Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

J.J.M.M. Rutten

Explicit canonical representatives for weak bisimulation equivalence and congruence

Computer Science/Department of Software Technology Report CS-R9062 November

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Research (N.W.O.).

Explicit Canonical Representatives for Weak Bisimulation Equivalence and Congruence

J.J.M.M. Rutten

Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

A labelled transition system is considered having as labels a set of actions containing τ, the silent move. Both for strong- and weak-bisimulation equivalence a model is defined that maps every state of the system to a canonical representative of its equivalence class. The latter model is obtained as an abstraction of the former. The main contribution is the fact that not only the *existence* of such representatives is established but that, moreover, an explicit (recursively defined) *description* is provided. The use of a (recursive) domain equation for the construction of tree-like structures, which are used for these representatives, is crucial.

1980 Mathematics Subject Classification: 68B10, 68C01.
1986 Computing Reviews Categories: D.3.1, F.3.2, F.3.3.
Key words and phrases: Structured operational semantics, labelled transition system, weak bisimulation, strong bisimulation, finite trees.
Note: This work was partially carried out in the context of ESPRIT Basic Research Action 3020 (Integration).

CONTENTS

1. INTRODUCTION

Let $\mathfrak{T} = (S, A, \rightarrow)$ be a finite *labelled transition system* (LTS), consisting of a set of *states* S, a set of *labels* (or *actions*) A and a *transition relation* $\rightarrow \subseteq S \times A \times S$. (See [Pl81].) Every LTS induces a *strong-bisimulation equivalence* ([Pa81]) on states. Intuitively two states are bisimilar if every step of the one can be simulated by a step of the other in such a way, that the respective resulting states are again bisimilar- and vice versa.

We shall assume that A contains a special element τ, the so-called *silent* or *internal move*. The notion of strong bisimulation does not discriminate between actions different from τ, which can be

Report CS-R9062
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

viewed as external, and the internal action τ. Therefore another equivalence has been introduced: *weak-bisimulation equivalence* (also called *observational equivalence* ([Mi83]) and *τ-bisimulation equivalence* ([BK85])). It is similar to the strong equivalence but it abstracts from τ actions: a step consists of an action, which is possibly preceded and followed by one or more τ actions.

An interesting notion concerning bisimulation equivalence is that of *canonical representatives* (or normal forms) for equivalence classes. The basic interest of canonical representatives is the fact that they can be used to decide whether two given states are bisimilar: first map each of them to the representative of its equivalence class and next compare these results. They can also be used in proving the completeness of certain axiom systems. The *existence* of such representatives is, amongst others, studied in [BK85], [Ca87] and [MSg89].

In [BK85], finite acyclic process graphs are used to represent computing agents or, in our terminology, states of labelled transition systems. Next a number of graph reduction procedures is defined, which are used to reduce such a graph step by step. It is shown that every sequence of graph reductions eventually terminates (basically because every reduction makes the graph smaller). Moreover it is proved that two graphs reduce to the same graph if and only if the states they represent are weakly bisimilar. Thus the existence of a canonical representative for each weak-bisimulation equivalence class is established.

In [Ca87] a similar result is given, again using graphs (called *non-deterministic systems*) and just one (somewhat more complicated) graph reduction procedure called *abstraction homomorphism*. Again the *existence* of a canonical representative for weak-bisimulation equivalence is established. In [MSg89], this approach is extended to the so-called *nondeterministic measurement systems* and *NMS* bisimulation, which allow a treatment of non-interleaving concurrency.

The result presented in this paper extends the above investigations in two ways. First, it uses finite *tree*-like structures satisfying a recursive *domain equation*. This offers the possibility to define a reduction procedure similar to the ones mentioned above in a recursive manner. Second, as an immediate consequence of this, not only the *existence* of a canonical representative is established; moreover an *explicit* (recursively defined) *description* is provided.

