Lecture 6 – The Monkey and Bananas Problem

1. Introduction

2. Green's original axiomatization (unreified)

3. Frame and qualification problems

4. Green's original axiomatization (reified)

5. Solutions for Frame and qualification problems

6. Green's axiomatization modified.

7. Using the modified axiomatization with conjectures

Introduction

– Monkey and bananas problem originally proposed by McCarthy as a "toy" problem for studying problem solving behavior.

Problem:

A monkey is faced with the problem of fetching a bunch of bananas from the ceiling. To do this, he must reason that: he should

1. push a box under the bananas,
2. climb the box,
3. reach for the bananas.

Given: An axiomatization of

1. The initial situation.
2. A description of actions and domain constraints.
3. A theorem prover.

Desired:

- The TP should generate a sequence of actions that permits the monkey to acquire the bananas when the TP is queried as to whether there is a state in which the monkey has the bananas.

Green's Axiomatization

– Green considered the use of theorem-proving for solving problems. In particular, he combined the situation calculus with a resolution theorem prover to construct plans.

– Original axiomatization was essentially unreified, though actions were represented as functions returning situations.

– The unreified version I present is technically the same as Green's but I make explicit the sorts.

– Green's Article.

Unreified Version

We will use the following sorts:

1. situations (s),
2. monkeys (m),
3. locations (l),
4. boxes (b),
5. bananas (z),
6. things (x) $x = m \oplus b \oplus z$.

– Our vocabulary, $\Lambda = \langle C, F, R, d \rangle$, contains the following:

1. $C = C_s \cup C_m \cup C_l \cup C_b \cup C_z$ where

$$C_s = \{S0\} \text{ and } \forall x \in C_s.d(x) = s,$$

$$C_m = \{Monkey\} \text{ and } \forall x \in C_m.d(x) = m,$$

$$C_l = \{Place_b, UnderBananas\} \text{ and } \forall x \in C_l.d(x) = l,$$

$$C_b = \{Box\} \text{ and } \forall x \in C_b.d(x) = b,$$

$$C_b = \{Bananas\} \text{ and } \forall x \in C_z.d(x) = z,$$

2. $F = \{reach, climb, move\}$ where

$$d(reach) = \langle m \times z \times s \to s \rangle,$$

$$d(climb) = \langle m \times b \times s \to s \rangle,$$

$$d(move) = \langle m \times b \times l \times s \to s \rangle,$$

3. $R = \{Movable, At, On, Has, Climbable, Reachable\}$ where

$$d(Movable) = \langle b \times s \rangle.$$

$$d(At) = \langle x \times l \times s \rangle.$$

$$d(On) = \langle m \times b \times s \rangle.$$

$$d(Has) = \langle m \times z \times s \rangle.$$

$$d(Climbable) = \langle m \times b \times s \rangle.$$

$$d(Reachable) = \langle m \times z \times s \rangle.$$

– The list of variables for sorts is:

$$\xi_1^s, \xi_2^s, \xi_3^s, \ldots, \xi_1^m, \xi_2^m, \xi_3^m, \ldots \text{etc.}$$

To make the axioms more readable we will use the sort type (possibly subscripted) for variables:

$$s, s_1, s_2, s_3, \ldots, m, m_1, m_2, m_3, \ldots \text{etc.}$$

Axioms for the initial situation:

$$At(Box, Place_b, S0) \tag{1}$$

$$\forall x.\neg At(x, UnderBananas, S0) \tag{2}$$

Axioms for the Laws of Motion:

$$\begin{aligned}\forall b, l, l', s.[At(b, l, s) \wedge Movable(b, s) \wedge \forall x.\neg At(x, l', s)] \rightarrow \\ [At(b, l', move(Monkey, b, l', s)) \wedge \\ At(Monkey, l', move(Monkey, b, l', s))]\end{aligned} \tag{3}$$

$$\begin{aligned}\forall m, l, b, s.[At(b, l, s) \wedge Climbable(m, b, s)] \rightarrow \\ [At(b, l, climb(m, b, s)) \wedge \\ On(m, b, climb(m, b, s))]\end{aligned} \tag{4}$$

$$\forall m, z, s.Reachable(m, z, s) \rightarrow Has(m, z, reach(m, z, s)) \tag{5}$$

