m4 - trying the MCP NMR retail - ver.4

John McCarthy Eyal Amir

December 13, 1996

In this version we build on the structure of version 3, still having actions restricted by the same set of Actions, but we allow some more nonmonotonicities, besides the persistence and qualification. The formulas changed, and the nonmonotonicities added on top of the previous version will be market with an asterisk to their left.

We try to keep, in the process of elaborating previous versions, the simplicity of previous results (such as the success with 3,3 and failure with 4,4), No elaborations are treated yet.

0.1 Improvements in this version

In this version we have the following nonmonotonicities: Qualification of actions, Persistence, Banks, available, ok, arbitrary objects in the domain are influential (see esp. the change in (14), and to its implications over (18)).

This last one is the most complicated one. It implies a possible change in our set of actions, and thus we need to add some strong justification for our previous solution to the (4,4) problem. Such a justification might be saying that the sets of missionaries and cannibals on the banks of the river, do not change over the process of applying the actions. We formalize this justification in corollary 1.3. In any case, we need to review the proofs for the (3,3) and (4,4) anyway.

Up till now we avoided the problem of having to deal with external entities to our story by not including them as restrictions to our actions, nor did we refer to them at all. We did that by specifying that only the missionaries and cannibals in our story (M and C respectively) count. We did that implicitly though.

When trying to put these arguments explicitly, we face several problems:

- We would not like to minimize the number of missionaries in the world, but rather minimize the set of missionaries in our story. Furthermore, we would like to minimize the set of missionaries that are on the banks.

- We want the possible extra missionaries or cannibals on the banks, to count in our analysis (they did not, in the previous version).

The first problem is dealt with in section 1.2.4 and the second is dealt with by amending the persistence axiom (9), and the definition of *ok* in (14).

1 Formulas

The following formulas describe the relations and objects in the basic story.

1.1 Basic Formulas (mainly action related)

1.1.1 Language

Our language will include the following object constants: M,C,$S0$, $Situations$,$Boat$,$Banks$, which will be the sets of Missionaries and Cannibals, the first Situation, the set of Situations, the Boat, and the set of Banks, respectively. We need all of these to define the sorts of objects that we are dealing with (esp. to make sure that functions and predicates act on the right sets of objects). We will define the function and relation constants with their associated axioms in 1.1.8.

In this version of the paper, we restrict ourselves to actions of the form

$$A(group, bank) = using(Boat, go(group, bank))$$

Notice that the Boat is fixed, and we therefore implicitly assume that the boat is needed, that there is exactly one, and that there is no more actions possible. We further restrict the set of actions to be

$$Actions(s) = \left\{ A(group, bank) \mid \begin{array}{l} bank \in Banks \wedge \\ group \subset inhabitants(opp(bank), s) \end{array} \right\} \tag{1}$$

All of the notions used in the above definitions are defined below.

1.1.2 can

Let A be a macro for $A(group, bank)$.

$$\begin{array}{rcl} \neg abq(A, s) & \Longrightarrow & can(A, s) \\ can(A, s) & \Longrightarrow & ok(s) \\ can(A, s) & \Longrightarrow & 0 < card(group) \\ can(A, s) & \Longrightarrow & (\forall p \in group)(location(p, s) = opp(bank)) \\ can(A, s) & \Longrightarrow & (\forall p \in group)(location(p, result(A, s)) = bank) \\ can(A, s) & \Longrightarrow & location(Boat, result(A, s)) = bank \\ can(A, s) & \Longrightarrow & card(group) \leq capacity(Boat) \\ can(A, s) & \Longrightarrow & water\text{-}crosser(Boat) \wedge available(Boat, group, s) \\ can(A, s) & \Longrightarrow & A \in Actions(s) \end{array} \tag{2}$$

1.1.3 other axioms

The important part (can) relied on the following definitions and axioms.

$$
\begin{aligned}
&(*) Lb, Rb \in Banks \wedge Lb \neq Rb \wedge opp(Lb) = Rb \wedge opp(Rb) = Lb \\
&isboat(Boat) \wedge capacity(Boat) = 2. \\
&isboat(boat) \implies water\text{-}crosser(boat) \\
&(*) available(means, group, s) \implies \\
&\qquad isboat(means) \wedge group \subset inhabitants(location(means, s), s)
\end{aligned} \tag{3}
$$

