences depart from the expected outcome of executing an action sequence.

Lifschitz and Rabinov, in their 1989 AI Journal paper, present a formal theory of temporal explanation based on circumscription applied to the situational calculus. We present a reformulation of their approach in the modal logic Z that offers the following improvements. First, the expressive power is enhanced to allow reasoning over not only direct consequences of actions, but also the ramifications of the entire theory. Second, we provide a simple decision procedure for those problems which we can cast propositionally. This decision procedure allows us to computationally experiment with applications of temporal explanation.

We first present Lifschitz and Rabinov's approach. We follow theirs with our approach in the modal logic Z, present four examples, and finally make a few remarks.

LIFSCHITZ AND RABINOV'S APPROACH

A solution to the temporal explanation problem using the situational calculus and circumscription is proposed in [Lif&Rab89]. Their method is to use a law of change (1) to represent actions and a law of inertia (2) to represent the frame law, where a is the action, f is the fluent, and v is the value that action a causes f to take, unless a miracle occurs.

```
(1) (→ (∧  (success a s)(causes a f v))
       (∨  (= v (value f (result a s)))
           (mir f (result a s))))

(2) (→ ¬(affects a f s)
       (∨  (= (value f (result a s)) (value f s))
           (mir f (result a s))))
```

In other words, if the preconditions of action a are met, (success a s), then normally the action causes f to change its value in the resulting situation. The "normally" phrase is achieved by minimizing the miracle predicate. Miracles normally do not happen, but sometimes they are the only way to explain how a given situation is consistent. According to [Lif&Rab89], there are two kinds of miracles, negative and positive. Positive miracles occur when something not affected by an action changes its value from one time to the next. Negative miracles occur when some or all of the effects of an action are suppressed. All fluents not affected by the action have the same value in the resulting situation.

A limitation of [Lif&Rab89]'s approach is in the definition of affects (3). The definition of success is given by (4).

(3) (define (affects a f s)
(∧ (success a s) ∃v(causes a f v)))

(4) (define (success a s)
∀f(→ (precond f a) (= (value f s) #t)))

An action affects a fluent if and only if it directly changes the value of that fluent. In our approach, we e extend the definition such that an action affects a fact if and only if the action's conclusions directly (as in [Lif&Rab 89]) or indirectly (i.e. its ramifications) affect the fact, as we show below.

In addition to miracles, [Lif&Rab89] also circumscribes the precond and causes predicates to achieve explicit definitions for precond and causes. This circumscription, in effect, makes the domain finite because precond and causes are finitely specified, and thus the laws of change and inertia apply only to the known fluents. We define all of our laws finitely as well, but forego explicit circumscription.

OUR APPROACH

In general, we follow [Lif&Rab89], except that we use the modal logic Z for representation and its inference rules for nonmonotonic reasoning. We first briefly describe Z, then we describe our representation in Z. We follow this with a brief description of automated reasoning in Z, and finally, we present a few examples.

Introduction to the Modal Logic Z

The modal logic Z provides a consistency-based approach to nonmonotonic reasoning. Z is a fragment of second order modal quantificational logic involving quantification over propositions but not over properties. A thorough description of Z is given in [Bro91]. Defaults with respect to some body of facts are expressed using what are called reflective equations. A reflective equation has the form (≡ k (Φ k)) where k is a propositional constant (i.e. a globally scoped universally quantified variable such as a Skolem function), ≡ is the symbol for necessary equivalence, and (Φ k) is a formula of Z possibly involving k and usually including defaults. An example default is (→ (∧ a <k>b) b), which states that if a and it is possible (or consistent to assume) b with whatever else we know in k, then conclude b. The equation expresses that in any interpretation, we want k to be equivalent to (Φ k). The goal is to transform the initial equation into a disjunction, (∨(≡ k b_1) (≡ k b_2)...(≡ k b_n)) with each b_i free for k. If (≡ k b_i) implies the original equation and is free for k in b_i, then (≡ k b_i) is a solution to the original equation. We call the process of finding solutions to a reflective equation, reflective reasoning.

