THE μ-CALCULUS AS AN ASSERTION LANGUAGE FOR FAIRNESS ARGUMENTS

F.A.Stomp, W.P.de Roever, and R.T.Gerth

RUU-CS-84-12
November 1984

Rijksuniversiteit Utrecht

Vakgroep informatica

Budapestlaan 6 3584 CD Utrecht
Corr. adres: Postbus 80.012 3508 TA Utrecht
Telefoon 030-53 1454
The Netherlands

THE μ-CALCULUS AS AN ASSERTION LANGUAGE FOR FAIRNESS ARGUMENTS

F.A.Stomp, W.P.de Roever, and R.T.Gerth

Technical Report RUU-CS-84-12
November 1984

Department of Computer Science
University of Utrecht
P.O.Box 80.012, 3508 TA Utrecht
the Netherlands

THE μ-CALCULUS AS AN ASSERTION LANGUAGE FOR FAIRNESS ARGUMENTS

by

F.A. Stomp -University of Utrecht.)*
W.P. de Roever-University of Nijmegen/
University of Utrecht.
R.T. Gerth -University of Utrecht.

-November 1984-

Abstract: Various principles of proof have been proposed to reason about fairness ([2],[5],[7],[12]). This paper addresses - for the first time - the question in what formalism such fairness-arguments can be couched. To wit: we prove that Park's monotone first-order μ-calculus ([6],[12]), augmented with constants for all recursive ordinals can serve as an assertion-language for proving fair termination of do-loops. In particular, the weakest precondition for fair termination of a loop w.r.t. some postcondition is definable.
The relevance of this result to proving eventualities in Manna and Pnueli's temporal logic formalism ([8]) is discussed.

)* Presently employed at the University of Nymegen.

1.4 Next, nearing the focus of this paper, the interaction between fairness and the interleaving model must be examined.
How does one deduce properties in the resulting model?
The properties of interest always contain eventualities which are enforced by the assumption of fairness. Pure invariances, i.e., properties which are invariant during execution, are not influenced by postulating fairness as extra requirement, and can be derived using more traditional methods.

1.5 The state of art offers the following picture:
To establish that for a concurrent program ψ eventually holds, i.e., $\Diamond\psi$ holds, using the eventuality operator $\Diamond$ from temporal logic, where ψ is a state formula, i.e., a direct property of the program state not requiring temporal operators s.a. $\Diamond$ anymore for its expression, the following strategy is taken:

(1) Amongst the concurrent processes an (amongst others state dependent) distinction is made between those processes -in Manna and Pnueli's ([8]) terminology dubbed helpful processes- whose execution brings satisfaction of ψ always nearer, and those processes that do not do so, i.e., whose execution possibly does not bring satisfaction of ψ any nearer, called steady (or unhelpful) processes.

(2) It must be proved that systematically avoiding execution of any helpful process either leads to an interleaving of steady processes which does not satisfy fairness, i.e., is unfair, since infinitely often a helpful process is enabled but not taken, or, due to some nondeterministic choice in a steady process or the interleaving, does bring satisfaction of ψ eventually nearer or even eventually establishes ψ.

Essential is here that upon closer inspection part (2) above requires application of the same strategy to a syntactically simpler program: just remove the helpful processes from the original program and prove that eventually one of the following holds: ψ, getting nearer to ψ or a, helpful process is enabled.

1.6 A technical formulation of this strategy requires the introduction of well-founded sets, and looks as follows ([8]):

The Well-founded Liveness Principle-WELL

Let $\mathcal{W} = (A, \leq)$ be a well-founded ordered structure. Let $\phi(\alpha)$ be a parametrized state formula (intuitively expressing how far establishing ψ is). Let $h: A \to \{1,..,k\}$ be a helpfulness function identifying for each $\alpha \in A$ the helpful process $P_{h(\alpha)}$ for states satisfying $\phi(\alpha)$.

(A) $\vdash$ P leads from $\phi(\alpha)$ to $[\psi \vee (\exists \beta \leq \alpha\ \phi(\beta))]$

(B) $\vdash P_{h(\alpha)}$ leads from $\phi(\alpha)$ to $[\psi \vee (\exists \beta < \alpha\ \phi(\beta))]$

(C) $\vdash \phi(\alpha) \supset \Diamond[\psi \vee (\exists \beta < \alpha\ \phi(\beta)) \vee \text{Enabled}(P_{h(\alpha)})]$

$\vdash (\exists \alpha\ \phi(\alpha)) \supset \Diamond\psi$

Here P leads from ϕ to ϕ' means: P_i leads from ϕ to ϕ', for i=1,...,k.

And P_i leads from ϕ to ϕ' means that every state transition in P_i establishes ϕ' afterwards provided ϕ is satisfied before (;here ϕ and ϕ' are state formulae).

The soundness proof of this rule requires induction over well-founded sets.
Conversely, given the fact that $\Diamond\psi$ is valid, (naive) set theory is used to argue the existence of the required auxiliary quantities (the well-founded ordered structure $\mathfrak{M}$, the ranking predicate $\phi(\alpha)$, and the helpfulness function h) which satisfy clauses (A), (B), (C), so that for each such ψ WELL can always be applied. This proves that WELL is semantically complete.

Manna and Pnueli ([8]) even prove that, for certain classes of formulae, their temporal logic formalism is relative complete.
Relative means here: all valid temporal formulae with the given domain interpretation are taken as axioms. Typically, their proof shows that issues concerning programs and executions can be reduced via their rules (from which the one above is derived and deals with eventualities) to state assertions concerning the given program, the so-called state properties.

Now we are ready to ask the one question this paper is about:

How do these results help us if we are sure that $\Diamond\psi$ holds and want to apply the rule above to verify $\Diamond\psi$?

The answer is: not much.

Questions such as:

- How to obtain the appropriate well-founded ordered structure $\mathfrak{M}$?
- How does one express, and reason about, the helpfulness-function h and the ranking predicate $\phi(\alpha)$?
- In general, which assertion language should be used to establish hypotheses (A), (B), (C), of WELL?

are not answered by the above results, since the state properties are not formalized.

1.7 The present paper suggest a direction to answer these questions, by concentrating on these problems as they occur when proving termination of do-loops under the above fairness assumptions, i.e., fair termination of do-loops.
That this does not lead to oversimplification follows from the fact that the same auxiliary quantities, with comparable objectives, occur in the rule whose expression and use we shall investigate ([5]):

The Well-founded Liveness Principle for loops-Orna's rule

Let $\mathfrak{M} = (A,\leq)$ be a well-founded structure.
Let $\pi: A \to (\text{States} \to \{\underline{\text{true}}, \underline{\text{false}}\})$ be a predicate, and q be a state predicate.
Let for $w \in A$, with w not minimal (denoted by $0<w$), be given pairwise disjoint sets D_w and S_w, such that $D_w \neq \emptyset$ and $D_w \cup St_w = \{1,..,n\}$.

(a) $\vdash [\pi(w) \wedge w>0 \wedge b_j] S_j [\exists v<w\ \pi(v)]$, for all $j \in D_w$

(b) $\vdash [\pi(w) \wedge w>0 \wedge b_j] S_j [\exists v \leq w\ \pi(v)]$, for all $j \in St_w$

(c) $\vdash [\pi(w) \wedge w>0]\, fair(*[\ \Box_{i \in St_w} b_i \wedge \bigwedge_{j \in D_w} \neg b_j \to S_i\ ])[\underline{\text{true}}]$

(d) $\vdash r \supset (\exists v\ \pi(v))$

$\vdash (\pi(w) \wedge w>0) \supset \bigvee_{i=1}^{n} b_i$

$\vdash \pi(0) \supset ((\bigwedge_{i=1}^{n} \neg b_i) \wedge q)$

$\vdash [r]\, fair(*[\ \Box_{i=1}^{n} b_i \to S_i\ ])[q]$

Here for a fair do-loop $fair(*[\ \Box_{i=1}^{n} c_i \to T_i\ ])$ only fair execution sequences are generated, i.e., finite ones or fair infinite ones, but no unfair infinite ones;

$[p]S[q]$ holds iff for all ξ:
if input state ξ satisfies p then every (generated) computation sequence of S in ξ terminates and its output satisfies q;

hence $[p]\, fair(*[\ \Box_{i=1}^{n} c_i \to T_i\ ])[q]$ expresses that every fair computation sequence of $*[\ \Box_{i=1}^{n} c_i \to T_i\ ]$ which starts in p terminates in q.

Note, when comparing Orna's rule with WELL, that the commands S_j act as state transitions.

Since in Orna's rule the assignment $w \to (D_w, St_w)$ for $w>0$ merely generalizes WELL's notion of helpfulness function, the same kind of auxiliary quantities are required to apply both rules.

1.8 This paper proves that to express and reason about $\mathfrak{M}$, ϕ, and the assignment $w \mapsto (D_w, St_w)$ for $w>0$ and $w \in A$, a slight extension is required of the formalism used to prove termination of recursive procedures, Park's μ-calculus ([6],[12]).

Finally we note that, historically, two rules have been formulated to prove termination of (nondeterministic) programs:
Orna's rule ([5]) and the LPS-rule([7]).
Both these rules model, each in their own way, a specific intuition related to the notion of eventuality implied by fairness assumptions.
For fairly terminating loops they have been proved to be equivalent (cf. ([5])), but the LPS-rule also applies to proving fair termination of concurrent processes.

This article is organized as follows:
You are still reading chapter 1, containing the motivation for this paper; chapter 2 specifies the programming language used in this paper. Chapter 3 discusses termination under fairness assumptions. In chapter 4 the proofsystem and in chapter 5 the assertion-language (i.e., the monotone μ-calculus) are dealt with. A term in the assertion-language, which expresses fair termination of a repetition is constructed in chapter 6. Completeness and soundness of Orna's rule are proven in chapters 7 and 8. Finally section 9 contains the conclusion.

Chapter 2
THE LANGUAGE OF GUARDED COMMANDS

2.1 In this chapter we describe the programming language used throughout this paper.
In section (2.2) its syntax and in section (2.3) its (relational) semantics is given.

A first-order structure $\mathfrak{M}$ consists of (i) a non empty domain (set) $|\mathfrak{M}|$, (ii) a set of n-ary function symbols and a set of n-ary predicate symbols ($n \geq$), such that for each n-ary function symbol (respectively predicate symbol) there corresponds a n-ary function (respectively predicate) over $|\mathfrak{M}|$, and (iii) a set of constants, corresponding to elements of $|\mathfrak{M}|$. (We assume the equality symbol "=" to be present as a binary predicate symbol, corresponding to the standard equality over $|\mathfrak{M}|$.)

2.2 SYNTAX

This paper is concerned with fair termination of repetitions. Hence for repetitions S, the notation fair(S) is introduced.
Let $\mathfrak{M}$ be some first-order structure. The language of guarded commands over $\mathfrak{M}$, LGC($\mathfrak{M}$), is defined by the following BNF-productions:
(braces enclose a repeated item, that may occur zero or more times)

```
<command>     ::= <assignment> | <repetition> | <composition> |
                  <fair loop>
<assignment>  ::= <variable>:=<expression>
<composition> ::= <command>;<command>
<selection>   ::= [{□<direction>}]
<direction>   ::= <guard>→<command>
<repetition>  ::= *<selection>
<fair loop>   ::= fair(<repetition>)
<expression>  ::= "term over (the signature) 𝔐"
<guard>       ::= "quantifier-free formula over 𝔐"
```

We identify *[ ] with x:=x (skip).
In the remainder of this paper, we shall often abbreviate

$$*[b_1 \rightarrow S_1 \square \ldots \square b_n \rightarrow S_n] \text{ to } *[\mathop{\square}_{i=1}^{n} b_i \rightarrow S_i].$$

2.3 SEMANTICS

A state is a function from the collection of all variables to the domain of interpretation: ξ, ξ', ξ_i etc. are used to denote states.

$\xi(e)$ denotes the value of expression e in state ξ.
If a guard b_i evaluates to true in state ξ (i.e., $\xi(b_i)$ holds) we say that b_i is enabled in state ξ. Otherwise b_i is disabled in ξ.