Axioms for state constraints:

$$\begin{aligned}\forall s.[At(Box, UnderBananas, s) \wedge On(Monkey, Box, s)] \rightarrow \\ Reachable(Monkey, Bananas, s)\end{aligned} \tag{6}$$

$$\forall s.Climbable(Monkey, Box, s) \tag{7}$$

$$\forall s.Movable(Box, s) \tag{8}$$

Querying the System

Query:

"Is there a state s in which the monkey has the bananas?"

$$\exists s.Has(Monkey, Bananas, s)? \tag{9}$$

– Although a theorem prover could prove the query statement given the axioms, we desire a binding for s also.

– Given a resolution theorem prover the following constructive reply can be generated from the axiomatization and the negation of the query:

$$S3 = reach(Monkey, Bananas, climb(Monkey, Box, move(Monkey, Box, UnderBananas, S0))) \tag{10}$$

The Frame Problem

Generally, the effects of actions are limited to a small number of changes in state, whereas the majority of properties holding in a state, where an action is executed, normally continue to hold in the state resulting from the execution of an action, unless explicitly stated not to hold.

– Observe the law of motion, axiom (4):

$$\begin{aligned}\forall m, l, b, s.[At(b, l, s) \wedge Climbable(m, b, s)] \rightarrow \\ [At(b, l, climb(m, b, s)) \wedge \\ On(m, b, climb(m, b, s))]\end{aligned}$$

– To make this more explicit, axiom (4) can be split into two axioms:

$$\begin{aligned}\forall m, l, b, s.[At(b, l, s) \wedge Climbable(m, b, s)] \rightarrow \\ On(m, b, climb(m, b, s))\end{aligned} \tag{11}$$

$$\forall m, l, b, s.At(b, l, s) \rightarrow At(b, l, climb(m, b, s)) \tag{12}$$

– Latter is called a *frame axiom*. Better rep. of action law in terms of preconds and direct effects.

– Still a problem with inference, flexibilty and qualification.

– Consider the following query:

"After the monkey reaches for the bananas and has them, is he still on the box?"

– A query to this effect is:

$$On(Monkey, Box, S3) \tag{13}$$

where,

$$\begin{aligned} S3 = reach(&Monkey, Bananas, climb(Monkey, Box, \\ &move(Monkey, Box, UnderBananas, S0))) \end{aligned} \tag{14}$$

– Commonsense tells us that the answer should be yes, but our theory does not allow us to infer that this is the case.

– Why? We lack frame axioms for $reach$

– Adding the following frame axiom would allow us to derive $On(Monkey, Box, S3)$:

$$\forall m, b, z, s. On(m, b, s) \rightarrow On(m, b, reach(m, z, s))$$

– **Initial Solution to the Frame Problem: Add frame axioms.**

Modified Axioms for the Laws of Motion:

$$\forall b,l,l',s.[At(b,l,s)\wedge Movable(b,s)\wedge\forall x.\neg At(x,l',s)]\rightarrow [At(b,l',move(Monkey,b,l',s))\wedge At(Monkey,l',move(Monkey,b,l',s))] \quad (15)$$

$$\forall m,l,b,s.[At(b,l,s)\wedge Climbable(m,b,s)]\rightarrow On(m,b,climb(m,b,s)) \quad (16)$$

$$\forall m,z,s.Reachable(m,z,s)\rightarrow Has(m,z,reach(m,z,s)) \quad (17)$$

Frame Axioms:

$$\forall m,l,b,b',s.At(b,l,s)\rightarrow At(b,l,Climb(m,b',s)) \quad (18)$$

$$\forall m,l,b,z,s.At(b,l,s)\rightarrow At(b,l,Reach(m,z,s)) \quad (19)$$

$$\forall m,b,z,s.On(m,b,s)\rightarrow On(m,b,Reach(m,z,s)) \quad (20)$$

$$\forall m,b,z,l,s.Has(m,z,s)\rightarrow Has(m,z,Move(m,b,l,s)) \quad (21)$$

– n actions and m properties $\Longrightarrow$ $n\times m$ frame axioms.
– What axioms should be added?
– Still a lack of flexibility.

<u>Reified Version</u>

We will use the following sorts:

1. situations (s),
2. **fluents (f)**, **
3. **actions (a)**, **
4. monkeys (m),
5. locations (l),
6. boxes (b),
7. bananas (z),
8. things (x) $x = m \oplus b \oplus z$.

Our vocabulary, $\Lambda = \langle C, F, R, d \rangle$, contains the following:

1. $C = C_s \cup C_m \cup C_l \cup C_b \cup C_z$ where

$$C_s = \{S0\} \text{ and } \forall x \in C_s . d(x) = s,$$

$$C_m = \{Monkey\} \text{ and } \forall x \in C_m . d(x) = m,$$

$$C_l = \{Place_b, UnderBananas\} \text{ and } \forall x \in C_l . d(x) = l,$$

$$C_b = \{Box\} \text{ and } \forall x \in C_b . d(x) = b,$$

$$C_z = \{Bananas\} \text{ and } \forall x \in C_z . d(x) = z,$$

2. $F = \{reach, climb, move, movable, at, on, has\}$
$\cup\{climbable, reachable, result\}$ where

$$d(reach) = \langle m \times z \to a \rangle,$$

$$d(climb) = \langle m \times b \to a \rangle,$$

$$d(move) = \langle m \times b \times l \to a \rangle,$$

$$d(movable) = \langle b \times s \to f \rangle.$$

$$d(at) = \langle x \times l \to f \rangle.$$

$$d(on) = \langle m \times b \to f \rangle.$$

$$d(has) = \langle m \times z \to f \rangle.$$

$$d(climbable) = \langle m \times b \to f \rangle.$$

$$d(reachable) = \langle m \times z \to f \rangle.$$

$$d(occupied) = \langle l \to f \rangle.$$

$$d(unmovable) = \langle b \to f \rangle.$$

$$d(tooheavy) = \langle b \to f \rangle.$$

$$d(unreachable) = \langle m \times z \to f \rangle.$$

$$d(result) = \langle a \times s \to s \rangle.$$

3. $R = \{Holds, Prevents\}$ where

$$d(Holds) = \langle f \times s \rangle.$$

$$d(Prevents) = \langle f \times f \times s \rangle.$$

The list of variables for sorts is:

$$\xi_1^s, \xi_2^s, \xi_3^s, \ldots, \xi_1^m, \xi_2^m, \xi_3^m, \ldots \text{etc.}$$

To make the axioms more readable we will use the sort type (possibly subscripted) for variables:

$$s, s_1, s_2, s_3, \ldots, f, f_1, f_2, f_3, \ldots \text{etc.}$$

Axioms for the initial situation:

$$Holds(at(Box, Place_b), S0) \tag{22}$$

$$\forall x.\neg Holds(at(x, UnderBananas), S0) \tag{23}$$

Axioms for the Laws of Motion:

$$\begin{aligned}\forall b, l, l', s.[Holds(at(b, l), s) \wedge Holds(movable(b), s) \wedge \\ \forall x.\neg Holds(at(x, l'), s)] \rightarrow \\ [Holds(at(b, l'), result(move(Monkey, b, l'), s)) \wedge \\ Holds(at(Monkey, l'), result(move(Monkey, b, l'), s))]\end{aligned} \tag{24}$$

$$\begin{aligned}\forall m, l, b, s.[Holds(at(b, l), s) \wedge Holds(climbable(m, b), s)] \rightarrow \\ [Holds(at(b, l), result(climb(m, b), s)) \wedge \\ Holds(on(m, b), result(climb(m, b), s))]\end{aligned} \tag{25}$$

$$\begin{aligned}\forall m, z, s.[Holds(reachable(m, z), s) \rightarrow \\ Holds(has(m, z), result(reach(m, z), s)\end{aligned} \tag{26}$$