1.1.4 (*)Domain Nonmonotonicities

$$
\begin{aligned}
&\mathbf{Circ}\ Banks \\
&\neg abq_a(means, group, s) \implies available(means, group, s) \\
&\mathbf{Circ}\ abq_a\ \mathbf{var}\ available
\end{aligned} \tag{4}
$$

Using the last two policies, we can infer CWA or predicate completion for *Banks* and *available*

Lemma 1.1

$$
\begin{aligned}
&\{Lb, Rb\} = Banks \\
&available(means, group, s) \iff \\
&\qquad isboat(means) \wedge group \subset inhabitants(location(means, s), s)
\end{aligned} \tag{5}
$$

Proof For the first one, it is simply minimizing *Banks*, with nothing varied, which is simple. For the second one, notice that if Γ is our database of first order formulas, then

$$Circ[\Gamma; abq_a; available] \equiv Circ[\Delta; abq_a; available] \wedge \Gamma$$

Where

$$
\begin{aligned}
\Delta = \quad &(available(means, group, s) \implies \\
&\qquad isboat(means) \wedge group \subset inhabitants(location(means, s), s)) \wedge \\
&(can(A, s) \implies water\text{-}crosser(Boat) \wedge available(Boat, group, s)) \wedge \\
&(\neg abq_a(means, group, s) \implies available(means, group, s))
\end{aligned}
$$

Thus, we get the required equivalence, and we are done. ■

1.1.5 Goals

In the standard case of MCP, we should be able to infer

$$
\begin{aligned}
&location(grp, s) = bank \wedge location(boat, s) = bank \wedge ok(s) \wedge \\
&isboat(boat) \wedge 0 < card(grp) \leq capacity(boat) \\
&\implies can(using(boat, go(grp, opp(bank))), s).
\end{aligned} \tag{6}
$$

Notice that the $location(grp, s)$ is not well defined. We might want to use a predicate instead of a function.

This will involve some nonmonotonic reasoning. Up till now, we got

$$
\begin{aligned}
&can(using(Boat, go(group, bank)), s) \iff \\
&\qquad card(group) \leq capacity(Boat) \wedge \\
&\qquad water\text{-}crosser(Boat) \wedge available(Boat, group, s)) \wedge \\
&\qquad ok(s) \wedge \\
&\qquad 0 < card(group) \wedge \\
&\qquad A \in Actions(s)
\end{aligned}
$$

Notice that

$$
\begin{array}{l}
(\forall p \in group)(location(p, result(A, s)) = bank) \\
location(Boat, result(A, s)) = bank
\end{array}
\tag{7}
$$

are no longer prerequisites of the action (as it was the situation in the previous version of this paper), since we vary result. Of course we neglected to mention the rest of the framework of the circumscription, and the next sub-section deals with that.

Putting the last equivalence in an *easier to compute and handle* manner (translating $A \in Actions(s)$ into its equivalent extensional form, and replacing some equivalents), it is equivalent to

$$
\begin{aligned}
&can(using(Boat, go(group, bank)), s) \iff \\
&\qquad bank \in Banks \wedge \\
&\qquad 0 < card(group) \leq capacity(Boat) \wedge \\
&\qquad isboat(Boat) \wedge ok(s) \wedge \\
&\qquad location(Boat, s) = opp(bank) \wedge \\
&\qquad (\forall p \in group)(p \in missionaries \cup cannibals \wedge location(p, s) = opp(bank))
\end{aligned}
\tag{8}
$$

1.1.6 The Nonmonotonic framework

This section takes the nonmonotonic approach to formalizing action, as proposed by McCarthy in [McC86]. It works in this scenario, although it has some known weaknesses (cf [SS94])[1]. We will use the following theory (for the definition of *Atoms* see section 1.1.7):

$$
\begin{array}{l}
\textbf{(*)}FrameFunction = \{location(obj) \mid obj \in Atoms\} \\
\forall f \in FrameFunction\ \forall s \in Situations\ \forall A \in Actions(s) \\
\qquad (\neg ab(f, A, s) \implies f(s) = f(result(A, s)))
\end{array}
\tag{9}
$$

The reason for not including $can(A, s)$ in the antecedent of our inerita law will become apparent in our proof for lemma 1.2. We choose the following circumscription policy as our nonmonotonic inference tool:

[1] It is my belief that there is a theorem relating John's approach and Baker's approach. I have yet to think about it. The hypothesis says that any model that is possible in Baker's approach is also possible in John's approach.

$$\begin{array}{l}\textbf{Circ } abq \textbf{ var } Holds, ab, can \\ \textbf{Circ } ab \textbf{ var } Holds\end{array} \tag{10}$$

Notice that we did not mention *Holds* in our formalism, and we turn to it here as a shorthand for varying all the fluents.[2] The motivation behind the above policy is to make as many actions possible, even in the price of making more changes result of actions [3].