For a variable x and an expression e, $\xi\{e/x\}$ is defined as usual:
$\xi\{e/x\}(x)=\xi(e)$
$\xi\{e/x\}(y)=\xi(y)$ if $x \neq y$

Since programs depend on only finitely many variables, states can be described as functions with finite domains.
We now associate with each program S its relational semantics $R_S \subseteq |\mathfrak{M}|x|\mathfrak{M}|$, where $|\mathfrak{M}|$ denotes the domain of interpretation of $\mathfrak{M}$.
Due to nondeterminism there may be more than one output-state and even infinitely many. If S nowhere terminates there will be no output-state.

$S \equiv x:=e : R_S=\{(\xi,\xi\{e/x\}) \mid \xi \text{ a state}\}$

$S \equiv S_1;S_2 : R_S=R_{S_1} \circ R_{S_2}$, where $\circ$ denotes composition of relations.

$S \equiv *[\mathop{\square}_{i=1}^{n} b_i \rightarrow S_i]$ $(n \geq 1)$: Let b denote the formula $\neg(b_1 \vee \ldots \vee b_n)$, and

$R_S = \bigcup_{i=1}^{n} R_{b_i} \circ R_{S_i}$, where $R_{b_i}=\{(\xi,\xi) \mid \xi \text{ a state such that } \xi(b_i) \text{ holds}\}$.

Then $R_S=(\bigcup_{i=1}^{n} R_S^i)\circ R_b$, where R_S^i denotes the i-fold composition of the relation R_S with itself.

$S \equiv \text{fair}(*[\mathop{\square}_{i=1}^{n} b_i \rightarrow S_i])$ $(n \geq 1)$: its semantics is given in chapter 3.

In the sequel we will concern ourselves exclusively with repetition statements. From now on, the term program will in general refer to a repetition.

Chapter 3
TERMINATION UNDER FAIRNESS ASSUMPTIONS

3.1 An execution sequence for a repetition $S \equiv *[\ \Box_{i=1}^{n}\ b_i \rightarrow S_i\ ]$, $(n \geq 1)$ is a maximal sequence $\xi_0, \xi_1, \xi_2, \ldots$ of states, such that $\xi_j R_i \xi_{j+1}$, where $j \geq 0$, $1 \leq i \leq n$ and R_i is the relation associated with $b_i; S_i$. (The sequence is considered to be maximal if it cannot be extended , i.e., it is either infinite or the sequence is finite and ends with a state ξ_k such that $\xi_k(\ \bigwedge_{i=1}^{n} \neg b_i\ )$ holds.

Termination of a (nondeterministic) program, S, is straightforwardly defined as the absence of an infinite execution sequence of S. This is, however, a very strong requirement.
Consider, e.g., Dijkstra's random number generator ([4]):
$S_0 \equiv$ *[b→x:=x+1 □ b→b:=<u>false</u>].

S_0 need not necessarily terminate if started in a state ξ such that $\xi(b)$=true, because its execution may be governed by an extremely one-sided scheduler that consistently refuses to execute the second direction of S_0 in any iteration.
Consequently, various constraints on schedulers have been proposed, which prohibit schedulers to neglect the execution of directions under certain circumstances. Termination of a program is considered relative to a set of schedulers thus constrained. We now present two important constraints or fairness-assumptions, that have been proposed ([5],[7]): <u>fairness</u>, and <u>impartiality</u>. Observe that, while the above program, S_0, admits infinite computations, none of them is fair ; i.e. S_0 terminates fairly.

3.2 <u>DEFINITION</u>
(1) An execution sequence of a program S is fair,
if it is finite or if it is infinite and every direction, which is infinitely enabled in this sequence, is chosen infinitely often.
(2) A program S terminates fairly if it admits no infinite fair execution sequences (i.e., fair(S) terminates).

In the sequel, we also need the notion of <u>impartiality</u>, that ignores the enabledness and disabledness of directions.

3.3 <u>DEFINITION</u>
(1) An execution sequence of a program is impartial,
if it is finite or it is infinite and every direction occurs infinitely often in the sequence.
(2) A program terminates impartially if it admits no infinite impartial execution sequences.

The program $S_1 \equiv *[x=0 \rightarrow x:=1 \ \square \ x=1 \rightarrow x:=x]$ does admit infinite fair computations, but no impartial ones.

Other examples of impartially, and fairly terminating programs can be found in e.g. [5]. (Some authors use a different terminology!)

3.4 The relation between the two fairness assumptions is as follows :

for each program S
(i) S terminates nondeterministically $\Rightarrow$
S terminates fairly
(ii) S terminates fairly $\Rightarrow$
S terminates impartially

The examples above show that all implications are proper.

Let $S \equiv *[\ \square_{i=1}^{n} b_i \rightarrow S_i\ ]$ $(n \geq 1)$. We now give the semantics $R_{fair(S)}$ associated with fair(S). For each execution sequence of $*[\ \square_{i=1}^{n} b_i \rightarrow S_i\ ]$ in which b_i is infinitely often enabled, S_i is executed infinitely many times $(i=1,\ldots,n)$.

3.5 Remark: definition (3.2) refers to so-called top-level fairness, according to which the following program need not terminate fairly (see e.g. [2]):

```
S≡*[b1→*[b2→skip
        □b2→b2:=false
        □b2→b1:=false
        ]
   □b1→skip
   ]
```

Fairness, as defined here, only constrains the choice of the directions guarded by b_i. It does not specify anything about choices inside the S_i $(i=1,\ldots,n)$. The problem of all-level fairness is not considered in this paper.

Chapter 4
THE PROOFSYSTEM

We use a Hoare-like proofsystem. Let S be any program.
By [p]S[q] we mean that for all states ξ satisfying p, the execution sequences of S, starting in ξ are finite. Moreover every final state of such a sequence satisfies q.

The axioms and rules are as follows:

(1) assignment

$$[p\{e/x\}]x:=e[p]$$

(2) composition

$$\frac{[p]S_1[q] \quad , \quad [q]S_2[r]}{[p]S_1;S_2[r]}$$

(3) consequence

$$\frac{p \supset p_1, \quad [p_1]S[q_1] \quad , q_1 \supset q}{[p]S[q]}$$

(4) Orna's rule (see section (1.7).

Note that only fair repetitions are considered. However, Orna's rule can also be applied to ordinary terminating do-loops (take the sets St_w to be empty). We then obtain Harel's rule for terminating loops ([15]).

Chapter 5
THE ASSERTION-LANGUAGE L

5.1 Our assertion-language is based on Park's monotone μ-calculus, ([6],[12]), which is appropriate both to prove e.g. termination of recursive parameterless procedures, see e.g. [3],[6], and to express the auxiliary quantities associated with those proofs.

5.2 This calculus is based on Knaster-Tarski's theorem ([14]):
let $(A,\sqsubseteq)$ be a complete lattice and $F:A\to A$ a monotonic function (, in fact a cpo suffices). Then F has a least fixedpoint, denoted by $\mu a.[F(a)]$, meaning that

(i) $F(\mu a.[F(a)])=\mu a.[F(a)]$;
i.e., $\mu a.[F(a)]$ is a fixedpoint of F.
(ii) if there exists some $b\in A$ such that $F(b)=b$,
then $\mu a.[F(a)]\sqsubseteq b$;
i.e., $\mu a.[F(a)]$ is the <u>least</u> fixedpoint of F.

As the partial ordering $\sqsubseteq$ is anti-symmetric, $\mu a.[F(a)]$ is unique.
(property (i) is referred to as the fixedpoint property.)

There are several ways to regard least fixedpoints. Using the notation as above, firstly $\mu a.[F(a)]=\prod\{x\in A\mid F(x)=x\}=\prod\{x\in A\mid F(x)\sqsubseteq x\}$, where $\prod$ denotes the infimum. A proof of this can be found in e.g. [3]. Secondly, the least fixedpoint can be obtained by iterating F into the transfinite ordinals. Define for each ordinal λ:

$$F^0(x)=x,$$

$$F^\lambda(x)=F(\bigsqcup_{\beta<\lambda} F^\beta(x)),\ \text{if } \lambda\neq 0.$$

Here $\bigsqcup$ denotes the supremum.

Let $\bot_A$ denotes A's least element, which exists since A is a cpo.
Then $\mu a.[F(a)]=F^\alpha(\bot_A)$ for some ordinal α. (For a proof see e.g. [9].)

Clearly, if $\mu a.[F(a)]=F^\alpha(\bot_A)$ then for all $\beta\geq\alpha$ $\mu a.[F(a)]=F^\beta(\bot_A)$.

Next, we introduce some fixedpoint definitions.

Let R be a relation and p a predicate.
Define $R\to p$ by $(R\to p)(x)$ iff $\forall x'[(x,x')\in R\Rightarrow p(x')]$
and its dual $R\circ p$ by $\neg(R\to\neg p)$. So $(R\circ p)(x)$ holds iff $\exists x'[(x,x')\in R\wedge p(x')]$.
Note that $R\to$<u>true</u> always holds.

Since the collection of predicates ordered by $p\sqsubseteq q$ iff $p\supset q$ forms a complete lattice with <u>false</u> as least element, and $R\to p$ (as well as $R\circ p$) is monotonic in p, $\mu p.[R\to p]$ exists.

We claim that $\mu p.[R \to p]$ describes the domain of well-foundedness of R; i.e., $\mu p.[R \to p](x)$ holds for those x such that there exists no infinite sequence $x_0, x_1, x_2, \ldots$ with $x = x_0$ and $(x_i, x_{i+1}) \in R$ $(i \geq 0)$.
$\mu p.[R \to p] = \tau^{\alpha}(\underline{\text{false}})$ for some ordinal α, where $\tau(p) = R \to p$.

Using induction on β, we prove that for all $\beta \leq \alpha$

$$\tau^{\beta}(\underline{\text{false}})(x) \Rightarrow \text{ there is no infinite sequence } x_0, x_1, x_2, \ldots \text{ with } x = x_0 \text{ and } (x_i, x_{i+1}) \in R \ (i \geq 0)$$

holds.

Inductionstep: $\beta = 0$: trivial.
Inductionhypothesis: suppose that the implication holds for all $\lambda < \beta$.

For $\beta \neq 0$:
$$\tau^{\beta}(\underline{\text{false}})(x) \Leftrightarrow (R \to \bigsqcup_{\lambda<\beta} \tau^{\lambda}(\underline{\text{false}}))(x)$$
$$\Leftrightarrow \forall x'[(x,x') \in R \Rightarrow \bigsqcup_{\lambda<\beta} \tau^{\lambda}(\underline{\text{false}})(x')].$$

So $\tau^{\beta}(\underline{\text{false}})(x)$ implies that for all x' such that $(x,x') \in R$ no infinite "descending" sequence starting in x' exists (inductionhypothesis). Then there is no infinite "descending" sequence starting in x.

To prove the other implication, assume that $\neg\mu p.[R \to p](x)$ holds (which is equivalent to $\neg\tau^{\alpha}(\underline{\text{false}})(x)$).

By the fixedpoint property, $\neg(R \to \mu p.[R \to p])(x)$ holds too. So, there is an x_1 such that $(x,x_1) \in R$ and $\neg\mu p.[R \to p](x_1)$. This process can be repeated ad infinitum, and we obtain an infinite "descending" sequence $x_0, x_1, x_2, \ldots$ such that $x = x_0$ and $(x_i, x_{i+1}) \in R$ $(i \geq 0)$. ■

If F is a monotonic operator mapping predicates to predicates, then its greatest fixedpoint, $\nu p.[F(p)]$, exists too. This is because the collection of predicates as defined above is a complete lattice. Moreover the greatest fixedpoint is representable in terms of the μ-operator:
$\nu p.[F(p)] \Leftrightarrow \neg\mu p.\neg[F(p)\{\neg p/p\}]$.
A proof of this equivalence can be found in e.g. [3].
Using this result, we see that $\nu p.[R \circ p]$ exists and that
$$\nu p.[R \circ p] \Leftrightarrow \neg\mu p.[\neg(R \circ \neg p)]$$
$$\Leftrightarrow \neg\mu p.[R \to p].$$

Recall that "$\circ$" denotes composition of relations.
We adopt the convention that "$\circ$" has priority over "$\cup$".
I.e., $R_1 \circ R_2 \cup R_3$ should be parsed as $(R_1 \circ R_2) \cup R_3$.