An important property of Z is that equations where k appears only in propositional contexts are decidably solvable. This has allowed the creation of an automated reasoning system for solving reflective equations. The reasoning system can also solve a significant number of quantifier-containing equations as well, but this capability is not needed here, and is not described.

Representing Temporal Explanation using Z

We follow the basic approach of [Lif&Rab89]. Where they use situations to mark time, we index time numerically as a subscript on predicates. For ease of presentation and reasoning, we use predicates instead of fluents. We define success to simply be the conjunction of all preconditions for a given action. We define the finite version of affects in (5) where a is the action, p is the predicate, x is the object of the predicate, R is the ramification theory, that is, a theory of consequences of actions, and i and i+1 are the times involved in the transition.

(5) (define (affects a p x R i i+1)
(∧ (success a i i+1)
(∨ [(∧ R (causes a i i+1))]p_{xi+1}
[(∧ R (causes a i i+1))] $\neg p_{xi+1}$)))

In (5), the notation $[\alpha]\beta$ is defined as β is a necessary consequence of α. In other words, the action a affects p if the preconditions are met and the direct consequences of the action ramify to affect either p_{xi+1} or $\neg p_{xi+1}$.

The laws of change and inertia are combined into the law of transition, which we define in (6) for a ramification theory R, an action a, a set of predicates, and a set of objects.

$$\begin{array}{l}
\text{(6) (define (transition R a i i+1 } \{p_1 \ldots p_m\}\ \{o_1 \ldots o_n\}) \\
\quad (\rightarrow occ_{ai} \\
\qquad (\wedge\ (\rightarrow \text{(success a i i+1)} \\
\qquad\qquad (\vee \text{ (causes a) } mir_{acti})) \\
\qquad\quad (\wedge s{=}1\ldots m, t{=}1\ldots n \\
\qquad\qquad (\rightarrow \neg(\text{affects a } (p_s\ o_t\ i{+}1)\ R) \\
\qquad\qquad\quad (\vee\ (mir\ (p_s\ o_t\ i{+}1)) \\
\qquad\qquad\qquad (\leftrightarrow (p_s\ o_t\ i)\ (p_s\ o_t\ i{+}1))))))))
\end{array}$$

Definition (6) says that if we know that an action a has occurred at time i, then if its preconditions are met, then either its effects or a negative miracle occurs, and all things not affected by the action keep their values unless a positive miracle occurs. The conditioning of the law of transition on occ_{ai} allows us to specify the behavior of multiple possible actions for a given time period without committing to the "execution" of the action. Whenever more than one action for a time period is given, we enforce mutual exclusion using the finite form of
$\forall i \exists x(\wedge\ occ_{xi}\ \forall y(\rightarrow \neg([=]\ x\ y)\ \neg occ_{yi}))$
so that only one action is allowed to execute.

We give four examples taken from the benchmark problems in [Lif89]. We demonstrate in [Lea93] solutions for all 6 of the temporal explanation benchmark problems. In each of these examples, we minimize the miracles by the use of defaults like the following: $(\rightarrow \langle k\rangle\neg(mir\ (p_s\ o_t\ i{+}1))\ \neg(mir\ (p_s\ o_t\ i{+}1)))$. Since we restrict actions to not occur concurrently, we minimize a miracle for action0, action1, etc. without being specific about which action occurs. To simplify the presentation, we recursively define in (7) a syntactic short-hand for miracles.

$$\begin{array}{l}
\text{(7) (define (minMir) \#t)} \\
\quad \text{(define (minMir a} \ldots \text{b)} \\
\qquad (\wedge\ (\rightarrow \langle k\rangle\neg(mir\ a)\ \neg(mir\ a))\ (minMir \ldots b)))
\end{array}$$

Propositional Reasoning in Z

When a reflective equation is propositional in all modal contexts, we can simplify it using case analysis. For example, consider the case where B is not on the table, and A is not on the table. The expected result is that A is on the table.

$$(\equiv k\ \ (\wedge\ \ \neg on_b\ (\rightarrow \langle k\rangle on_a\ on_a)))$$