Axioms for state constraints:

$$\begin{aligned}\forall s.[Holds(at(Box, UnderBananas), s) \wedge \\ Holds(on(Monkey, Box), s)] \rightarrow \\ Holds(reachable(Monkey, Bananas), s)\end{aligned} \tag{27}$$

$$\forall s.Holds(climbable(Monkey, Box), s) \tag{28}$$

$$\forall s.Holds(movable(Box), s) \tag{29}$$

Querying the System

Query:

"Is there a state s in which the monkey has the bananas?"

$$\exists s.Holds(has(Monkey, Bananas), s)? \tag{30}$$

– We'd like a theorem prover to return

$$\begin{aligned} s = result(&reach(Monkey, Bananas), \\ &result(climb(Monkey, Box), \\ &result(move(Monkey, Box, UnderBananas), S0))) \end{aligned} \tag{31}$$

The Frame Problem

– The reified version provides a nice way to state a general frame rule of the type

"unless explicitely stated a state property normally does not change value in a state resulting from the execution of an action."

– This is made possible because we can quantify over actions, states and properties.

Let f, a, and s be sort variables of type fluent, action and situation, respectively,

$$\forall f, a, s. Holds(f, s) \wedge \neg Ab(f, a, s) \rightarrow Holds(f, result(a, s)) \quad (32)$$

If a fluent f holds in state s and that property is not abnormal relative to the action a occuring in state s, then that fluent will persist or retain the same value in the state resulting from the execution of a in s.

Idea: The general frame axiom plus explicit statements as to the abnormality of an action to a fluent replaces the individual frame axioms.

Example: at is abnormal relative to $move$

$$\forall s, b, l, l'.Ab(at(b, l), move(b, l'), s) \tag{33}$$

The fluent $at(b, l)$ does not persist when the $move$ action is executed.

– Note that rather than representing the facts that do not change, we represent only those that we have explicit information about as changing.

Problem:

"How do we prove that fluents are normal relative to actions?"

$$\neg Ab(f, a, s)?$$

or more specifically,

$$\neg Ab(at(b, l), move(b, l'), s)?$$

The Qualification Problem

$$\forall m, l, b, s.[Holds(at(b,l),s) \wedge Holds(climbable(m,b),s)] \rightarrow$$
$$[Holds(at(b,l), result(climb(m,b),s)) \wedge$$
$$Holds(on(m,b), result(climb(m,b),s))]$$

– Note that the $climb$ move is performable **if**

$$Holds(climbable(m,b),s)$$

.

– A state constraint states that

$$\forall s.Holds(climbable(Monkey, Box), s)$$

.

– Together, they represent a rigid qualification. There are no exceptions and $climbable$ can be viewed as a blanket term for all qualifications or simply as one.

– Obviously, this will not do. We would like a flexible means to state qualifications that is epistemologically adequate.

"*UNLESS* we can show that for a monkey the box is unclimbable, assume it isn't.

"*UNLESS* we can show that somthing prevents us from performing the $climb$ action assume there isn't anything.

– Based on our discussions let's look at an alternate axiomatization with the frame and qualification solutions incorporated.