Let R denote a relation over some set, and I the identity relation over the same set.
It is easily seen that $F(X) = R \circ X \cup I$ is monotonic in X, where X denotes a relation-variable. So F's least fixedpoint $\mu X[R \circ X \cup I]$ exists. In informal notation $\mu X.[R \circ X \cup I] = I \cup R \cup R^2 \cup \ldots \cup R^n \cup \ldots$.

We abbreviate $\mu X.[R \circ X \cup I]$ to R^*, the relation obtained by composing R, zero or more times with itself.

Let $R^+ = R \circ R^*$. Then we have the following

5.3 FACT

$I \subseteq R^*$, $R^+ \subseteq R^*$, $R^+ = R^* \circ R$.
If T denotes a relation and $T \subseteq R$ then $T^* \subseteq R^*$ and $R^* \circ T \subseteq R^*$

5.4 Let $\mathfrak{M}$ be some first-order structure.
The first-order logic over $\mathfrak{M}$ is defined as usual. Now we extend this logic so as to be able to express fixedpoint definitions. For this an infinite set of n-ary predicate-variables,
p, X, Y,..., is introduced for every $n \geq 0$. These predicate-variables may appear in formulae, but may not be bound by quantifiers. These variables form the basis of the fixedpoint definitions.

To ensure the existence of least (and greatest) fixedpoints, monotonicity has to be imposed.
In fact, we introduce the notion of syntactic monotonicity of formulae, which implies their (semantic) monotonicity. In essence, this notion requires that each occurrence of the induction-variable p is within the scope of an even number of $\neg$-signs.

5.5 DEFINITION
We inductively define sets sm(p), respectively, sa(p), denoting the class of formulae that are syntactically monotonic, respectively, syntactically anti-monotonic in a variable p:

(i) $\phi \in sm(p)$, if p does not occur free in ϕ.
(ii) $\neg\phi \in sm(p)$, if $\phi \in sa(p)$

(iii) $\phi_1 \supset \phi_2 \in sm(p)$, if $\phi_1 \in sa(p)$ and $\phi_2 \in sm(p)$.
(iv) $\forall x\phi, \exists x\phi \in sm(p)$, if $\phi \in sm(p)$.
(v) $p \in sm(p)$.

(vi) $\mu p_1.[\phi], \nu p_1.[\phi] \in sm(p)$, if $\phi \in sm(p) \cap sm(p_1)$.
(vii) (i)-(iv) with sm and sa interchanged.
(viii) $\mu p_1.[\phi], \nu p_1[\phi] \in sa(p)$, if $\phi \in sa(p) \cap sm(p_1)$.

Under the usual ordering, $\phi_1 \sqsubseteq \phi_2$ iff $\phi_1 \supset \phi_2$, it can be proved by induction on the structure (complexity) of the formula that syntactic monotonicity implies semantic monotonicity.

5.6 DEFINITION
The assertion-language L over some structure $\mathfrak{M}$, is the smallest class B such that

(i) $\phi, \mu p.[\psi(p)], \nu p.[\psi(p)] \in B$, where ϕ and ψ are first-order formulae over $\mathfrak{M}$, ϕ does not contain any free predicate-variables and $\psi \in sm(p)$.
(ii) if $\phi, \psi \in B$ then $\phi \wedge \psi, \phi \vee \psi, \phi \supset \psi$, and $\neg\phi \in B$, too.

Remark: If in a formula $\mu p.[\psi(p)]$ or $\nu p.[\psi(p)]$ p does not occur free in ψ, then we will often write ψ instead. Note that formulae of the form $\mu p.[\psi(p)]$, where ψ contains a μ-operator, are not allowed. However, we shall use such formulae, in which such a nesting of μ-operators occur, since they are representable in L (see [9]).

As a well-founded set is required to apply Orna's rule, we shall need recursive ordinals. In the sequel it is assumed that there are constants for all recursive ordinals. $\underline{\alpha},\underline{\beta},\ldots$ shall denote ordinal-constants. $\alpha,\beta,\ldots$ are used as ordinal-variables.

5.7 The definition of validity of L-formulae is clear, except for the cases $\mu p.[\psi(p)]$ and $\nu p.[\psi(p)]$. Recall that $\mu p.[\psi(p)]$ can be obtained by iteration. We now formalize this idea in the following construct by defining

predicates, I_ψ^β for $\beta \geq 0$ "by iterating ψ from below".

Note that the clauses (i) and (ii) below assures us that I_ψ^β is monotonic in β and that there exists some ordinal k for which the fixedpoint is reached. In fact, I_ψ (as defined below) is obtained after k iterations of ψ. Moreover, in this way indeed the least fixed point is obtained. This is just clause (iii) below.

To define validity of $\mu p.[\psi(p)]$, define predicates I_ψ^β for ordinals β by

$$I_\psi^0 = \lambda\bar{x}.\underline{\text{false}},\quad I_\psi^\beta = \lambda\bar{x}.\psi(\bar{x}, \bigsqcup_{\alpha<\beta} I_\psi^\alpha) \quad (\text{if } \beta \neq 0),\quad I_\psi = \lambda\bar{x}.\bigcup_{\alpha \geq 0} I_\psi^\alpha(\bar{x}).$$

By the monotonicity of ψ the following holds (see [9]):

(i) $(\alpha \leq \beta) \Rightarrow (I_\psi^\alpha(\bar{x}) \Rightarrow I_\psi^\beta(\bar{x}))$.

(ii) for some ordinal k: $I_\psi = I_\psi^k = \bigcup_{\alpha<k} I_\psi^\alpha$.

(iii) I_ψ is the least predicate C satisfying $C(\bar{x}) \Leftrightarrow \psi(\bar{x},C)$;
i.e., $I_\psi(\bar{x}) \Leftrightarrow \psi(\bar{x},I_\psi)$ and if C satisfies

$C(\bar{x}) \Leftrightarrow \psi(\bar{x},C)$ then $I_\psi(\bar{x}) \Rightarrow C(\bar{x})$.

We now put $\mathfrak{M} \models \mu p.[\psi(p)] \Leftrightarrow$ for all $\bar{x}$, $\mathfrak{M} \models \mu p.[\psi(p)](\bar{x})$

and $\mathfrak{M} \models \mu p.[\psi(p)](\bar{x}) \Leftrightarrow I_\psi(\bar{x})$.
Next, $\mathfrak{M} \models \nu p.[\psi(p)]$ iff $\mathfrak{M} \models \neg\mu p.\neg[\psi(p)\{\neg p/p\}]$.

As is usual in completeness proofs, we need the ability to code finite sequences. In this case, to define the well-founded set necessary for applying Orna's-rule.
For this, we introduce the notion of acceptability of a structure ([9)].

5.8 DEFINITION

(a) A coding scheme for a set A is a triple $\mathcal{C}=\langle N^{\mathcal{C}}, \leq^{\mathcal{C}}, \langle\rangle^{\mathcal{C}}\rangle$ such that

(i) $N^{\mathcal{C}} \subseteq A$, $\leq^{\mathcal{C}}$ is an ordering on $N^{\mathcal{C}}$ and the structure $\langle N^{\mathcal{C}}, \leq^{\mathcal{C}}\rangle$ is isomorphic to the integers with their usual ordering.

(ii) $\langle\rangle^{\mathcal{C}}$ is a one-to-one function, mapping the set $\bigcup_{n\geq 0} A^n$ of all finite sequences over A to A.

By convention $A^0=\emptyset$; the empty sequence $\langle\rangle^{\mathcal{C}}$ is the only sequence of length 0.

(b) With each coding scheme, $\mathcal{C}$, we assiociate the following decoding relations and functions:

(i) $\mathrm{Seq}^{\mathcal{C}}(x) \Leftrightarrow$ there exists $x_1,\ldots x_n$ such that $x=\langle x_1,\ldots,x_n\rangle^{\mathcal{C}}$.

(Here, $x=\langle\rangle^{\mathcal{C}}$, the code of the empty sequence, is covered by the convention that $x=\langle x_1,\ldots,x_n\rangle^{\mathcal{C}}$ if $n=0$.)

(ii) The length-function, $\mathrm{lh}^{\mathcal{C}}$, for sequences maps A into $N^{\mathcal{C}}$ and hence into the integers, because of the isomorphism of $\langle N^{\mathcal{C}}, \leq^{\mathcal{C}}\rangle$ with $\langle \mathbb{N}, \leq\rangle$:

$$\mathrm{lh}(x)^{\mathcal{C}}=\begin{cases} 0 & \text{if } \neg\mathrm{Seq}^{\mathcal{C}}(x) \\ n & \text{if } \mathrm{Seq}^{\mathcal{C}}(x) \wedge x=\langle x_1,\ldots,x_n\rangle^{\mathcal{C}} \text{ for some } x_1,\ldots,x_n. \end{cases}$$

(iii) The projection $(x)_i^{\mathcal{C}}$ (as a function of x and i) is defined by

$$(x)_i^{\mathcal{C}}=\begin{cases} x_i & \text{if for some } x_1,\ldots,x_n,\ x=\langle x_1,\ldots,x_n\rangle^{\mathcal{C}} \text{ and } 1\leq i\leq n \\ 0 & \text{otherwise} \end{cases}$$

5.9 DEFINITION

A coding scheme $\mathcal{C}$ is elementary on a structure $\mathfrak{M}$ if the relations and functions $N^{\mathcal{C}}$, $\leq^{\mathcal{C}}$, $\mathrm{seq}^{\mathcal{C}}$, $\mathrm{lh}^{\mathcal{C}}$, $()^{\mathcal{C}}$ are all elementary, i.e., first-order definable on $\mathfrak{M}$.
(A function f is elementary if its graph is, i.e., if $\{(\bar{x},\bar{y})\,|\,f(\bar{x})=\bar{y}\}$ is first-order definable.)

Note that the class of elementary relations on a structure is closed under conjunction and quantification. This is an immediate consequence of definition (5.9). It follows that the functions $p_n^{\mathcal{C}}$ defined by

$p_n^{\mathcal{C}}(x_1,\ldots,x_n)=\langle x_1,\ldots,x_n\rangle^{\mathcal{C}}$ are elementary, as

$p_n^{\mathcal{C}}(x_1,\ldots,x_n)=u \Leftrightarrow (\mathrm{Seq}^{\mathcal{C}}(u) \wedge \mathrm{lh}^{\mathcal{C}}(u)=n \wedge \forall i[1\leq i\leq n \supset ((u)_i^{\mathcal{C}}=x_i)]$.
In the sequel we shall omit the subscripts $\mathcal{C}$.

5.10 DEFINITION

A first-order structure $\mathfrak{M}$ is acceptable if it admits an elementary coding scheme on $\mathfrak{M}$.

Next, we show that a number of predicates that are extensively used in the sequel are representable in L.

5.11 Let R_1 and R_2 denote relations, elementary in $\mathfrak{M}$.
The following constructs are representable in L:
(i) $R_1 \circ R_2$ and $R_1 \cup R_2$: this is clear.

(ii) R_1^* : this term is representable by $\mu X.[R_1 \circ X \cup I]$
(where I denotes the identity relation).

(iii) $\mu p.[R_1 \to p]$: define $\phi(x,p) = \forall x'[R_1(x,x') \supset p(x')]$.[1)]
Then $\mu p.[\phi(x,p)]$ represents $\mu p.[R_1 \to p](x)$.
Note that this implies that $\nu p.[R_1 \circ p]$ is representable, too.

(iv) For predicates r and relations R, we define a construct $r \circ R$:
$r \circ R$ holds in x iff there exists some y satisfying r and yRx.
("x is R-reachable from r"). So $r \circ R(x) \Leftrightarrow \mathfrak{M} \models \exists y[r(y) \wedge R(y,x)]$.