= class. logic since $(\vee\ \langle k\rangle on_a\ \neg\langle k\rangle on_a)$ iff #t

$$\begin{array}{l}
(\wedge\ \ (\vee\ \langle k\rangle on_a\ \neg\langle k\rangle on_a) \\
\quad (\equiv k\ (\wedge\ \neg on_b\ (\rightarrow \langle k\rangle on_a\ on_a))))
\end{array}$$

= classical logic (distribution)

$$\begin{array}{l}
(\vee\ \ (\wedge\ \ \langle k\rangle on_a \\
\qquad (\equiv k\ (\wedge\ \neg on_b\ (\rightarrow \langle k\rangle on_a\ on_a)))) \\
\quad (\wedge\ \ \neg\langle k\rangle on_a \\
\qquad (\equiv k\ (\wedge\ \neg on_b\ (\rightarrow \langle k\rangle on_a\ on_a)))))
\end{array}$$

= modalized assumption idiom, i.e., $\langle k\rangle on_a$ iff $[]\langle k\rangle on_a$ iff $(\equiv \langle k\rangle on_a\ \#t)$ and $\neg\langle k\rangle on$ iff $[]\neg\langle k\rangle on_a$ iff $(\equiv \neg\langle k\rangle on_a\ \#t)$, so we can substitute #t and #f within the equation, resp.

$$\begin{array}{l}
(\vee\ \ (\wedge\ \ \langle k\rangle on_a \\
\qquad (\equiv k\ (\wedge\ \neg on_b\ (\rightarrow \#t\ on_a)))) \\
\quad (\wedge\ \ \neg\langle k\rangle on_a \\
\qquad (\equiv k\ (\wedge\ \neg on_b\ (\rightarrow \#f\ on_a)))))
\end{array}$$

= classical logic

$$\begin{array}{l}
(\vee\ \ (\wedge\ \langle k\rangle on_a\ (\equiv k\ (\wedge\ \neg on_b\ on_a))) \\
\quad (\wedge\ \neg\langle k\rangle on_a\ (\equiv k\ \neg on_b)))
\end{array}$$

= substitute $(\wedge\ \neg on_b\ on_a)$ for k in $\langle k\rangle on_a$ and $\neg on_b$ for k in $\neg\langle k\rangle on_a$

$$\begin{array}{l}
(\vee\ \ (\wedge\ \Diamond(\wedge\ \neg on_b\ on_a)\ (\equiv k\ (\wedge\ \neg on_b\ on_a))) \\
\quad (\wedge\ \neg\Diamond(\wedge\ \neg on_b\ on_a)\ (\equiv k\ \neg on_b)))
\end{array}$$

= $\Diamond(\wedge\ \neg on_b\ on_a)$ is #t

$$(\vee\ (\wedge\ \#t\ (\equiv k\ (\wedge\ \neg on_b\ on_a)))\ (\wedge\ \#f\ (\equiv k\ \neg on_b))))$$

= classical logic

$$(\equiv k\ (\wedge\ \neg on_b\ on_a))$$

In other words, A is on the table, but B is not.

Example 1: Temporal Explanation with Unknown Initial Conditions

Assumptions:

After an action is performed, things normally remain as they were.
When the robot grasps a block, the block will be normally in the hand.
When the robot moves a block onto the table, the block will be normally on the table.
Moving a block that is not on the hand is an exception to this rule.
Initially block A was not on the table.
After the robot moved A onto the table and then waited, A was on the table.

Conclusion:

Initially A was in the hand.

In this problem, we must recognize that the results of the actions have been achieved, and that, barring miracles, this is because the preconditions for the actions, which are not completely specified in the problem, are nevertheless true in the initial time period. We achieve the desired results using the mechanism described above. We define in (8) the preconditions for moving to be that the object being moved during the transition from i to i+1 must be held. We define in (9) that the consequences of moving x to the table during the transition from i to i+1 to be on_{xi+1}. That the wait action from i to i+1 has no preconditions nor consequences is defined in (10) and (11). We use the definitions of affects and transition as given in the introduction to this section. We represent the description of the action "move a to the table" using (transition #t m_a 0 1 {h on} {a}), and the description of the wait action using (transition #t w 0 1 {h on} {a}). The #t in both transitions stands for the empty ramification theory. The known information about the initial condition is that A is not on the table: $\neg on_{a0}$. We are told that after the two actions, occ_{m0} and occ_{w1}, A is on the table: on_{a2}.