Axioms for the initial situation:

$$Holds(at(Box, Place_b), S0) \tag{34}$$

$$Holds(at(Monkey, Place_b), S0) \tag{35}$$

$$\forall x. \neg Holds(at(x, UnderBananas), S0) \tag{36}$$

Axioms for the Laws of Motion:

$$\begin{aligned} \forall m, b, l, l', s. [Holds(at(b, l), s) \wedge Holds(at(m, l), s) \wedge \\ (\forall f. \neg Prevents(f, move(m, b, l'), s))] \rightarrow \\ [Holds(at(b, l'), Result(move(m, b, l'), s)) \wedge \\ Holds(at(m, l'), Result(move(m, b, l'), s))] \end{aligned} \tag{37}$$

$$\begin{aligned} \forall m, l, b, s. [Holds(at(b, l), s) \wedge \\ (\forall f. \neg Prevents(f, climb(m, b), s))] \rightarrow \\ Holds(on(m, b), result(climb(m, b), s)) \end{aligned} \tag{38}$$

$$\begin{aligned} \forall m, z, s. (\forall f. \neg Prevents(f, reach(m, z), s)) \rightarrow \\ Holds(has(m, z), result(reach(m, z), s) \end{aligned} \tag{39}$$

Qualification Axioms:

$$\begin{aligned}\forall m, x, b, l', s.Holds(at(x, l'), s) \rightarrow \\ Prevents(occupied(l'), move(m, b, l'), s)\end{aligned} \tag{40}$$

$$\begin{aligned}\forall m, b, l', s.Holds(tooHeavy(b), s) \rightarrow \\ Prevents(unmovable(b), move(b, l'), s)\end{aligned} \tag{41}$$

$$\begin{aligned}\forall m, b, z, l, s.[\neg Holds(at(b, l), s) \vee \\ \neg Holds(on(m, b), s)] \rightarrow \\ Prevents(unreachable(m, z), reach(m, z), s)\end{aligned} \tag{42}$$

Frame Axioms:

$$\forall f, a, s.Holds(f, s) \wedge \neg Ab(f, a, s) \rightarrow Holds(f, result(a, s)) \tag{43}$$

$$\forall m, s, b, l, l'.Ab(at(b, l), move(m, b, l'), s) \tag{44}$$

$$\forall m, s, b, l, l'.Ab(at(m, l), move(m, b, l'), s) \tag{45}$$

Axioms for state constraints:

Empty.

<u>Discussion</u>

1. Epistemological Adequacy
   (a) Incrementality of the KB.
   (b) Flexibility of the action effect axioms.
   (c) Conceptual cleanliness of the representation.

2. Heuristic Adequacy
   (a) Representing what changes rather than what doesn't.
   (b) Representing what we know qualifies rather than what might possibly qualify.
   (c) In terms of theorem proving.

– Flexibility is wishful thinking. It is not yet built in. To get Ab and $Prevents$ to work, we must "circumscribe" their extensions.

Ex1 Proving No Qualifications

– Suppose we wanted to show that nothing prevents the monkey from moving Box to $UnderBananas$. Based on the alt. axiomatization including,*

$$Holds(at(Box, Place_b), S0) \tag{46}$$

$$Holds(at(Mky, Place_b), S0) \tag{47}$$

$$\begin{aligned}\forall m, b, l, l', s.[Holds(at(b, l), s) \wedge Holds(t(m, l), s) \wedge \\ (\forall f. \neg Prevents(f, move(m, b, l'), s))] \rightarrow \\ [Holds(at(b, l'), Result(move(m, b, l'), s)) \wedge \\ Holds(at(m, l'), Result(move(m, b, l'), s))]\end{aligned} \tag{48}$$

– we can instantiate (48) resulting in,

$$\begin{aligned}&[Holds(at(Box, Place_b), S0) \wedge Holds(at(Mky, Place_b), S0) \\ &\wedge (\forall f. \neg Prevents(f, move(Mky, Box, UBa), S0))] \rightarrow \\ &[Holds(at(Box, UBa), result(move(Mky, Box, UBa), S0)) \\ &\wedge Holds(at(Mky, UBa), result(move(Mky, Box, UBa), S0))]\end{aligned}$$