Because of (5.11) we are justified in using informal notation.

Let $\mathfrak{M}$ be a first-order acceptable structure.
For completeness, we need, amongst others, representability of the guarded commands semantics. First note that the I/O-relation of a program S only constrains the valuation of its free variables (in the output-state). I.e., if $\xi R_S \xi'$ holds, then $\tau R_S \tau'$ holds, too, provided

$\xi|X = \tau|X$, $\xi'|X = \tau'|X$ and $\tau|X^C = \tau'|X^C$, where X is the set of free variables in S and $|$ denotes restriction. Using this observation, the semantics is easily seen to be representable:
For example, if $S \equiv *[b \to S']$ then $R_S(\xi,\xi') \Leftrightarrow \mathfrak{M} \models \mu X.[(b \circ R') \circ X \cup \neg b](x,y)$,

where x,y are the codes of $\xi|B$, respectively, $\xi'|B$. (Here R' denotes the relation associated with S', B the set of free variables occurring in S).

We construct an extension of $\mathfrak{M}$ by adding for every guarded command S a relation-symbol R_S, interpreted as the semantics of S. Since R_S is representable, we obtain a structure $\mathfrak{M}'$ such that $Th(\mathfrak{M}) = Th(\mathfrak{M}')$, where $Th(\mathfrak{M}) = \{p \in L \mid \mathfrak{M} \models p\}$. I.e., $Th(\mathfrak{M}')$ is conservative over $Th(\mathfrak{M})$ and we do not obtain a more expressive language in this way.

)1 $(x,x') \in R$ and $R(x,x')$ are used interchangeably in this paper.

Chapter 6

CONSTRUCTION OF A μ-TERM EXPRESSING FAIR TERMINATION

6.1 In this section we show that the property "S is fairly terminating" is representable in L.

More precisely, let $S \equiv *[\ \square_{i=1}^{n}\ b_i \to S_i\ ]$, and let $\mathfrak{M}$ be some acceptable structure. We construct a formula $FAIR(R_1,\ldots,R_n)$ such that

$\mathfrak{M} \models FAIR(R_1,\ldots,R_n)(\xi)$ iff "S terminates fairly in ξ"

Here, R_i denotes the relation associated with $b_i;S_i$ $(i=1,\ldots,n)$.

For programs with two directions, a μ-term expressing fair termination, has been constructed in [13].

To give the reader an idea, we construct such a term for the program $S \equiv *[y>0 \to x:=x+1 \ \square\ y>0 \to y:=y-1]$. This program terminates fairly. (Note that for this program fairness and impartiality coincide.)

Let R_1, respectively R_2, denote the relations associated with $y>0;x:=x+1$, respectively $y>0;y:=y-1$.

From section (3.2) we obtain that both R_1 and R_2, occur infinitely often in an infinite fair merge.

Now, we ask the question by what term the existence of an infinite fair sequence can be described. We consider such a sequence as consisting of an infinite number of so-called impartial parts, roughly being a finite subsequence of the infinite sequence in which every move occurs at least once.

Such an impartial part can be described as follows: $R_1^+ \circ R_2 \cup R_2^+ \circ R_1$.
This characterization stems from ([11]).
Remembering that truth of the predicate $\nu p.[R \circ p]$ expresses the existence of an infinite sequence $x_0,x_1,x_2,\ldots$ such that $x_i R x_{i+1}$ for $i \geq 0$, the existence of an infinite fair sequence is captured by the predicate $\nu p.[(R_1^+ \circ R_2 \cup R_2^+ \circ R_1) \circ p]$. Hence, program S terminates fairly in ξ iff

$\neg \nu p.[(R_1^+ \circ R_2 \cup R_2^+ \circ R_1) \circ p]$ $(\mu p.[(R_1^+ \circ R_2 \cup R_2^+ \circ R_1) \to p])$ holds in ξ.
It can be shown that this predicate holds for every ξ; consequently, S terminates fairly.

6.2 IMPARTIAL TERMINATION

At first, we ignore enabledness and disabledness of directions. I.e., we consider programs $*[\ \Box_{i=1}^{n}\ b \to S_i\ ]$.

For such programs fairness and impartiality coincide.

Assume that $R_1,\ldots,R_n$ are the relations associated with the statements $b;S_1,\ldots,b;S_n$ and also assume that truth of b in a state implies proper termination of S_i $(i=1,\ldots,n)$, when started in that state.

Consequently, we first consider the problem of describing in L the existence of an infinite sequence of R_i-moves in which each of the R_i occurs infinitely often $(i=1,\ldots,n)$.

Consider such an infinite sequence.

Since each R_i $(i=1,\ldots,n)$ occurs an infinite number of times, this sequence may be viewed as consisting of an infinite number of finite sequences, the so-called imp(artial)parts.

Every imppart satisfies:

(i) each R_i occurs in the imppart.

(ii) this imppart is the smallest sequence satisfying (i); i.e., any initial fragment of imppart leaves some R_i out.

To define a relation $\mathrm{Imppart}(R_1,\ldots,R_n)$, which expresses for every pair of states (ξ,ξ'), whether ξ' can be reached from ξ by executing an imppart (w.r.t. $R_1,\ldots,R_n$), it suffices to consider impparts in which the first occurrences of the moves are in some predescribed order, so-called impsegments, since any imppart of $R_1,\ldots,R_n$ is an impsegment of some permutation $R_{i_1},\ldots,R_{i_n}$.

More clearly, an impsegment of the ordered sequence of moves $R_1,\ldots,R_n$ is a finite sequence in which for no $1\leq i<j\leq n$ a R_j-move occurs before a R_i-move has occurred.

The relation $Imppart(R_1,\ldots,R_n)$ is defined inductively (w.r.t. n) as follows:
The case n=1 is simple: take $Impsegment(R_1)=R_1$.

Now, suppose that $Impsegment(R_1,\ldots,R_k)$ has been defined.
Then, $Impsegment(R_1,\ldots,R_{k+1})$ looks like $R_1,\ldots,R_i,\ldots,R_k,\ldots,R_{k+1}$, where the first occurrences of R_1,R_i,R_k,R_{k+1} are shown $(1<i<k)$.

First, observe that R_{k+1} occurs only once; this is a consequence of requirement (ii) above.

Secondly, observe that the prefix $R_1,\ldots,R_i,\ldots,R_k$ of the above sequence is an impsegment of $R_1,\ldots,R_k$. Hence, the sequence up to, but not including R_{k+1} is not necessarily an imppart of $R_1,\ldots,R_k$. However, it starts at least with an impsegment of $R_1,\ldots,R_k$. The remaining part may contain any (finite) number of R_i-occurrences (but no R_{k+1}).
This motivates the following definitions.

6.3 DEFINITION

$Impsegment(R_1)=R_1$
and for $n\geq 1$:
$Impsegment(R_1,\ldots,R_{n+1})=Impsegment(R_1,\ldots,R_n)\circ(R_1\cup\ldots\cup R_n)^*\circ R_{n+1}$.

EXAMPLE:
$Impsegment(R_1,R_2,R_3)=R_1\circ R_1^*\circ R_2\circ(R_1\cup R_2)^*\circ R_3$.

6.4 DEFINITION
For $n\geq 1$:
$$Imppart(R_1,\ldots,R_n)=\bigcup_{i_1,\ldots,i_n \text{ perm of } 1,\ldots,n} Impsegment(R_{i_1},\ldots,R_{i_n}).$$

(I.e., in $Imppart(R_1,\ldots,R_n)$ the order of the R_i $(i=1,\ldots,n)$ is immaterial.)

Remembering the example given in section (6.1), the existence of an infinite sequence of impartial parts, starting in a state ξ is expressed by satisfaction of a predicate $Imp(R_1,\ldots,R_n)$ in ξ, defined as follows:

6.5 DEFINITION
For $n\geq 1$: $Imp(R_1,\ldots,R_n)=\nu p.[Imppart(R_1,\ldots,R_n)\circ p]$
(Recall that R_i denote relations.)

So the program $S\equiv *[\ \Box_{i=1}^{n}\ b\to S_i\ ]$ admits an infinite fair execution sequence in ξ iff $Imp(R_1,\ldots,R_n)$ holds in ξ.
Here R_i denotes the relation associated with $b;S_i$ $(i=1,\ldots,n)$.

6.6 FAIR TERMINATION

Now, consider a program $S \equiv *[\ \underset{i=1}{\overset{n}{\square}}\ b_i \rightarrow S_i\ ]$

in which moves can be disabled. Assume that truth of b_i in a state implies proper termination of S_i, when started in that state. Let R_i denote the relation associated with S_i $(i=1,\ldots,n)$.

The case of an infinite fair execution-sequence in which every move $b_i \circ R_i$ is infinitely often enabled is easily tackled by the predicate $Imp(b_1 \circ R_1, \ldots, b_n \circ R_n)$.

Next, suppose that move $b_n \circ R_n$ becomes <u>eventually never enabled anymore.</u>

Then an infinite fair sequence of $b_1 \circ R_1, \ldots, b_n \circ R_n$-moves consists of some finite sequence of $b_1 \circ R_1, \ldots, b_n \circ R_n$-moves followed by an infinite fair sequence of $b_1 \circ R_1, \ldots, b_{n-1} \circ R_{n-1}$-moves in which every intermediate state satisfies $\neg b_n$. In case no other move becomes eventually continuously disabled, this is expressed by a predicate

$(b_1 \circ R_1 \cup \ldots \cup b_n \circ R_n)^* \circ Imp(b_1 \wedge \neg b_n \circ R_1, \ldots, b_{n-1} \wedge \neg b_n \circ R_{n-1})$. The possibility that other moves may become disabled, too, leads to the following definition:

6.7 DEFINITION)1

Let $n \geq 2$ and suppose that $i_1, \ldots, i_n$ is some permutation of $1, \ldots, n$.

For k, satisfying $1 \leq k < n$, define

$$fair(b_{i_1} \circ R_{i_1}, \ldots, b_{i_k} \circ R_{i_k}) fin(b_{i_{k+1}} \circ R_{i_{k+1}}, \ldots, b_{i_n} \circ R_{i_n}) =$$

$$\left(\bigcup_{i=1}^{n} b_i \circ R_i\right)^* \circ Imp\left(\left(b_{i_1} \wedge \bigwedge_{j=k+1}^{n} \neg b_{i_j}\right) \circ R_{i_1}, \ldots, \left(b_{i_k} \wedge \bigwedge_{j=k+1}^{n} \neg b_{i_j}\right) \circ R_{i_k}\right).$$

REMARK:

$fair(b_{i_1} \circ R_{i_1}, \ldots, b_{i_k} \circ R_{i_k}) fin(b_{i_{k+1}} \circ R_{i_{k+1}}, \ldots, b_{i_n} \circ R_{i_n})$ holds in state ξ iff there exists an infinite fair sequence, starting in ξ, in which the moves $b_{i_{k+1}} \circ R_{i_{k+1}}, \ldots, b_{i_n} \circ R_{i_n}$ are eventually never anabled anymore.

Now, finally the predicate expressing the existence of infinite fair sequences can be formulated.

)1 This definition is due to P. van Emde Boas.

6.8 DEFINITION

$FAIR(b_1 \circ R_1) = Imp(b_1 \circ R_1)$,

and for $n \geq 2$:

$$FAIR(b_1 \circ R_1, \ldots, b_n \circ R_n) = Imp(b_1 \circ R_1, \ldots, b_n \circ R_n) \vee$$

$$\vee_{i_1,\ldots,i_n \text{ perm of } 1,\ldots,n} \; \bigwedge_{1 \leq k < n} fair(b_{i_1} \circ R_{i_1}, \ldots, b_{i_k} \circ R_{i_k}) fin(b_{i_{k+1}} \circ R_{i_{k+1}}, \ldots, b_{i_n} \circ R_{i_n}).$$

In the sequel we assume that the guards b_i are incorporated in the relation R_i. Also, with R_i we always associate b_i as enabling-condition.