Lemma 1.2 *The Circumscription policy (10) implies*

1.

$$\begin{array}{l}\forall s \in Situations \ \forall A \in Actions(s) \ (can(A, s) \implies \\ \quad (\forall p \in group)(location(p, result(A, s)) = bank) \wedge \\ \quad location(Boat, result(A, s)) = bank \wedge \\ \quad (\forall p \in (Atoms \setminus group \setminus \{Boat\}) \\ \qquad location(p, result(A, s)) = location(p, s)))\end{array}$$

2.

$$\begin{array}{l}\forall s \in Situations \ \forall A \in Actions(s) \\ ((\exists p \in Atoms \ (location(p, result(A, s)) \neq location(p, s))) \implies \\ \qquad (can(A, s))\end{array}$$

Proof For the first part, notice that the first two rows of the succedent follow directly from (2). The third row is proved true given $can(A, s)$, using the persistence axiom (9). Assume that $\exists p \notin group, p \neq Boat$ s.t., $location(p, result(A, s)) \neq location(p, s)$. For this p we have no axiom refering to its location, with regard to the specific application of A to s (since it is neither the *Boat* nor a member of *group*). Thus we can vary $location(p, result(A, s)) = location(p, s)$, and get a smaller ab, contradicting the fact that ab is minimal.

For the second part, let $s \in Situations, A \in Actions(s)$, and assume that $\neg can(A, s)$, but the location of $p \in Atoms$ changed after applying A to s. Notice that since $\neg can(A, s)$ holds, all the sentences having $can(A, s)$ as an antecedent, are automatically tautologies. Thus, when we change $location(p, result(A, s))$ to be $location(p, s)$, the validity of (2) is preserved, and since we are allowed to vary $Holds$, we may change the values of $available$ (5) and ok (18), both relate to the location of some objects, and still preserve validity. Thus we can allow ab smaller than the one we got with $location(p, result(A, s)) \neq location(p, s)$ (and notice that the corressponding value for abq does not change). Contradiction. ∎

The last lemma related can to its intended meaning.

[2] What we would really like to say is that we vary $Holds$ in the resulting situation... or do we?

[3] actually the order does not matter in this case, since we can put the circumscirption policy to force circumscribing ab before abq, as long as we keep can varying only for the circumscription of abq. We do prefer to do that the way we do, since it seems to make more sense, and we hope to generalize it in the future framework

Corollary 1.3 *The domain of the story stays the same. More formally*

$$\begin{array}{l} \forall s \in Situations\ \forall A \in Actions(s) \\ \quad \{x \in Atoms \mid location(x, s) \in Banks\} = \\ \qquad \{x \in Atoms \mid location(x, result(A, s)) \in Banks\} \end{array} \tag{11}$$

PROOF If no $x \in Atoms$ changed its location, then it is obvious. Otherwise, according to lemma 1.2, $can(A, s)$ holds, and according to the same lemma, the only things that changed their places are those mentioned in the action A. Since $A \in Actions(s)$, and according to (1), each one of the atoms that possibly changed its place, has its location on one of the banks in $Banks$. ∎

1.1.7 (*)Atoms, and why do we need them

We have to make sure that no missionary suddenly appears on one of the banks. For this, we need the persistence to include all kinds of objects that were not mentioned in the previous version of this paper. On the other hand, we can not allow Persistence to include *all* the objects, as there are some objects for which $location$ is defined according to some constraints regarding some other objects locations. We therefore need to reuse the idea of $FrameFluents$ in the domain of specifying which objects are more *atomic*.

$$\begin{array}{l} Boat \in Atoms \\ missionaries \subset Atoms \\ cannibals \subset Atoms \end{array}$$

We further use nonmonotonicity to apply CWA to $Atoms$.

$$\textbf{Circ}\ Atoms$$

And the result follows easily

$$Atoms = \{Boat\} \cup missionaries \cup cannibals \tag{12}$$

1.1.8 Axioms regarding the functions and relations

$$\forall s \in Situations\ \forall a \in Actions(s)\ (result(a, s) \in Situations) \tag{13}$$

1.2 Story-specific details

The following section deals with *domain axioms* and the *initial situation*. We further define some notions previously used, which relate to the specific problem at hand.