(8) (define (success (m x) i i+1) h_{xi})
(9) (define (causes (m x) i i+1) on_{xi+1})
(10) (define (success (w) i i+1) #t)
(11) (define (causes (w) i i+1) #t)

A rough description of the reasoning for this problem is as follows. We know on_{a2}, and assuming no miracle occurs, waiting does not affect the on-ness of A from time 1 to time 2, so we must know on_{a1}. Now, since on_{a1}, and assuming no miracle occurs, we know that on_{a1} is a result of the move action, and hence, that h_{a0} must be true.

$\exists$k($\wedge$k($\equiv$k($\wedge$ $\neg on_{a0}$
(transition #t m_a 0 1 {h on} {a})
(transition #t w 0 1 {h on} {a})
occ_{m0} occ_{w1} on_{a2}
(minMir act_0 h_{a1} on_{a1} act_1 h_{a2} on_{a2}))))

= Machine proof by case analysis and disjunctive quantifier idiom. Because there are 6 defaults, the case analysis involves 2^6 cases.

($\wedge$ h_{a0} h_{a1} h_{a2} occ_{m0} occ_{w1} $\neg on_{a0}$ on_{a1} on_{a2}
$\neg mir_{act0}$ $\neg mir_{ha1}$ $\neg mir_{ona1}$ $\neg mir_{act1}$
$\neg mir_{ha2}$ $\neg mir_{ona2}$)

=>
h_{a0} (Block A is held in the initial situation.)

qed.

Example 2: Temporal Explanation with Unknown Actions

Assumptions:

After an action is performed, things normally remain as they were.
When the robot grasps a block, the block will be normally in the hand.
When the robot moves a block on to the table, the block will be normally on the table.
Moving a block that is not in the hand is an exception to this rule.
Initially block A was not on the table.
Initially block A was not in the hand.
After the robot grasped some block and then moved some block onto the table, A was on the table.

Conclusions:

The block that was grasped was A.
The block that was moved on the table was A.

This problem tests whether we can determine the specific actions knowing the generic actions, but not the objects involved. We begin by defining the grasp action using (12) and (13). Something may be grasped, without precondition, resulting in the object normally being held, beginning at time i and ending at time i+1. We represent that a non-specific grasp action followed by a non-specific move action take place by writing the finite domain forms of (14) and (15) using (16) and (17), resp. To ensure that either grasping B or moving B is possible, we also include their respective transitions as well as the ones for A. As in example 1, the ramification theory is empty.

(12) (define (success (g x) i i+1) #t)
(13) (define (causes (g x) i i+1) h_{xi+1})
(14) $\exists$c($\wedge$ occ_{gc0} $\forall$x($\rightarrow$ $\neg$([=] x c) $\neg occ_{gx0}$))
(15) $\exists$c($\wedge$ occ_{mc1} $\forall$x($\rightarrow$ $\neg$([=] x c) $\neg occ_{mx1}$))
(16) ($\leftrightarrow$ occ_{ga0} $\neg occ_{gb0}$)
(17) ($\leftrightarrow$ occ_{ma1} $\neg occ_{mb1}$)

$$
\begin{array}{l}
\exists k(\wedge k(\equiv k\ \ (\wedge\ \ \neg on_{a0}\ \neg h_{a0}\ on_{a2} \\
\qquad\qquad (\leftrightarrow occ_{ga0}\ \neg occ_{gb0}) \\
\qquad\qquad (\leftrightarrow occ_{ma1}\ \neg occ_{mb1}) \\
\qquad\qquad \text{(transition \#t ga 0 1 \{h on\} \{a b\})} \\
\qquad\qquad \text{(transition \#t gb 0 1 \{h on\} \{a b\})} \\
\qquad\qquad \text{(transition \#t ma 1 2 \{h on\} \{a b\})} \\
\qquad\qquad \text{(transition \#t mb 1 2 \{h on\} \{a b\})} \\
\qquad\qquad (\mathrm{minMir}\ act_0\ h_{a1}\ on_{a1}\ act_1\ h_{a2}\ on_{a2})\ )))
\end{array}
$$