6.9 LEMMA

$\mathfrak{M} \models \neg FAIR(R_1, \ldots, R_n) \Leftrightarrow$

$[\mathfrak{M} \models \neg Imp(R_1, \ldots, R_n)$ and for all k such that $1 \leq k < n$

$$\mathfrak{M} \models \bigwedge_{i_1,\ldots,i_n \text{ perm of } 1,\ldots,n \; 1 \leq k < n} (\bigcup_{i=1}^{n} R_i)^* \rightarrow \neg Imp(\bigwedge_{j=k+1}^{n} \neg b_{i_j} \circ R_{i_1}, \ldots, \bigwedge_{j=k+1}^{n} \neg b_{i_j} \circ R_{i_k})].$$

PROOF

For n=1 this follows by definition (6.8). So assume that $n \geq 2$. Then the lemma follows from definition (6.8), definition (6.7) and section (5.2)(the representability of the greatest fixed-point operator in terms of the least fixed-point operator).

As a last preparation for the soundness and completeness proofs, we mention the notion of the weakest precondition:

6.10 DEFINITION

An assertion p=wlp(S,q) is the weakest liberal precondition w.r.t. a command S and a condition q if

(i) $\mathfrak{M} \models \{p\}S\{q\}$

(ii) For each r $\mathfrak{M} \models \{r\}S\{q\}$ implies $\mathfrak{M} \models r \supset p$.

Here {p}S{q} holds iff for all ξ: if input state ξ satisfies p and if S terminates, when started in ξ, then each output state satisfies q (partial correctness).

Let $S \equiv *[\Box_{i=1}^{n} b_i \rightarrow S_i]$ and let R_i denotes the relation associated with $b_i;S_i$ (i=1,...,n).

It has been shown that for each assertion q the wlp(S,q) is definable in L (see [3]). It is useful to mention that in this case

$$wlp(S,q) = ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i) \rightarrow q).$$

"if the repetition S terminates then q is satisfied in each final state, provided wlp(S,q) is satisfied at the start of the execution".

6.11 DEFINITION
An assertion p=wp(S,q) is the weakest precondition w.r.t. a command S and a condition q if
$\mathfrak{M} \models [p]S[q]$ and for all r, $\mathfrak{M} \models [r]S[q]$ implies $\mathfrak{M} \models r \supset p$.
"S always terminates in a state satisfying q, provided wp(S,q) is satisfied at the start of the execution" (total correctness).

The key theorem of this section is the following

6.12 THEOREM

For every ξ: $wp(fair(*[\ \Box_{i=1}^{n} b_i \to S_i\ ])(\xi) \Leftrightarrow$

$$\mathfrak{M} \models \neg FAIR(R_1,\ldots,R_n) \wedge ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i) \to q)(\xi),$$

where R_i are the relations associated with $b_i;S_i$ $(i=1,\ldots,n)$.

PROOF
We have to show that

$$\mathfrak{M} \models [r]fair(*[\ \Box_{i=1}^{n} b_i \to S_i\ ])[q] \Leftrightarrow$$

$$\mathfrak{M} \models r \supset (\neg FAIR(R_1,\ldots,R_n) \wedge (\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i \to q),$$

"$\Rightarrow$" Suppose that $\mathfrak{M} \models [r]fair(*[\ \Box_{i=1}^{n} b_i \to S_i\ ])[q]$ holds.

Choose some state ξ such that $\mathfrak{M} \models r(\xi)$ holds.

Assume to obtain a contradiction that $\mathfrak{M} \models FAIR(R_1,\ldots,R_n)(\xi)$.

Then this leads immediately to a contradiction, since this implies the existence of an infinite fair execution sequence, starting in ξ.

So $\mathfrak{M} \models \neg FAIR(R_1,\ldots,R_n)(\xi)$ holds.

It remains to prove that $\mathfrak{M} \models (((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i) \to q)(\xi)$ holds, too.

To do this, choose some ξ' such that $\mathfrak{M} \models ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i)(\xi,\xi')$.

Clearly, then also $\mathfrak{M} \models fair(*[\ \Box_{i=1}^{n} b_i \to S_i\ ])(\xi,\xi')$, and so by the hypothesis $\mathfrak{M} \models q(\xi')$.

"$\Leftarrow$" Suppose that $\mathfrak{M} \models r \supset (\neg FAIR(R_1,\ldots,R_n) \wedge ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i) \rightarrow q)$. Choose state ξ such that $\mathfrak{M} \models r(\xi)$.

Since, by hypothesis $\mathfrak{M} \models \neg FAIR(R_1,\ldots,R_n)(\xi)$, the repetition always terminates fairly. To prove, that in this case, each final state satisfies q.

Choose some ξ' such that $\mathfrak{M} \models fair(*[\ \Box_{i=1}^{n} b_i \rightarrow S_i\ ])(\xi,\xi')$.

Clearly, then also $\models ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i)(\xi,\xi')$ and so by the hypothesis $\mathfrak{M} \models q(\xi')$ holds, which had to be shown.

6.13 COROLLARY
For every ξ:

$$wp(fair(*[\ \Box_{i=1}^{n} b_i \rightarrow S_i\ ]), \underline{true})(\xi) \Leftrightarrow$$
$$\mathfrak{M} \models \neg FAIR(R_1,\ldots,R_n)(\xi).$$

PROOF
Immediately from theorem (6.12).

This corollary states that fair termination of a repetition is indeed expressible in the μ-calculus.

The remainder of this paper deals with the soundness and completeness proof of our proofsystem.
Note that the only rule for which this is non-trivial is Orna's rule.
The proof that this rule is both sound and complete is given by induction on n, the number of directions of the repetition. The case of only one guard is trivial, so assume that $n \geq 2$.

In the next two chapters, we prove completeness and soundness of this rule under the induction hypothesis that for all repetitions

$S \equiv *[\ \Box_{i=1}^{k} b_i \rightarrow S_i\ ]$, where $k<n$, and for all r and q,

$$\mathfrak{M} \models [r]fair(*[\ \Box_{i=1}^{k} b_i \rightarrow S_i\ ])[q] \Leftrightarrow Th(\mathfrak{M}) \vdash [r]fair(*[\ \Box_{i=1}^{k} b_i \rightarrow S_i\ ])[q]$$

where $Th(\mathfrak{M}) = \{p \in L \mid \mathfrak{M} \models p\}$, i.e., we assume that the theorem has been proved for syntactically simpler fair repetitions (meaning programs with less then n directions).

Chapter 7
COMPLETENESS

7.1 THEOREM
Let $\mathfrak{M}$ be a first-order acceptable structure. Then our system is relative complete, meaning that for any statement S and assertions r,q∈L:
$\mathfrak{M}\models[r]S[q] \Rightarrow Th(\mathfrak{M})\vdash[r]S[q]$, where $Th(\mathfrak{M})=\{p\in L \mid \mathfrak{M}\models p\}$.
PROOF

The only non-trivial case is when $S\equiv fair(*[\ \Box_{i=1}^{n} b_i\to S_i])$, and $n\geq 2$. In that case, we must show that Orna's rule can be applied.

Assume that $\mathfrak{M}\models[r]S[q]$ holds. By theorem (6.12), we may assume, too, that $\mathfrak{M}\models r \supset (\neg FAIR(R_1,\ldots,R_n) \wedge ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i)\to q)$.

At first, we must define a well-founded set W and a predicate $\pi: W\to(States\to\{true,false\})$, ranking every state (reachable by S).

To do so, we observe that the usual approach of counting moves does not work, because not every move need to bring the program closer to termination. (E.g., in case of Dijkstra's random number generator (see section (3.1)) move R_1 will not help reaching termination.)

Now S terminates fairly and hence also impartially (see section (3.4)). At any time, there is at least one decreasing move (otherwise there exists a state in which no move would bring the program closer to termination, resulting in the existence of an infinite fair sequence; contradiction). So, if in a successive sequence of iterations, "every enabled move has been executed at least once", then certainly the program has come closer to termination. This shows that viewing execution-sequences as consisting of impparts, is a natural thing to do. Unfortunately, counting impparts does not quite work, because we have to rank all states in order for Orna's rule to apply.

Consider such an imppart. It suffices that the states reached by execution this imppart, are ranked in such a way that it reflects the "progress" that is made w.r.t. executing this imppart itself.

Now a move leads to "progress" if it is a new one that has not been made in the imppart as yet.
This gives the intuition behind the definitions of W and π that we now develop.
From $\mathfrak{M}\models\neg FAIR(R_1,\ldots,R_n) \Rightarrow \mathfrak{M}\models\neg Imp(R_1,\ldots,R_n)$, we obtain that

$\mathfrak{M}\models\neg FAIR(R_1,\ldots,R_n) \Rightarrow \mathfrak{M}\models\mu p.[Imppart(R_1,\ldots,R_n)\to p]$, by applying the definitions.

As we saw in section (5.2) least fixedpoints can be obtained by iteration: Define τ by $\tau(p)=\lambda\xi.(Imppart(R_1,\ldots,R_n)\to p)(\xi)$

Then there exists some ordinal $\bar{\lambda}$ such that
$\tau^{\bar{\lambda}}(false)=\mu p.[Imppart(R_1,\ldots,R_n)\to p]$. (*)

Let $\bar{\alpha}$ denote the least ordinal satisfying (*).
If $\bar{\beta}\leq\bar{\alpha}$ then $\tau^{\bar{\beta}}(false)(\xi)$ holds for some ξ iff in ξ we are at most $\bar{\beta}$ impparts away from termination.
This gives us a way to rank the states related by impparts.

Of course, for this idea to work we need to show that $\tau^{\bar{\beta}}(\underline{\text{false}})$ is representable by a formula in L:
(Note that α is a recursive ordinal since it is less than or equal to the ordinal associated with the execution tree of S, which is recursive, cf. [1]).

7.2 THEOREM
Let $\mathfrak{M}$ be a first-order acceptable structure.
There exists a formula ϕ in L such that for all ξ and all $\bar{\beta}\leq\bar{\alpha}$

$\tau^{\bar{\beta}}(\underline{\text{false}})(\xi)$ holds iff $\mathfrak{M}\models\phi(\bar{\beta})(\xi)$.

PROOF

Define $\phi(\beta)=\mu r.[\exists\alpha<\beta(\text{Imppart}(R_1,\ldots,R_n)\rightarrow r(\alpha))]$. By induction on $\bar{\beta}\leq\bar{\alpha}$ we prove that for all $\bar{\beta}\leq\bar{\alpha}$ and all ξ, $\tau^{\bar{\beta}}(\underline{\text{false}})(\xi)$ iff $\mathfrak{M}\models\phi(\bar{\beta})(\xi)$.

$\bar{\beta}=\bar{0}$: trivial, since for all ξ, $\tau^{\bar{0}}(\underline{\text{false}})(\xi)\Leftrightarrow$ false
and $\mathfrak{M}\models\phi(\bar{0})(\xi)\Leftrightarrow\mathfrak{M}\models\underline{\text{false}}(\xi)\Leftrightarrow$ false.