After some preliminary definitions in section 2, a model is defined (in section 3) that assigns to every state in the labelled transition system a canonical representative for its strong-bisimulation equivalence class. It is the tree-like structure obtained by unfolding the state according to the transition relation. (See also [GR89], [Ab90] and [Ru90a].) As a domain for these structures, the smallest set satisfying the following domain equation is taken:

$$P = \mathcal{P}_{fin}(A \times P)$$

The domain P is characterized by the fact that two elements in P, called *processes*, are equal if and only if they are strongly bisimilar (in a sense closely related to the notion of strong bisimilarity mentioned above). Then, in section 4, a representative for weak bisimulation is given by composing this model with a recursively defined abstraction operation. Because of the mathematical structure offered by the domain equation P, the definitions are very short and transparent. The abstraction operation consists of alternatingly applying two one-step reduction procedures that are also present (together with some other ones) in [BK85]: *pruning* and *lifting*. (In [BK85] they are called *arc reduction* and *removal of a deterministic τ-step*, respectively.) The proof of the fact that indeed canonical representations for weak bisimulation are obtained is more difficult than one would expect. Therefore a separate section (5) is dedicated to it. Only a minor variation in the definitions and proofs of sections 4 and 5 is needed to establish, in section 6, a similar result for the substitutive variant of weak bisimulation, *observational congruence*. (See [Mi83]; it is also known as *rooted τ-bisimulation* in [BK85]). Finally, in section 7, a number of possible extensions and future research are discussed.

Acknowledgements: Discussions with Ilaria Castellani and the members of the Amsterdam Concurrency Group have been very helpful.

2. Preliminary definitions

DEFINITION 2.1 (Action alphabet A): Let A be a (possibly infinite) set. The elements of A are to be thought of as atomic (uninterpreted) actions. The set A contains a special element τ, the so-called *silent move*. We shall use μ and ν to range over A, and a, b and c to range over $A-\{\tau\}$. The set ($w \in$) A^* of finite words over A is provided with the following equivalence: two words w_1 and w_2 are equivalent (notation $w_1 \equiv w_2$) if and only if there exists $\mu \in A$ such that both w_1 and w_2 are in $\{\tau\}^*\{\mu\}\{\tau\}^*$. (E.g., $\tau\tau a\tau \equiv a \equiv a\tau\tau$.)

DEFINITION 2.2 (LTS): We introduce a fixed *labelled transition system* $\mathcal{T} = (S, A, \rightarrow)$, consisting of a set of *states* S, a set of *labels* A (which is the action alphabet introduced above), and a *transition relation* $\rightarrow \subseteq S \times A \times S$. We shall write $s \xrightarrow{\mu} s'$ for $(s, \mu, s') \in \rightarrow$ and say that s can go to s' by performing an a step. The LTS $\mathcal{T}$ is *finite* in the following sense. For all $s \in S$ the set of transition sequences

$$s \xrightarrow{\mu_1} s_1 \xrightarrow{\mu_2} \cdots \xrightarrow{\mu_{n-1}} s_{n-1} \xrightarrow{\mu_n} s_n$$

is finite. For $s \in S$ the depth of s (notation: $d(s)$) is defined as the length of the longest transition sequence starting in s.

The following abbreviations will be used:

$$\Rightarrow^a \; = \; \xrightarrow{\tau^n} \xrightarrow{a} \xrightarrow{\tau^m} \quad n, m \geqslant 0$$

$$\Rightarrow^\tau \; = \; \xrightarrow{\tau^n} \quad n \geqslant 0$$

$$\Rightarrow^\mu_+ \; = \; \begin{cases} \Rightarrow^a & \text{if } \mu = a \\ \xrightarrow{\tau^n} \; (n>0) & \text{if } \mu = \tau \end{cases}$$

Here $\xrightarrow{\mu}$ is a binary relation on states defined by

$$\xrightarrow{\mu} \; = \; \{(s, t): \; s, t \in S \wedge s \xrightarrow{\mu} t\}$$

(End of definition.)

Often, the set S of states consists of all closed terms over some signature Σ. Here we want to abstract from any specific choice for Σ. However, in order to be able to give some examples later, we assume a minimum of structure on our transition system $\mathcal{T}$. We stipulate the presence of a special state $nil \in S$, from which no transitions are possible. Further there are two binary operators $\cdot$ and $+$ on S, representing sequential composition and non-deterministic choice (in the style of CCS ([Mi80])). We assume $\rightarrow$ to satisfy the following axiom and rules:

$$\mu \xrightarrow{\mu} nil$$

$$\text{if } s \xrightarrow{\mu} s' \text{ then } s+t \xrightarrow{\mu} s' \text{ and } t+s \xrightarrow{\mu} s'$$

$$\text{if } s \xrightarrow{\mu} s' \text{ and } s' \neq nil \text{ then } s \cdot t \xrightarrow{\mu} s' \cdot t$$

$$\text{if } s \xrightarrow{\mu} nil \text{ then } s \cdot t \xrightarrow{\mu} t$$

Often we shall omit *nil* if it occurs at the end of a state expression; e.g., we write $a \cdot b + c$ rather than $a \cdot b \cdot nil + c \cdot nil$.