1.2.1 Story-specific definitions and axioms

The only formulas we need for our previous definitions/formulas are

$$\begin{array}{l}
(*) inhabitants(bank, s) = \{x \in missionaries \cup cannibals \mid location(x, s) = bank\} \\
(*) m(s, bank) = card(\{m \in missionaries | location(m, s) = bank\}) \\
(*) c(s, bank) = card(\{c \in cannibals | location(c, s) = bank\}) \\
ok(s) \implies \neg(0 < m(s, Lb) < c(s, Lb)) \wedge \neg(0 < m(s, Rb) < c(s, Rb)) \wedge s \in Situations
\end{array} \tag{14}$$

The rest will be useful for the proofs (section 2):

$$\begin{array}{l}
m(group) \equiv card(group) = 1 \wedge group \subset M \\
mm(group) \equiv card(group) = 2 \wedge group \subset M \\
c(group) \equiv card(group) = 1 \wedge group \subset C \\
cc(group) \equiv card(group) = 2 \wedge group \subset C \\
mc(group) \equiv card(group) = 2 \wedge (\exists x)(x \in group \cap M) \\
\qquad\qquad\qquad\qquad\qquad \wedge (\exists x)(x \in group \cap C)
\end{array} \tag{15}$$

maximized,

1.2.2 Domain axioms

$$\begin{array}{l}
(*) M \subset missionaries \\
(*) C \subset cannibals \\
card(M) = 3 \\
card(C) = 3
\end{array} \tag{16}$$

1.2.3 The initial situation $S0$

$$\begin{array}{l}
S0 \in Situations \\
m(S0, Lb) = M \\
c(S0, Lb) = C \\
location(Boat, S0) = Lb
\end{array} \tag{17}$$

1.2.4 (*)Story completion

First, let us complete ok. We need that in order to say that for a situation to be ok it is sufficient that its mentioned properties are satisfied.

$$\begin{array}{c}
\neg abq_{ok}(s) \implies ok(s) \\
\textbf{Circ } abq_{ok} \textbf{ var } ok
\end{array}$$

From which we get predicate completion for ok

Lemma 1.4

$$ok(s) \equiv \neg(0 < m(s, Lb) < c(s, Lb)) \wedge \neg(0 < m(s, Rb) < c(s, Rb)) \wedge s \in Situations \tag{18}$$

PROOF The only place where ok shows negatively is (14), which is the formula we would like to complete. ∎

We would also like to say that the only things located on our river are those that are mentioned. Notice that this solves the problem of the *scope*, i.e., the objects that we deal with are those that are relevant/important to our story, but we deal with that only implicitly.

$$\textbf{Circ}\ \{x \mid location(x, S0) \in Banks\}\ \textbf{var}\ \lambda x.location(x, S0) \tag{19}$$

Lemma 1.5 [4] *The circumscription policy (19) implies*

$$\forall x \in Atoms\ (location(x, S0) \in Banks \iff x \in (M \cup C \cup \{Boat\})) \tag{20}$$

PROOF The backward direction follows simply from (17). For the forward direction, notice that since we apply the circumscription implied by (19) before applying persistence (9), the only objects that are forced to be on the banks are mentioned in (17), where the rest of the theory only implies some constraints between the location of an object in one situation, and its location in a resulting situation.[5] Therefore (19) implies predicate completion for $\{x \mid location(x, S0) \in Banks\}$, and we are done. ∎

1.3 Putting order into our nonmonotonic inference

The following is sorted according to the order of circumscription application:

$$\begin{array}{l}
\textbf{Circ}\ Atoms \\
\textbf{Circ}\ Banks \\
\textbf{Circ}\ \{x \mid location(x, S0) \in Banks\}\ \textbf{var}\ \lambda x.location(x, S0) \\
\textbf{Circ}\ abq_{ok}\ \textbf{var}\ ok \\
\textbf{Circ}\ abq_{a}\ \textbf{var}\ available \\
\textbf{Circ}\ abq\ \textbf{var}\ Holds, ab, can \\
\textbf{Circ}\ ab\ \textbf{var}\ Holds
\end{array} \tag{21}$$

2 Proofs about the resulting formalism

2.1 A Proof that (3,3) works

Let us first see what happens when we want to move the first two cannibals. Let CC be a group of two cannibals in $S0$ (such that $cc(CC)$ holds).