Inductionhypothesis (IH): suppose that for all $\bar{\lambda}<\bar{\beta}$ and all ξ,

$\tau^{\bar{\lambda}}(\underline{\text{false}})(\xi)$ holds iff $\mathfrak{M}\models\phi(\bar{\lambda})(\xi)$.
For $\bar{\beta}\neq\bar{0}$ we have that
$\mathfrak{M}\models\phi(\bar{\beta})(\xi)\Leftrightarrow$

$\mathfrak{M}\models\mu r.[\exists\alpha<\bar{\beta}(\text{Imppart}(R_1,\ldots,R_n)\rightarrow r(\alpha))](\xi)$ (definition of ϕ) $\Leftrightarrow$
$\mathfrak{M}\models\exists\alpha<\bar{\beta}(\text{Imppart}(R_1,\ldots,R_n)\rightarrow\phi(\alpha))(\xi)$ (fixedpoint property) $\Leftrightarrow$

for some $\bar{\lambda}<\bar{\beta}$, $\mathfrak{M}\models(\text{Imppart}(R_1,\ldots,R_n)\rightarrow\phi(\bar{\lambda}))(\xi)\Leftrightarrow$
for some $\bar{\lambda}<\bar{\beta}$ and for all ξ', $\mathfrak{M}\models[\text{Imppart}(R_1,\ldots,R_n)(\xi,\xi')\supset\phi(\bar{\lambda})(\xi')]\Leftrightarrow$
for some $\bar{\lambda}<\bar{\beta}$ and all for ξ',

$\mathfrak{M}\models[\text{Imppart}(R_1,\ldots,R_n)(\xi,\xi')]\Rightarrow\tau^{\bar{\lambda}}(\underline{\text{false}})(\xi')$ (IH) $\Leftrightarrow$
for all ξ',
$\mathfrak{M}\models\text{Imppart}(R_1,\ldots,R_n)(\xi,\xi')\Rightarrow(\exists\lambda<\bar{\beta}\ \tau^{\lambda}(\underline{\text{false}})(\xi'))\Leftrightarrow$

for all ξ', $\mathfrak{M}\models\text{Imppart}(R_1,\ldots,R_n)(\xi,\xi')\Rightarrow\bigsqcup_{\bar{\lambda}<\bar{\beta}}\tau^{\bar{\lambda}}(\underline{\text{false}})(\xi')\Leftrightarrow$

$\tau^{\bar{\beta}}(\underline{\text{false}})(\xi)$.

Now, we define the well-founded ordered set W and the ranking predicate π: Each $w \in W$, w not minimal, consists of two components: the first one counts impparts, the second one records "progress" within the last (incomplete) imppart, and is a sequence of length at most n (the number of directions of the repetition), which records the directions within this imppart, that have already been taken.

7.3 DEFINITION

$$\mathrm{seq}_n(s) = \mathrm{Seq}(s) \wedge \mathrm{lh}(s) \leq n \wedge \forall\, i[(1 \leq i \leq \mathrm{lh}(s)) \supset (1 \leq (s)_i \leq n)] \wedge$$
$$\wedge \forall\, i,j[(1 \leq i,j \leq \mathrm{lh}(s) \wedge i \neq j) \supset (s)_i \neq (s)_j].$$

(cf. definition (5.8))

Note that only directions are recorded in $s \in \mathrm{seq}_n$ and each direction at most once!

7.4 DEFINITION

$$W_{\bar{\alpha},n} = \{(\bar{\lambda},s) \mid \bar{0} \leq \bar{\lambda} \leq \bar{\alpha} \wedge \mathrm{seq}_n(s)\} \cup \{\bar{0}\}.$$

The ordering $\prec$ defined on $W_{\bar{\alpha},n}$ is the following:

$\bar{0} \prec (\bar{\lambda},s)$ for all $(\bar{\lambda},s) \in W_{\bar{\alpha},n}$ and

$$(\bar{\lambda}_1,s_1) \prec (\bar{\lambda}_2,s_2) \text{ iff } (\bar{\lambda}_1 < \bar{\lambda}_2) \vee ((\bar{\lambda}_1 = \bar{\lambda}_2) \wedge \mathrm{lh}(s_2) < \mathrm{lh}(s_1) \wedge$$
$$\wedge \forall i[(1 \leq i \leq \mathrm{lh}(s_2)) \supset (s_2)_i = (s_1)_i]).$$

7.5 DEFINITION

The predicate $\pi : W_{\bar{\alpha},n} \to (\mathrm{States} \to \{\underline{\mathrm{true}},\underline{\mathrm{false}}\})$ is defined by:

$$\pi(\bar{\lambda},\langle\rangle) = \tau^{\bar{\lambda}}(\underline{\mathrm{false}}) \wedge r \circ (\bigcup_{i=1}^{n} R_i)^* \wedge \bigvee_{i=1}^{n} b_i,$$

$$\pi(\bar{\lambda},\langle i_1,\ldots,i_k\rangle) = \tau^{\bar{\lambda}}(\underline{\mathrm{false}}) \circ (\mathrm{Impsegment}(R_{i_1},\ldots,R_{i_k}) \circ (\bigcup_{i=1}^{k} R_{i_j})^*) \wedge$$
$$\wedge r \circ (\bigcup_{i=1}^{n} R_i)^* \wedge \bigvee_{i=1}^{n} b_i \quad (\text{for } 1 \leq k < n),$$

$$\pi(\bar{\lambda},\langle i_1,\ldots,i_n\rangle) = \bigsqcup_{\bar{\beta}<\bar{\lambda}} \tau^{\bar{\beta}}(\underline{\mathrm{false}}) \wedge r \circ (\bigcup_{i=1}^{n} R_i)^* \wedge \bigvee_{i=1}^{n} b_i,$$

$$\pi(\bar{0}) = \bigwedge_{i=1}^{n} \neg b_i.$$

REMARK

Note that accessibility is demanded in case $w \succ \bar{0}$ (i.e., $\bar{0} \prec w$).

If $1 \leq k < n$ and $\pi(\bar{\lambda},\langle i_1,\ldots,i_k\rangle)(\xi)$ holds, then there exists a state ξ' in which the program is at most $\bar{\lambda}$ impparts away from termination. It takes a fragment (i.e., an initial part) of an impsegment to reach ξ from ξ', namely $\mathrm{Impsegment}(R_{i_1},\ldots,R_{i_k}) \circ (\bigcup_{j=1}^{k} R_{i_j})^*$.

Satisfaction of the clauses (a),...,(d) of Orna's rule for this choice of W and π follows from several lemmata and definitions, by checking that its four clauses hold indeed.

Defining St_w and D_w for $w \succ \bar{0}$ is simple now. If we are at the start of an imppart (i.e., $w=(\bar{\lambda},<>)$ or $w=(\bar{\lambda},<i_1,\ldots,i_n>)$ for some $\bar{\lambda}\leq\bar{\alpha}$) then every move leads to eventual completion of this imppart.
Otherwise, $w=(\bar{\lambda},<i_1,\ldots,i_k>)$ for some $\bar{\lambda}$, $1\leq k<n$, and only moves different from $R_{i_1},\ldots,R_{i_k}$ lead to eventual completion of this imppart.

7.6 DEFINITION
Let $w\in W_{\bar{\alpha},n}$, $w=(\bar{\lambda},s)$.

If $lh(s)=0$ or if $lh(s)=n$ then $D_w=\{1,\ldots,n\}$ and $St_w=\emptyset$.
If $0<lh(s)<n$ then $D_w=\{i \mid (1\leq i\leq n) \wedge \forall j\leq lh(s)[(s)_j\neq i]\}$, $St_w=\{1,\ldots,n\}-D_w$.

Note that for all $w\in W_{\bar{\alpha},n}$, $w \succ \bar{0}: D_w \cap St_w=\emptyset$, $D_w\neq\emptyset$ and $D_w \cup St_w=\{1,\ldots,n\}$.

7.7 LEMMA

Let $w\in W_{\bar{\alpha},n}$, $j\in D_w$, (i.e., R_j is a decreasing move). Suppose that $\mathfrak{M}$ is a first-order acceptable structure and that

$$\mathfrak{M} \models r \supset (\neg FAIR(R_1,\ldots,R_n) \wedge ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i) \rightarrow q) \text{ holds.}$$

Then $Th(\mathfrak{M}) \vdash [\pi(w) \wedge w \succ \bar{0} \wedge b_j] S_j [\exists v \prec w\ \pi(v)]$ holds, too.

PROOF

We have to prove that for all $\xi,\xi'\in$States such that $\mathfrak{M} \models R_j(\xi,\xi')$,

$$\mathfrak{M} \models (\pi(w) \wedge w \succ \bar{0})(\xi) \Rightarrow \mathfrak{M} \models \exists v \prec w\ \pi(v)(\xi')^{1)}.$$

Choose states ξ,ξ' satisfying $\mathfrak{M} \models R_j(\xi,\xi')$ and suppose that $\mathfrak{M} \models (\pi(w) \wedge w \succ \bar{0})(\xi)$ holds.

We distinguish two cases:

(a) $\mathfrak{M} \models \bigwedge_{i=1}^{n} \neg b_i(\xi')$.
In this case, $\mathfrak{M} \models \pi(\bar{0})(\xi')$, and we are done.

---

1) Remember that R_j is the relation associated with $b_j;S_j$.

(b) $\mathfrak{M} \models \bigvee_{i=1}^{n} b_i(\xi')$. (i)

Since $\mathfrak{M} \models \pi(w)(\xi)$ holds, $\mathfrak{M} \models r \circ (\bigcup_{i=1}^{n} R_i)^*(\xi)$ holds, too.

I.e., $\mathfrak{M} \models \exists \xi''[r(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'', \xi)]$. (ii)

Now $(\bigcup_{i=1}^{n} R_i)^* \circ R_j \subseteq (\bigcup_{i=1}^{n} R_i)^*$ (fact (5.3)).

So it follows from $\mathfrak{M} \models R_j(\xi, \xi')$ and (ii) that

$\mathfrak{M} \models \exists \xi''[r(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'', \xi')]$, i.e.,

$\mathfrak{M} \models r \circ (\bigcup_{i=1}^{n} R_i)^*(\xi')$. (iii)

Let $w=(\bar{\lambda}, s)$. To prove $\mathfrak{M} \models \exists v \prec w\ \pi(v)(\xi')$.
We distinguish three cases:

(1) $lh(s)=0$, so $s=\langle\rangle$.
Since $\mathfrak{M} \models \pi(w)(\xi)$, $\mathfrak{M} \models \tau^{\bar{\lambda}}(\underline{\text{false}})(\xi)$ holds. Consequently, it follows

that $\mathfrak{M} \models \exists \xi''[\tau^{\bar{\lambda}}(\underline{\text{false}})(\xi'') \wedge R_j(\xi'', \xi')]$.
Hence, from $R_j \subseteq R_j^+$ (fact (5.3)) we have that

$\mathfrak{M} \models \exists \xi''[\tau^{\bar{\lambda}}(\underline{\text{false}})(\xi'') \wedge R_j^+(\xi'', \xi')]$, i.e., $\mathfrak{M} \models (\tau^{\bar{\lambda}}(\underline{\text{false}}) \circ R_j^+)(\xi')$.

Together with (i) and (iii), $\mathfrak{M} \models \pi(\bar{\lambda}, \langle j \rangle)(\xi')$ follows and hence $\mathfrak{M} \models \exists v \prec w\ \pi(v)(\xi')$.

(2) $1 \leq lh(s) < n$, so $s=\langle i_1, \ldots, i_k \rangle$ for some $i_1, \ldots, i_k$ such that $\{i_1, \ldots, i_k\} \subseteq \{1, \ldots, n\}$ and $1 \leq k < n$. From $\mathfrak{M} \models \pi(w)(\xi)$ we derive

$\mathfrak{M} \models (\tau^{\bar{\lambda}}(\underline{\text{false}}) \circ \text{Impsegment}(R_{i_1}, \ldots, R_{i_k}) \circ (\bigcup_{t=1}^{k} R_{i_t})^*)(\xi)$.

Since $\text{Impsegment}(R_{i_1}, \ldots, R_{i_k}) \circ (\bigcup_{t=1}^{k} R_{i_t})^* \circ R_j$

$= \text{Impsegment}(R_{i_1}, \ldots, R_{i_k}, R_j)$

(definition (6.3) and $j \neq i_1, \ldots, i_k$ for $j \in D_w$)

$\subseteq \text{Impsegment}(R_{i_1}, \ldots, R_{i_k}, R_j) \circ (\bigcup_{t=1}^{k} R_{i_t} \cup R_j)^*$ (fact (5.3)).

This, together with the fact that $\mathfrak{M} \models R_j(\xi, \xi')$ holds, implies that

$\mathfrak{M} \models \tau^{\bar{\lambda}}(\underline{\text{false}}) \circ \text{Impsegment}(R_{i_1},\ldots,R_{i_k},R_j)(\xi')$ holds, too. It follows together with (i) and (iii) that $\mathfrak{M} \models \pi(\bar{\lambda},\langle i_1,\ldots,i_k,j\rangle)(\xi')$ holds.