Next a set of finite tree-like structures, which are called *processes*, is introduced.

DEFINITION 2.3 (Processes): The set $(p, q, r \in)$ P of *processes* is defined as the smallest set satisfying the following recursive domain equation:

$$P = \mathcal{P}_{fin}(A \times P)$$

Here $\mathcal{P}_{fin}(A \times P)$ contains all finite subsets of $A \times P$, and A is the action alphabet. The set P can be obtained as the union of a sequence $(P_n)_n$, thus $P = \bigcup_n P_n$, where for every $n \geqslant 0$, the set P_n is inductively defined by

$$P_0 = \emptyset$$

$$P_{n+1} = \mathcal{P}_{fin}(A \times P_n)$$

For $p \in P$ the depth of p (notation: $d(p)$) is defined as the length of the longest branch in p.
(End of definition.)

A process $p \in P$ is a finite set of elements $<\mu, p'>$, with $\mu \in A$ and $p' \in P$. Each such pair represents a possible step, μ, of p followed by a process p' indicating all possible steps that can be taken after μ.

DEFINITION 2.4: We introduce a transition relation on processes similar to the transition relation on states. Let $\rightarrow \subseteq P \times A \times P$ be defined by

$$p \xrightarrow{\mu} p' \text{ if and only if } <\mu, p'> \in p$$

(Again we write $p \xrightarrow{\mu} p'$ for $(p, \mu, p') \in \rightarrow$.) We call μ a *branch* in p leading to the *node* p'. We define the double arrow between processes, i.e.,

$$p \Rightarrow^a p', \quad p \Rightarrow^\tau p', \quad p \Rightarrow^a_+ p', \quad p \Rightarrow^\tau_+ p'$$

in precisely the same manner as in the case of transitions between states above. (We are confident that the use of the same symbols $\rightarrow$ and $\Rightarrow$ for transitions between both states and processes will not cause any confusion.)
(End of definition.)

3. Strong bisimulation

The LTS $\mathcal{T} = (S, A, \rightarrow)$ induces the notion of strong bisimulation equivalence ([Pa81]) as follows.

Definition 3.1 (Strong bisimulation): A relation $R \subseteq S \times S$ is called a *strong bisimulation* if it is symmetric and for all $s, t \in S$ and $\mu \in A$:

$$\text{if } sRt \text{ and } s \xrightarrow{\mu} s' \text{ then } \exists t' \in S\ [t \xrightarrow{\mu} t' \wedge s'Rt']$$

Two states are *strongly bisimilar,* notation $s =_{sb} t$, if there exists a strong bisimulation relation R with sRt. (Note that bisimilarity is an equivalence relation on states.)

Intuitively two states are bisimilar if every step of the one can be simulated by a step of the other in such a way, that the respective resulting states are again bisimilar- and vice versa.

The following model assigns to a state $s \in S$ its tree-like unfolding as specified by the transition relation.

Definition 3.2 ($\mathcal{M}_{sb}$): The model $\mathcal{M}_{sb}$: $S \rightarrow P$ is defined, using induction on the depth $d(s)$ of s (Definition 2.2), by

$$\mathcal{M}_{sb}[\![s]\!] = \{<\mu, \mathcal{M}_{sb}[\![s']\!]>: s \xrightarrow{\mu} s'\}$$

The above model assigns the same value to states that are strongly bisimilar. In other words, $\mathcal{M}_{sb}[\![s]\!]$ can be viewed as a canonical representative for the equivalence class of s of the relation of strong bisimilarity. This is the content of the following.