– Modus ponens + deduction theorem allows us to derive

$$\begin{aligned}&(\forall f. \neg Prevents(f, move(Mky, Box, UBa), S0)) \rightarrow \\ &[Holds(at(Box, UBa), Result(move(Mky, Box, UBa), S0)) \wedge \\ &Holds(at(Mky, UBa), Result(move(Mky, Box, UBa), S0))]\end{aligned}$$

– Here, we run into a problem. How do we show that

$$\forall f. \neg Prevents(f, move(Mky, Box, UBa), S0)? \tag{49}$$

*We will take the liberty of using abbreviations for constants: $Monkey$ is Mky, $UnderBananas$ is UBa.

– The only related axioms are (40) and (41),

$$\forall m, x, l', b, s. Holds(at(x, l'), s) \to Prevents(occupied(l'), move(m, b, l'), s)$$

$$\forall m, b, l', s. Holds(tooHeavy(b), s) \to Prevents(unmovable(b), move(m, b, l'), s)$$

or the equivalent contrapositions (instantiated):

$$\forall x. \neg Prevents(occupied(UBa), move(Mky, Box, UBa), S0) \to \neg Holds(at(x, UBa), S0) \qquad (40')$$

$$\neg Prevents(unmovable(Box), move(Mky, Box, UBa), S0) \to \neg Holds(tooHeavy(Box), S0) \qquad (41')$$

If (49) could somehow be conjectured then the following instances could be derived

$$\neg Prevents(occupied(UBa), move(Mky, Box, UBa), S0),$$

$$\neg Prevents(unmovable(Box), move(Mky, Box, UBa), S0),$$

then via (40') and (41'), we could conclude

$$\forall x. \neg Holds(at(x, UBa), S0)$$

$$\neg Holds(tooHeavy(Box), S0)$$

which state that there are no objects at location UBa and that the Box is not too heavy.

Ex1: A problem with conjectures

The problem is how to incorporate the conjecture (49) into our axiomatization. It is clearly not classically derivable .

– Suppose we had (49), then we could use axiom (37) to conclude that the $Monkey$ can move Box to the location $UnderBananas$.

Ex2 Showing Violation of Persistence

– Being able to perform the $move$ action should cause a violation in our persistency assumption about the location of Box. Let's take a look at the frame axioms to see what we can derive.

– Given the following instantiation of axiom (44),

$$Ab(at(Box, Place_b), move(Mky, Box, Uba), S0)$$

and an instantiation of axiom (43),

$$\begin{array}{l} Holds(at(Box, Place_b), S0) \wedge \\ \neg Ab(at(Box, Place_b), move(Mky, Box, UBa), S0) \rightarrow \\ \quad Holds(at(Box, Place_b), \\ \qquad result(move(Mky, Box, UBa), S0)) \end{array}$$

it should not be possible to derive its consequent,

$$\begin{array}{l} Holds(at(Box, Place_b), \\ \quad result(move(Mky, Box, UBa), S0)). \end{array}$$

– Obviously, we can not. Consequently, under the assumption that the move action is performed, we could derive the the following properties for the next state:

$$\begin{array}{l} Holds(at(Box, UBa), result(move(Mky, Box, UBa), S0)) \wedge \\ Holds(at(Mky, UBa), result(move(Mky, Box, UBa), S0)). \end{array}$$

Ex3 Proving Persistence

– Extending the axiomatization to include a new block and the additional property of being $tooHeavy$.

– Suppose, there are two boxes Box and Box' and two places $Place_b$ and $Place_{b'}$. In addition, we add the following axioms

$$Box \neq Box' \tag{50}$$

$$Place_b \neq Place_{b'} \tag{51}$$

$$Holds(At(Box', Place_{b'}), S0) \tag{52}$$

$$Holds(tooHeavy(Box'), S0) \tag{53}$$

to our axiomatization.