Again, $\mathfrak{M} \models \exists v \prec w\ \pi(v)(\xi')$ follows.

(3) $lh(s)=n$.
From $\mathfrak{M} \models \pi(\bar{\lambda},s)(\xi)$ and definition (7.5), the existence of a $\bar{\beta}<\bar{\lambda}$ such that $\mathfrak{M} \models \pi(\bar{\beta},\langle\rangle)(\xi)$ follows.
As in case (1), $\mathfrak{M} \models \exists v \prec (\bar{\beta},\langle\rangle)[\pi(v)(\xi')]$, and so
$\mathfrak{M} \models \exists v \prec (\bar{\lambda},\langle\rangle)[\pi(v)(\xi')]$.

7.8 LEMMA

Let $w \in W_{\bar{\alpha},n}$, $j \in St_w$, (i.e., R_j is a steady move). Suppose that $\mathfrak{M}$ is a first-order acceptable structure and that

$$\mathfrak{M} \models r \supset (\neg FAIR(R_1,\ldots,R_n) \wedge ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i) \rightarrow q) \text{ holds.}$$

Then $Th(\mathfrak{M}) \vdash [\pi(w) \wedge w>0 \wedge b_j]S_j[\exists v \preceq w\ \pi(v)]$ holds, too.

PROOF
We have to show that for all $\xi,\xi' \in States$ such that $\mathfrak{M} \models R_j(\xi,\xi')$,

$\mathfrak{M} \models (\pi(w) \wedge w>0)(\xi) \Rightarrow \mathfrak{M} \models \exists v \preceq w\ \pi(v)(\xi')$.
Choose states ξ,ξ' and suppose that $\mathfrak{M} \models (\pi(w) \wedge w>0)(\xi)$ holds.
Let $w=(\bar{\lambda},s)$. As in lemma (7.7) there are two cases.

(a) $\mathfrak{M} \models \bigwedge_{i=1}^{n} \neg b_i(\xi')$.
Trivial.

(b) $\mathfrak{M} \models \bigvee_{i=1}^{n} b_i(\xi')$. (i)
To prove $\mathfrak{M} \models \exists v \prec w\ pi(v)(\xi')$.
Note that $lh(s) \neq 0$ and $lh(s) \neq n$, because $lh(s)=0$ or $lh(s)=n$ implies that $St_w=\emptyset$.

So let $w=(\bar{\lambda},\langle i_1,\ldots,i_k\rangle)$, $1 \leq k<n$, $\{i_1,\ldots,i_k\} \subseteq \{1,\ldots,n\}$.

Since $j \in St_w$, $j=i_t$ for some t, $1 \leq t \leq k$.
Now, $\mathfrak{M} \models \pi(w)(\xi)$, so

$$\mathfrak{M} \models (\tau^{\bar{\lambda}}(\underline{\text{false}}) \circ \text{Impsegment}(R_{i_1},\ldots,R_{i_k}) \circ (\bigcup_{t=1}^{k} R_{i_t})^*(\xi), \text{ i.e.,}$$

$$\mathfrak{M} \models \exists \xi''[\tau^{\bar{\lambda}}(\underline{\text{false}})(\xi'') \wedge \text{Impsegment}(R_{i_1},\ldots,R_{i_k}) \circ (\bigcup_{t=1}^{k} R_{i_t})^*(\xi'',\xi)]. \quad \text{(ii)}$$

Since $(\bigcup_{t=1}^{k} R_{i_t})^* \circ R_j \subseteq (\bigcup_{t=1}^{k} R_{i_t})^*$ (fact (5.3)), we obtain that

$$\text{Impsegment}(R_{i_1},\ldots,R_{i_t}) \circ (\bigcup_{t=1}^{k} R_{i_t})^* \circ R_j \subseteq \text{Impsegment}(R_{i_1},\ldots,R_{i_t}) \circ (\bigcup_{t=1}^{k} R_{i_t})^*.$$

From (ii) and the fact that $\mathfrak{M} \models R_j(\xi,\xi')$ it then follows that

$$\mathfrak{M} \models \exists \xi'[\tau^{\bar{\lambda}}(\underline{\text{false}})(\xi'') \wedge \text{Impsegment}(R_{i_1},\ldots,R_{i_k}) \circ (\bigcup_{t=1}^{k} R_{i_t})^*(\xi'',\xi')],$$

i.e., $$\mathfrak{M} \models (\tau^{\bar{\lambda}}(\underline{\text{false}}) \circ \text{Impsegment}(R_{i_1},\ldots,R_{i_k}) \circ (\bigcup_{t=1}^{k} R_{i_t})^*)(\xi')], \quad \text{(iii)}$$

Moreover, as in the proof of lemma (7.7), we see that

$$\mathfrak{M} \models r \circ (\bigcup_{i=1}^{n} R_i)^*(\xi') \quad \text{(iv)}$$

Now, (i), (iii) and (iv) imply $\mathfrak{M} \models \pi(\bar{\lambda},\langle i_1,\ldots,i_k\rangle)(\xi')$,

whence $\mathfrak{M} \models \exists v \lessdot w\ \pi(v)(\xi')$.

7.9 LEMMA

Suppose that $\mathfrak{M} \models r \supset (\neg \text{FAIR}(R_1,\ldots,R_n) \wedge ((\bigcup_{i=1}^{n} R_i)^* \circ \bigwedge_{i=1}^{n} \neg b_i) \rightarrow q)$ and the inductionhypothesis (i.e., for all k such that $1 \leq k < n$

$$\mathfrak{M} \models [r]\text{fair}(*[\ \Box_{i=1}^{k}\ b_i \rightarrow S_i\ ])[q] \Leftrightarrow \text{Th}(\mathfrak{M}) \vdash [r]\text{fair}(*[\ \Box_{i=1}^{k}\ b_i \rightarrow S_i\ ])[q]))$$ both hold.

Then $\text{Th}(\mathfrak{M}) \vdash [\pi(w) \wedge w \succ \bar{0}]\text{fair}(*[\ \Box_{i \in St_w} b_i \wedge \bigwedge_{j \in D_w} \neg b_j \rightarrow S_i\ ])[\underline{\text{true}}]$ holds,

too.

PROOF

Observe that for all $w \in W_{\alpha,n}$ such that $w \succ \bar{0}$, $D_w \neq \emptyset$.

So $St_w \subsetneq \{1,\ldots,n\}$. It follows that the program

$S' \equiv *[\ \Box_{i \in St_w} b_i \wedge \bigwedge_{j \in D_w} \neg b_j \rightarrow S_i\ ]$ contain less directions, than the original program, so we may apply the inductionhypothesis. If $St_w = \emptyset$ then by convention $S' \equiv \underline{\text{skip}}$ (x:=x), in which case the lemma is trivial. So assume $St_w \neq \emptyset$.

After a possible renumbering, we may assume, too, that $St_w = \{1,\ldots,k\}$, $1 \leq k < n$. So, $D_w = \{k+1,\ldots,n\}$.

Let b' denote $\bigwedge_{j \in D_w} \neg b_j = \bigwedge_{j=k+1}^{n} \neg b_j$, and let $R_i' = b' \circ R_i$.

By induction $\text{Th}(\mathfrak{M}) \vdash [\pi(w) \wedge w \succ \bar{0}]\text{fair}(*[\ \Box_{i=1}^{k}\ b_i \wedge \bigwedge_{j=k+1}^{n} \neg b_j \rightarrow S_i\ ])[\underline{\text{true}}]$

iff $\mathfrak{M} \models (\pi(w) \wedge w \succ \bar{0}) \supset \neg \text{FAIR}(R_1',\ldots,R_k')$ (cf. section (5.2)).

So, to prove the lemma, it suffices to show that $\mathfrak{M} \models (\pi(w) \wedge w \succ \bar{0}) \supset \neg \text{FAIR}(R_1',\ldots,R_k')$.

This follows from the next two claims:

CLAIM 1
Under the aforementioned assumptions,

$\mathfrak{M} \models (\pi(w) \wedge w \succ \overline{0}) \supset \neg Imp(R'_1,\ldots,R'_k))$ holds.

PROOF (of claim 1)
Suppose that $\mathfrak{M} \models \pi(w)(\xi) \wedge w \succ \overline{0}$.

Then $\mathfrak{M} \models r \circ (\bigcup_{i=1}^{n} R_i)^*(\xi)$, i.e., $\mathfrak{M} \models \exists \xi''[r(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'',\xi)]$.
As a consequence of our assumption, we obtain
$\mathfrak{M} \models r \supset \neg FAIR(R_1,\ldots,R_n)$ and so

$\mathfrak{M} \models \exists \xi''[\neg FAIR(R_1,\ldots,R_n)(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'',\xi)]$. Thus,

$\mathfrak{M} \models \exists \xi''[((\bigcup_{i=1}^{n} R_i)^* \rightarrow \neg Imp((\bigwedge_{i=k+1}^{n} \neg b_i) \circ R_1,\ldots,(\bigwedge_{i=k+1}^{n} \neg b_i) \circ R_k))(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'',\xi)]$,
(lemma (6.9)). Consequently,

$\mathfrak{M} \models \exists \xi''[((\bigcup_{i=1}^{n} R_i)^* \rightarrow \neg Imp(R'_1,\ldots,R'_k))(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'',\xi)]$,
from which $\mathfrak{M} \models \neg Imp(R'_1,\ldots,R'_k)(\xi)$ follows (definition of $R \rightarrow p$). ■

Now, if k=1, the lemma follows immediately from claim 1 and definition (6.8). So assume that $k \geq 2$.

CLAIM 2
Under the aforementioned assumptions,

$$\mathfrak{M} \models (\pi(w) \wedge w \succ \overline{0}) \supset \bigwedge_{\substack{i_1,\ldots,i_k \text{ perm of } 1,\ldots,k \\ 1 \leq l < k}} \neg fair(R'_{i_1},\ldots,R'_{i_l}) fin(R'_{i_{l+1}},\ldots,R'_{i_k}) \text{ holds.}$$

PROOF (of claim 2)
Let $1 \leq l < k$. For simplicity we prove that

$\mathfrak{M} \models (\pi(w) \wedge w \succ \overline{0}) \supset \neg fair(R'_1,\ldots,R'_l) fin(R'_{l+1},\ldots,R'_k)$ (any other permutation is treated in the similar way).
By definition (6.7), we must show that

$\mathfrak{M} \models (\pi(w) \wedge w \succ \overline{0}) \supset ((\bigcup_{i=1}^{k} R'_i)^* \rightarrow \neg Imp((\neg b' \vee \bigwedge_{i=l+1}^{k} \neg b_i) \circ R'_1,\ldots,(\neg b' \vee \bigwedge_{i=l+1}^{k} \neg b_i) \circ R'_l))$
holds. This is a consequence of the following chain of implications:
$\mathfrak{M} \models (\pi(w) \wedge w \succ \overline{0})(\xi) \Rightarrow$

$\mathfrak{M} \models r \circ (\bigcup_{i=1}^{n} R_i)^*(\xi)$ (definition (7.5)) $\Rightarrow$

$\mathfrak{M} \models \exists \xi''[r(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'',\xi)]$ (section (5.11)) $\Rightarrow$

$\mathfrak{M} \models \exists \xi''[\neg FAIR(R_1,\ldots,R_n)(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'',\xi)]$ (assumption) $\Rightarrow$

$\mathfrak{M} \models \exists \xi''[((\bigcup_{i=1}^{n} R_i)^* \rightarrow \neg Imp(\bigwedge_{j=l+1}^{n} \neg b_j \circ R_1, \ldots, \bigwedge_{j=l+1}^{n} \neg b_j \circ R_l))(\xi'') \wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'',\xi)]$ (*.