= machine case analysis and disjunctive quantifier idiom

$$
\begin{array}{l}
(\wedge\ \ occ_{ga0}\ \neg occ_{gb0}\ occ_{ma1}\ \neg occ_{mb1} \\
\quad \neg h_{a0}\ h_{a1}\ h_{a2}\ \neg on_{a0}\ \neg on_{a1}\ on_{a2} \\
\quad \neg mir_{act0}\ \neg mir_{ha1}\ \neg mir_{ona1} \\
\quad \neg mir_{act1}\ \neg mir_{ha2}\ \neg mir_{ona2})
\end{array}
$$

=> classical logic

$(\wedge\ occg_{a0}\ occm_{a1})$

In other words, the object grasped at time 0 was A, and the object moved at time 1 was also A.

qed.

Example 3: Reasoning about Unexpected Change.

Assumptions:

After an action is performed, things normally remain as they were.
When the robot moves a block to another location, the block will be normally at that location.
After the robot moved a block to Location 1 and then to Location 2, the block changed its color.

Conclusion:

The block changed its color only once, either after the first move, or after the second.

In this problem, we see the usefulness of miracles for explaining the otherwise inexplicable. The normal behavior of our actions cannot explain why a block changes its color when it is moved. Any number of outside explanations are possible, including "the block was moved under a red light" and "the block was placed under a stream of paint." The miracle does not suggest an explanation; it only indicates that the explanation lies outside the theory.

We represent the presence of a block at location 1 during time interval i by $L1_i$. Similarly, a block at location 2 during time i is represented as $L2_i$. We assume there are only two colors in the domain, red and not red. We represent that the block being moved is red at time i by r_i. We define moving the block to location 1 as an action independent from the action of moving the block to location 2. We assume in (18) and (19) that there are no preconditions for either action. We define in (20) and (21) that the consequences of the move action are that the block is at its respective location.

(18) (define (success $mL1_0$) #t)
(19) (define (success $ml2_1$) #t)
(20) (define (causes $mL1_0$) $L1_1$)
(21) (define (causes $ml2_1$) $L2_2$)

$$
\begin{array}{l}
\exists k\ (\wedge k \\
\quad (\equiv k\ (\wedge\ \ \text{(transition \#t occ mL1 0 1 \{L1 L2\} \{ \})} \\
\qquad\qquad\quad \text{(transition \#t occ mL2 1 2 \{L1 L2\} \{ \}))} \\
\qquad\qquad\quad r_0\ \neg r_2\ occmL1_0\ occmL2_1 \\
\qquad\qquad\quad (\mathrm{minMir}\ act0\ L10\ L20\ r0 \\
\qquad\qquad\qquad\quad act1\ L11\ L21\ r1)\ )))
\end{array}
$$

= machine analysis

$$
\begin{array}{l}
(\vee\ \ (\wedge\ \ L1_1\ L1_2\ L2_2\ occmL1_0\ occmL2_1 \\
\qquad\quad r_0\ \neg r_1\ \neg r_2\ (\leftrightarrow L2_0\ L2_1) \\
\qquad\quad \neg mir_{act0}\ \neg mirL1_0\ \neg mirL2_0\ mirr_0 \\
\qquad\quad \neg mir_{act1}\ \neg mirL1_1\ \neg mirL2_1\ \neg mirr_1) \\
\quad\ \ (\wedge\ \ L1_1\ L1_2\ L2_2\ occmL1_0\ occmL2_1 \\
\qquad\quad r_0\, r_1\ \neg r_2\ (\leftrightarrow L2_0\ L2_1) \\
\qquad\quad \neg mir_{act0}\ \neg mirL1_0\ \neg mirL2_0\ \neg mirr_0 \\
\qquad\quad \neg mir_{act1}\ \neg mirL1_1\ \neg mirL2_1\ mirr_1)\ )
\end{array}
$$

=> classical logic

$(\vee\ (\wedge\ r_0\ \neg r_1\ \neg r_2)\ (\wedge\ r_0\ r_1\ \neg r_2))$, i.e. the block changed its color only once, either in transition from 0 to 1, or in transition from 1 to 2.

qed.