[4] Notice that since we modify the $location(x, S0)$ of some objects x during that circumscription, we must do that circumscription *before* we do the others.

[5] Very important. This argument forces us to use Reiter's view of Situations as including their history.

2.1.1 The first step - $can(using(Boat, go(CC, Rb), S0))$

Using (8):

$$
\begin{array}{lll}
Rb \in Banks & & (3) \\
0 < card(CC) \leq capacity(Boat) & & (3)+(15) \\
isboat(Boat) & & (3) \\
\textbf{(*)}ok(S0) & & (18)+(16)+(17)+(20) \\
location(Boat, S0) = opp(Rb) & & (3)+(17) \\
(\forall p \in CC)(p \in M \cup C \wedge & location(p, S0) = opp(Rb)) & (3)+(15)+(17)
\end{array} \quad (22)
$$

We are left to prove the existence of CC, but for $S0$ it follows from (16) and (15).

For the rest of the stages it will be more difficult and might require some more notations. One is refered to the PVS proof of the MCP (ver. 2.0 and up). It uses set theory and besides the problems involving the applications of extensionality and counting, it has the problem of "selecting" memebers from a set. We ignore these notational issues here, noticing that they are solvable, in the expense of complicated ad-hoc notations, that do not contribute to the understanding of the problem and its analysis.

2.1.2 The generic step

Let us carry one more step. Let a_1 be the action performed in the previous section, and let

$$S1 = result(a_1, S0)$$

According to lemma 1.2,

$$
\begin{array}{l}
\forall p \in CC\ (location(p, S1) = Rb) \\
location(Boat, S1) = Rb \\
\forall p \in (C \cup M \setminus CC)\ (location(p, S1) = Lb)
\end{array} \tag{23}
$$

Thus, when we need one cannibal $C1$ to go back, the action is enabled as before. The only changes in the justifications are:

$$
\begin{array}{ll}
ok(S1) & (18)+(16)+(23)+(13)+(11) \\
location(Boat, S1) = opp(Lb) & (3)+(23) \\
(\forall p \in C1)(p \in M \cup C \wedge & \\
\qquad location(p, S1) = opp(Lb)) & (3)+(15)+(23)
\end{array} \tag{24}
$$

The existence of the needed group of people at each stage is also guaranteed by (23) above.

2.2 Proving that (4,4) has no solution

Let us first notice that what we are looking for is a sequence of actions that will result in transfering 4 missionaries and 4 cannibals from the Lb to the Rb.

For this version, we will change 16 to say

$$\begin{aligned} card(M) = 4 \\ card(C) = 4 \end{aligned} \tag{25}$$

Now, let us assume that such a sequence Seq of length n exists. Notice, that because of lemma 1.2, any change in the location of any missionary or cannibal (or Boat) is governed by a *possible* action. Thus we have

$$1 \leq \forall i \leq n \ (can(a_i, s_{i-1}))$$

for $s_i = result(a_i, s_{i-1})$, and a_i the i-th element in Seq.

Notice that there are at most 5 possible actions in every situation that is accessible (by means of chains of actions), from $S0$, *given that we do not distinguish among the missionaries or the cannibals that are located on the bank of the river.* We will justify this approach later in this section.

Lemma 2.1 *If $s \in Situations$ is accessible by a chain of actions application (for actions in $Actions(s)$) from $S0$, then there are at most 5 possible actions in that situation, (*given that we do not distinguish among the missionaries or the cannibals that are located on the bank of the river*), and they are the obvious ones.*

Proof Notice that some of can's preconditions are $A \in Actions(s)$, and $0 < card(group) \leq 2$, by (8).