(lemma (6.9))

Hence, for all $t=1,\ldots,l$

$$(\neg b' \vee \bigwedge_{i=l+1}^{k} \neg b_i) \circ R_t' =$$

$$(\neg b' \vee \bigwedge_{i=l+1}^{k} \neg b_i) \wedge b' \circ R_t =$$

$$(b' \wedge \bigwedge_{i=l+1}^{k} \neg b_i) \circ R_t =$$

$$(\bigwedge_{i=k+1}^{n} \neg b_i \wedge \bigwedge_{i=l+1}^{k} \neg b_i) \circ R_t =$$

$$\bigwedge_{i=l+1}^{n} \neg b_i \circ R_t \quad \text{(since } l+1 \leq k < n\text{)}.$$

So (*) implies that

$\mathfrak{M} \models \exists \xi''[((\bigcup_{i=1}^{n} R_i)^* \rightarrow \neg Imp((\neg b' \vee \bigwedge_{i=l+1}^{k} \neg b_i) \circ R_1', \ldots, (\neg b' \vee \bigwedge_{i=l+1}^{k} \neg b_i) \circ R_l'))(\xi'') \wedge$
$\wedge (\bigcup_{i=1}^{n} R_i)^*(\xi'',\xi)]$ and finally

$\mathfrak{M} \models ((\bigcup_{i=1}^{n} R_i')^* \rightarrow \neg Imp((\neg b' \vee \bigwedge_{i=l+1}^{k} \neg b_i) \circ R_1', \ldots, (\neg b' \vee \bigwedge_{i=l+1}^{k} \neg b_i) \circ R_l'))(\xi)$.

(using fact (5.3)).

7.10 COROLLARY (theorem(7.1))

$\mathfrak{M} \models [r]fair(*[\Box_{i=1}^{n} b_i \rightarrow S_i])[q] \Rightarrow Th(\mathfrak{M}) \vdash [r]fair(*[\Box_{i=1}^{n} b_i \rightarrow S_i])[q]$.

PROOF

From theorem(6.12), chapter 4, definition (7.4), definition (7.5), definition (7.6), lemma(7.7), lemma(7.8), lemma(7.9) and the following two observations:

(i) $Th(\mathfrak{M}) \vdash r \supset \exists v\ \pi(v)$. For let $\xi \in States$ satisfy $\mathfrak{M} \models r(\xi)$.

If $\mathfrak{M} \models \bigwedge_{i=1}^{n} \neg b_i(\xi)$, then we are done, because $\mathfrak{M} \models \pi(\bar{0})(\xi)$ holds.

Hence, let $\mathfrak{M} \models \bigvee_{i=1}^{n} b_i(\xi)$. (i)

That $\mathfrak{M} \models r \circ (\bigcup_{i=1}^{n} R_i)^*(\xi)$ holds, follows immediately. (ii)

Since, $\mathfrak{M} \models r(\xi)$, also $\mathfrak{M} \models \neg FAIR(R_1,\ldots,R_n)(\xi)$, and consequently

$\mathfrak{M} \models \neg Imp(R_1,\ldots R_n)(\xi)$. (i.e., $\models \tau^{\bar{\alpha}}(\underline{false})(\xi)$) (iii)

It follows from (i), (ii), (iii) that $\mathfrak{M} \models \pi(\bar{\alpha},<>)(\xi)$ holds.

(ii) $Th(\mathfrak{M})\vdash\pi(\bar{0})\supset((\bigwedge_{i=1}^{n}\neg b_i)\wedge q)$.

Note that actually we showed that

$$\mathfrak{M}\models_r\supset(\neg FAIR(R_1,\ldots,R_n)\wedge((\bigcup_{i=1}^{n}R_i)^*\circ\bigwedge_{i=1}^{n}\neg b_i)\rightarrow q)\Rightarrow$$

$$Th(\mathfrak{M})\vdash[r]fair(*[\Box_{i=1}^{n}b_i\rightarrow S_i])[\pi(\bar{0}]. \qquad (*)$$

$Th(\mathfrak{M})\vdash\pi(\bar{0})\supset\bigwedge_{i=1}^{n}\neg b_i$ follows immediately from definition (7.5).

As a consequence of the hypothesis,

$\mathfrak{M}\models_r\supset(((\bigcup_{i=1}^{n}R_i)^*\circ\bigwedge_{i=1}^{n}\neg b_i)\rightarrow q)$.

So, by (*) $\mathfrak{M}\models\pi(\bar{0})\supset q$. Now, $Th(\mathfrak{M})\vdash\pi(\bar{0})\supset q$ follows.

Chapter 8
SOUNDNESS

Soundness of Orna's rule amounts to the following

8.1 THEOREM
Let $\mathfrak{M}$ be a first-order acceptable structure. Then

$$Th(\mathfrak{M})\vdash[r]fair(S)[q]\Rightarrow\mathfrak{M}\models[r]fair(S)[q],\ \text{where } S\equiv*[\Box_{i=1}^{n}b_i\rightarrow S_i]\ (n\geq 1),$$

PROOF
Again, the non-trivial case is when $n\geq 2$.
Assume that $Th(\mathfrak{M})\vdash[r]S[q]$.
By theorem (6.12) it suffices to show that

$\mathfrak{M}\models_r\supset(\neg FAIR(R_1,\ldots,R_n)\wedge((\bigcup_{i=1}^{n}R_i)^*\circ\bigwedge_{i=1}^{n}\neg b_i)\rightarrow q)$ holds, where R_i is the relation associated with $b_i;S_i$ $(i=1,\ldots,n)$.

Let W and π be the well-founded set, respectively, the ranking function, that where used when applying Orna's rule.

8.2 LEMMA

Assume that $Th(\mathfrak{M})\vdash[r]fair(*[\Box_{i=1}^{n}b_i\rightarrow S_i])[q]$ holds.

Then $\mathfrak{M}\models_r\supset\neg Imp(R_1,\ldots,R_n)$ holds, too.

PROOF
Let $\mathfrak{M}\models_r(\xi)$ and suppose, to obtain a contradiction, that

$\mathfrak{M}\models Imp(R_1,\ldots,R_n)(\xi)$ holds. Since $D_w\neq\emptyset$ for $w>0$, there exists an infinite decreasing sequence in W, starting in some $w\in W$ such that $\mathfrak{M}\models\pi(w)(\xi)$ holds. This contradicts the well-foundedness of W.

8.3 LEMMA

Assume that $Th(\mathfrak{M})\vdash[r]fair(*[\overset{n}{\underset{i=1}{\square}} b_i \to S_i])[q]$ holds.

Let k be given, $1 \leq k < n$, and assume furthermore that $i_1,\ldots,i_n$ is some permutation of $1,\ldots,n$ $(n \geq 2)$.

Then $\mathfrak{M} \models r \supset \neg fair(R_{i_1},\ldots,R_{i_k})fin(R_{i_{k+1}},\ldots,R_{i_n})$ holds, too.

PROOF

Assume that $\mathfrak{M} \models r(\xi)$ holds for some ξ.

Possibly, after a renumbering, let $i_1,\ldots,i_n$ be the identity permutation of $1,\ldots,n$. Hence, we show that

$\mathfrak{M} \models \neg fair(R_1,\ldots,R_k)fin(R_{k+1},\ldots,R_n)(\xi)$ holds. According to definition (6.7), it suffices to prove the following

CLAIM

For all ξ' satisfying $\mathfrak{M} \models (\overset{n}{\underset{i=1}{\cup}} R_i)^*(\xi,\xi')$,

$\mathfrak{M} \models \neg Imp(b' \circ R_1,\ldots,b' \circ R_k)(\xi')$ holds, where $b' = \overset{n}{\underset{i=k+1}{\wedge}} \neg b_i$.

PROOF (of the claim)

Assume this is false. Both ξ and ξ' are accessible states; i.e., both $\mathfrak{M} \models r \circ (\overset{n}{\underset{i=1}{\cup}} R_i)^*(\xi)$ and $\mathfrak{M} \models r \circ (\overset{n}{\underset{i=1}{\cup}} R_i)^*(\xi')$ hold.

From our assumption that $\mathfrak{M} \models Imp(b' \circ R_1,\ldots,b' \circ R_k)(\xi')$ holds, we infer the existence of an infinite fair sequence of moves $b' \circ R_1,\ldots,b' \circ R_k$. As a consequence of our assumption that $Th(\mathfrak{M})\vdash[r]fair(*[\overset{n}{\underset{i=1}{\square}} b_i \to S_i])[q]$ holds, we conclude that Orna's rule has been applied. Consequently, related to this sequence of moves, is an infinite sequence $w_1,w_2,w_3,\ldots$ in W such that $w_i \geq w_{i+1}$ and $\mathfrak{M} \models \pi(w_1)(\xi')$. Since W is well-founded, eventually $w_i = w_{i+1}$.

This implies that none of the moves eventually taken, are decreasing moves.

Furthermore there is a state ξ'' such that

(a) $\mathfrak{M} \models (Imppart(b' \circ R_1,\ldots,b' \circ R_k))^*(\xi',\xi'')$,

(b) $\mathfrak{M} \models Imp(b' \circ R_1,\ldots,b' \circ R_k)(\xi'')$, and

(c) there is a w'' (not minimal) satisfying $w'' \leq w_1$, $\mathfrak{M} \models \pi(w'')(\xi'')$ and $\{1,\ldots,k\} \subseteq St_{w''}$.

Let $St_{w''} = \{j_1,\ldots,j_{k+m}\}$ for some $m \geq 0$, where $j_t = t$ for $t = 1,\ldots,k$.

(so $D_{w''} = \{j_{k+m+1},\ldots,j_n\}$.) Now, $w'' > 0$ and

$Th(\mathfrak{M})\vdash[\pi(w'')\wedge w''>0]fair(*[\ \Box\bigwedge_{i\in St_{w''}} b_i \wedge \bigwedge_{j\in D_{w''}} \neg b_j \to S_i])[\underline{true}]$ holds by the third clause of Orna's rule.
Hence, as a consequence of the inductionhypothesis and the fact that $\mathfrak{M}\models(\pi(w'')\wedge w>0)(\xi'')$, we obtain that

$$\mathfrak{M}\models\neg FAIR(\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\circ R_{j_1},\ldots,\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\circ R_{j_{k+m}})(\xi''), \qquad (i)$$

i.e., there is no infinite fair sequence of steady moves only in which no decreasing move is ever enabled.

There are two cases:
(A) $m=0$.

Then (i) implies that $\mathfrak{M}\models\neg Imp(b'\circ R_1,\ldots,b'\circ R_k)(\xi'')$ as $j_t=t$ for $1\leq t\leq k$ (using definition (6.8)), contradicting (b).

(B) $m\neq 0$.
Note that for all $s=j_{k+1},\ldots,j_{k+m}$ the actually enabling-condition for $\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\circ R_s$ is $\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\wedge b_s$.
By (i) and definition (6.8)

$$\mathfrak{M}\models\neg fair(\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\circ R_1,\ldots,\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\circ R_k)fin(\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\circ R_{j_{k+1}},\ldots\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\circ R_{j_{k+m}})(\xi'')$$

holds. So by definition (6.7)

$$\mathfrak{M}\models(\bigcup_{l=1}^{k+m}(\bigwedge_{t=k+m+1}^{n} \neg b_{j_t})\circ R_{j_l})^*\to\neg Imp(C\circ R_1,\ldots,C\circ R_k)(\xi'') \text{ holds, too,}$$

where $C=\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\wedge\bigwedge_{s=k+1}^{k+m}\neg(\bigwedge_{t=k+m+1}^{n} \neg b_{j_t}\wedge b_{j_s})$.
Hence, we obtain $\mathfrak{M}\models\neg Imp(C\circ R_1,\ldots,C\circ R_k)(\xi'')$.

As $\mathfrak{M}\models C=\bigwedge_{t=k+1}^{n} \neg b_t$, this implies

$$\mathfrak{M}\models\neg Imp(\bigwedge_{t=k+1}^{n} \neg b_t\circ R_1,\ldots,\bigwedge_{t=k+1}^{n} \neg b_t\circ R_k)(\xi''),$$

again contradicting (b).

This proves the claim and hence the theorem.

($\mathfrak{M}\models\neg\Box((\bigcup_{i=1}^{n} R_i)^*\circ\bigwedge_{i=1}^{n}\neg b_i)\to q)$ is trivial Under the aforementioned assumptions.)