Example 4: Reasoning about the Unexpected Absence of Change.

Assumptions:

When the robot moves a block to another location, the block will normally be at that location.
After the robot moved block A onto the table, and then moved block B onto the table, at most one of the blocks A, B was on the table.

Conclusion: After the two actions were performed, exactly one of the blocks A, B was on the table.

In this problem, one of two sequential move actions fails to make a change, requiring a negative miracle to explain the absence of change. We define in (22) that the move has no prerequisite. Its consequence is defined in (23).

(22) (define (success m_{xi}) #t).

(23) (define (causes m_{xi}) on_{xi+1}).

$$
\begin{array}{l}
\exists k(\wedge\ k \\
\quad (\equiv k\ (\wedge\ \ (\text{transition \#t occ } m_a\ 0\ 1\ \{on\}\ \{a\ b\}) \\
\qquad\qquad (\text{transition \#t occ } m_b\ 1\ 2\ \{on\}\ \{a\ b\}) \\
\qquad\qquad occm_{a0}\ occm_{b1} \\
\qquad\qquad (\vee\ \ \neg on_{a2}\ \neg on_{b2}) \\
\qquad\qquad (\text{minMir } act_0\ on_{a0}\ on_{b0}\ act_1\ ona_1\ onb_1)\ )))
\end{array}
$$

= reflective reasoning

$$
\begin{array}{l}
(\vee\ \ (\wedge\ \ occm_{a0}\ occm_{b1}\ \neg on_{a1}\ \neg on_{a2} \\
\qquad\quad (\leftrightarrow on_{b0}\ on_{b1})\ on_{b2} \\
\qquad\quad mir_{act0}\ \neg miron_{b0}\ \neg mir_{act1}\ \neg miron_{a1}) \\
\quad\ (\wedge\ \ occm_{a0}\ occm_{b1}\ on_{a1}\ on_{a2} \\
\qquad\quad (\leftrightarrow on_{b0}\ on_{b1})\ \neg on_{b2} \\
\qquad\quad \neg mir_{act0}\ \neg miron_{b0}\ mir_{act1}\ \neg miron_{a1}) \\
\quad\ (\wedge\ \ occm_{a0}\ occm_{b1}\ on_{a1}\ \neg on_{a2} \\
\qquad\quad (\leftrightarrow on_{b0}\ on_{b1})\ on_{b2} \\
\qquad\quad \neg mir_{act0}\ \neg miron_{b0}\ \neg mir_{act1}\ miron_{a1}))
\end{array}
$$

=> classical logic

$(\leftrightarrow \neg on_{b2}\ on_{a2})$, i.e. exactly one of the blocks A, B is on the table at time 2."

qed.

CONCLUSION

We have shown that the circumscribed situational calculus solution of [Lif&Rab89] can be represented in Z and automatically solved, owing to the propositional representation. We believe this is an important accomplishment because it allows experimental applications to be built that employ temporal explanation as a reasoning component.

References

[Bro91] Brown, Frank M., "The Modal Quantificational Logic Z Applied to the Frame Problem", *International Journal of Expert Systems Research and Applications, Special Issue: The Frame Problem. Part A.* eds. Kenneth Ford and Patrick Hayes vol. 3 number 3, pp169-206 JAI Press 1991.

[Lea93] David Elden Leasure, *The Modal Logic Z Applied To Lifschitz's Benchmark Problems For Formal Nonmonotonic Reasoning,* Doctoral Dissertation, Dept. of Computer Science, The University of Kansas, Lawrence, KS, April 1993.

[Lif89] Vladimir Lifschitz, "Benchmark Problems for Formal Nonmonotonic Reasoning, Version 2.00," in M. Reinfrank, J. de Kleer, M. L. Ginsberg, and E. Sandewall (Eds.), *Non-Monotonic Reasoning -- Proceedings of the 2nd. International Workshop, Grassau, FRG, June 1988*, *Lecture Notes in Artificial Intelligence 346* (1989) Springer Verlag, pp. 202-219.

[Lif&Rab89] Vladimir Lifschitz and Arkady Rabinov, "Miracles in Formal Theories of Action," *Artificial Intelligence* 38 (1989) 225-237.