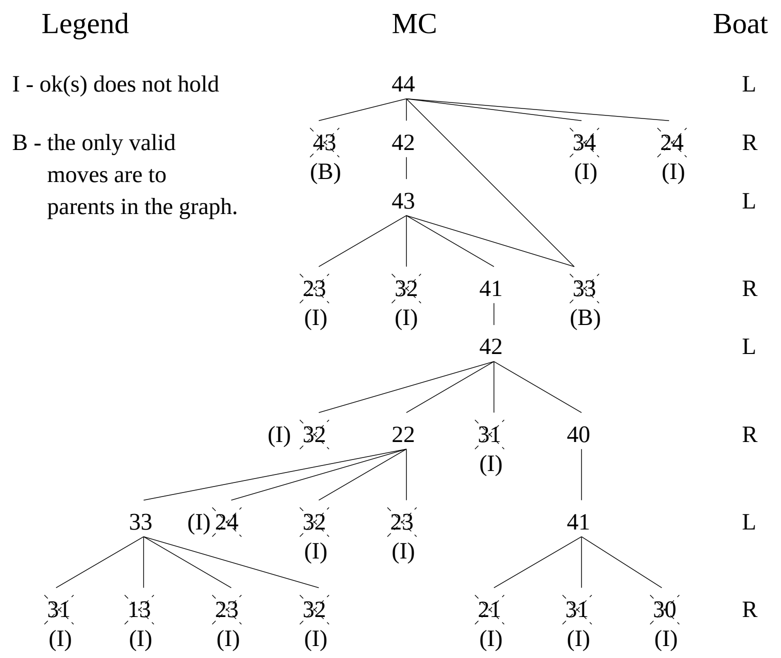
We will prove that by induction on the length of the sequence of actions. For 0, notice that there are 5 different sets of the missionaries and cannibals that are located on the banks of the river (again, taking two sets of different persons, with same characteristics (missionaries/cannibals) to be equivalent), with size ≤ 2. This is guaranteed by (17) and (16). Therefore the induction base holds.

The induction step follows simply using corollary 1.3. ■

The following graph is a straightforward *unformal* proof that there is no such Sequence. The nodes are possible states represented in terms of the number of missionaries and cannibals on the left bank, and the location of the boat (Right or Left banks). Each load that is crossed out, is labeled with the reason for the crossing. There are some branches that are not mentioned, since they are obviously illegal.

Note that this is an undirected graph (although it is presented in a top-down manner). It formalizes the accessibility relation between situations. Since the actions are always retractable by the the appropriate conjugate action, there is always an action that will return us to the *parent* situation (parent node in the graph view).

Missionaries and Cannibals (4,4)



In order to make that proof more formal we need to follow several steps. First, make sure that the possible actions and only those, are shown. This is pretty straightforward, as there are at most 5 possible actions possible in every situation (as argued above), and the preconditions that are situation/action dependent are on the cardinality of the group, on the location of the group (and the boat), and on the ok-ness of the situation.

Second we should make sure that the amount of people on the left bank, and the location of the boat, are the only issues that matter, in the current formalization. But as we just argued, the possible actions refer exactly to these properties of the selected group and the boat. The selection of specific members of the appropriate sets in each bank has no influence on anything, and in particular on the outcome location of the group (we picked the members of the set *from that bank*), or the set's cardinality.

Also, the *ok*-ness of the situation is dependent only on the cardinality of the appropriate sets on each bank, and the situation-ness of the situation (i.e., that s is actually a situation). Since the set of missionaries and the set of cannibals is constant (they are not situation dependent - (25)), and location is a complete function (on missionaries, cannibals and the boat), the cardinalities of the sets of missionaries and cannibals in the right bank is directly calculatable from the cardinalities of the relative sets in the left bank. Since the fact that s is a situation follows from the application of the previous action to a situation and (13), we conclude that *the amount of missionaries and cannibals on each* *bank in the resulting situation, is dependent only on the relative amounts in the prior situation, the amounts of people from each set in the boat (the group that actually* goes*), and the location of the boat* (all of that - given (1.2)).

After these formal notes, we can see that the above graph is an exact proof of the inaccessibility of (44) and (00), and thus an inaccessibility of the relative situations[6].

References

[McC86] John McCarthy. Applications of circumscription to formalizing common sense knowledge. *Artificial Intelligence*, 28:89–116, 1986. Reprinted in [?].

[SS94] Erik Sandewall and Yoav Shoham. Nonmonotonic temporal reasoning. In *Handbook of Logic in Artificial Intelligence and Logic Programming, Volume 4: Epistemic and Temporal Reasoning.* Oxford University Press, 1994.

This is /u/eyal/ongo/mcp/jmc/m4-nmr4.tex.

[6] A somewhat simpler and more intuitive proof than the graph might go along the lines of showing that every situation in which the two quantities (missionaries and cannibals on the left bank) are between 1 and 3, is illegal (not *ok*) (and thus there is no action possible in these situations). The graph seems to be more simple to comprehend, though.