– **Question 1.** "Is Box' is still at $Place_{b'}$ after the $Monkey$ moves Box?":

$$\begin{aligned} &Holds(at(Box', Place_{b'}),\\ &\quad result(move(Mky, Box, UBa), S0)) \end{aligned} \tag{54}$$

– **Question 1.** "Is the Box' still $tooHeavy$ after the same action?":

$$\begin{aligned} &Holds(tooHeavy(Box'),\\ &\quad result(move(Mky, Box, UBa), S0)). \end{aligned} \tag{55}$$

– First we show that Box' can not be moved using axiom (41)

$$\forall m, b, l', s.Holds(tooHeavy(b), s) \rightarrow$$
$$Prevents(unmovable(b), move(m, b, l'), s)$$

– Instantiating (41) appropriately results in

$$Holds(tooHeavy(Box'), S0) \rightarrow$$
$$Prevents(unmovable(Box'), move(Mky, Box', UBa), S0) \quad (56)$$

– Modus ponens applied to (53) and (56) allows us to derive

$Prevents(unmovable(Box'), move(Mky, Box', UBa), S0)$.

– This blocks the use of axiom (37) to derive that Box' can be moved.

– Next, we can use the frame rule to show that Box' is still at $Place_{b'}$ after the move action. Given the following instantiation of (43)

$$\begin{aligned} & Holds(at(Box', Place_{b'}), S0) \wedge \\ & \neg Ab(at(Box', Place_{b'}), move(Mky, Box, UBa), S0) \rightarrow \\ & \quad Holds(at(Box', Place_{b'}), S0), \\ & \qquad result(move(Mky, Box, UBa), S0)) \qquad (43') \end{aligned} \tag{57}$$

(58)

– Modus ponens applied to (52) and (43′) allows us to derive

$$\begin{aligned} & \neg Ab(at(Box', Place_{b'}), move(Mky, Box, UBa), S0) \rightarrow \\ & \quad Holds(at(Box', Place_{b'}), S0), \\ & \qquad result(move(Mky, Box, UBa), S0)). \end{aligned}$$

– Since axiom (50) states that $Box \neq Box'$, axiom (44) is not applicable.

Although $\neg Ab(at(Box', Place_{b'}), move(Mky, Box, UBa), S0)$ is not clssicially derivable, suppose there was a way to conjecture the fact. Then we could derive the desired persistence statement that

$$Holds(at(Box', Place_{b'}), result(move(Mky, Box, UBa), S0)).$$

Again, it appears we need an inference rule of sorts that states that the only things that are abnormal are those that can be shown to be abnormal given our theory.

– As regards the second question, we would like to derive that

$$Holds(tooHeavy(Box'), result(move(Mky, Box, UBa), S0)).$$

– Again, we would use the frame rule and frame axioms to show that this is true. Assume the following instantiation of (43),

$$\begin{aligned} &Holds(tooHeavy(Box'), S0) \wedge \\ &\neg Ab(tooHeavy(Box'), move(Mky, Box, UBa), S0) \rightarrow \\ &\quad Holds(tooHeavy(Box'), \\ &\qquad result(move(Mky, Box, UBa), S0)) \end{aligned} \tag{43''}$$

– Applying modus ponens to axioms (53) and (43″) results in

$$\begin{aligned} &\neg Ab(tooHeavy(Box'), move(Mky, Box, UBa), S0) \rightarrow \\ &\quad Holds(tooHeavy(Box'), \\ &\qquad result(move(Mky, Box, UBa), S0)). \end{aligned}$$

– Suppose again, that we were somehow able to conjecture that

$$\neg Ab(tooHeavy(Box'), move(Mky, Box, UBa), S0).$$

– Using this conjecture we would then be able to derive the desired persistence statement,

$$\begin{aligned} &Holds(tooHeavy(Box'), \\ &\qquad result(move(Mky, Box, UBa), S0)). \end{aligned}$$