Theorem 3.3: *For all s, $t \in S$,*

$$\mathcal{M}_{sb}[\![s]\!] = \mathcal{M}_{sb}[\![t]\!] \Leftrightarrow s =_{sb} t$$

This theorem, which will be proved below, is a direct consequence of the more general fact that processes are strongly bisimilar if and only if they are equal. Here the notion of bisimilarity of processes is a straightforward generalisation of the one above. Formally, it is defined as follows.

DEFINITION 3.4 (Strong bisimulation for processes): A relation $R \subseteq P \times P$ is called a *strong process bisimulation* if it is symmetric and for all $p,q \in P$ and $\mu \in A$:

$$\text{if } pRq \text{ and } p \xrightarrow{\mu} p' \text{ then } \exists q' \in P\ [q \xrightarrow{\mu} q' \wedge p'Rq']$$

Two process are *strongly bisimilar,* notation $p =_{sb} q$, if there exists a strong bisimulation relation R with pRq. (Recall that $p \xrightarrow{\mu} p'$ is defined as $<\mu, p'> \in p$.)

There is the obvious relation between the two notions of strong bisimilarity.

LEMMA 3.5: *For all s, $t \in S$,*

$$\mathfrak{M}_{sb}[\![s]\!] =_{sb} \mathfrak{M}_{sb}[\![t]\!] \Leftrightarrow s =_{sb} t$$

We have the following theorem, which says that on P, the notions of equality and strong process bisimilarity coincide. This property is sometimes expressed by the phrase that P is *strongly extensional.* (See, e.g., [Ac88]).

THEOREM 3.6: *For all p, $q \in P$,*

$$p = q \Leftrightarrow p =_{sb} q$$

PROOF:
$\Rightarrow$: Trivial.
$\Leftarrow$: We proceed by induction on $\gamma(p, q) = \max\{d(p), d(q)\}$. Suppose $p =_{sb} q$ and suppose we have for all p' and q' with $\gamma(p', q') < \gamma(p, q)$ that $p' =_{sb} q'$ entails $p' = q'$. We prove that $p = q$.

Let $p \xrightarrow{\mu} p'$. Since $p =_{sb} q$ there exists $q' =_{sb} p'$ with $q \xrightarrow{\mu} q'$. Since $\gamma(p', q') < \gamma(p, q)$ we have $p' = q'$. Thus $q \xrightarrow{\mu} p'$. This proofs $p \subseteq q$. Similarly one can show $q \subseteq p$ and thus we may conclude $p = q$.

Now the proof of Theorem 3.3 above is immediate:

PROOF OF THEOREM 3.3: For all s, $t \in S$,

$$s =_{sb} t \Leftrightarrow \mathfrak{M}_{sb}[\![s]\!] =_{sb} \mathfrak{M}_{sb}[\![t]\!]$$

$$\Leftrightarrow \mathfrak{M}_{sb}[\![s]\!] = \mathfrak{M}_{sb}[\![t]\!]$$

A proof for the more general case of infinite (and possibly non-image-finite) transition systems can be found in [GR89] and [Ru90a]. In [Ab90], a similar result is given that also covers divergence.

Note that if we interpret the $+$ by set-theoretic union ($\cup$) and the state *nil* by the empty process ($\emptyset$), the process domain P satisfies the well-known absorption law and sum laws: For all $p, q, r \in P$,

$$p + p = p$$

$$p + q = q + p$$

$$p + (q + r) = (p + q) + r$$

$$p + nil = p$$

4. WEAK BISIMULATION

The notion of strong bisimilarity does not discriminate between actions that are internal, modelled by τ, and actions that are different from τ. E.g., we have $\mathcal{M}_{sb}[\![a \cdot \tau]\!] \neq \mathcal{M}_{sb}[\![a]\!]$:

$$\mathcal{M}_{sb}[\![a]\!] = \{<a, \emptyset>\}$$

$$\mathcal{M}_{sb}[\![a \cdot \tau]\!] = \{<a, \{<\tau, \emptyset>\}>\}$$

In a sense, this does not do justice to the intuition that τ is an *internal* action, which should be *invisible*. The notion of weak bisimulation equivalence was invented to overcome this objection. It is more abstract than strong bisimilarity in that it identifies more states (like the two just mentioned).

DEFINITION 4.1 (Weak bisimulation): A relation $R \subseteq S \times S$ is called a *weak bisimulation* if it is symmetric and for all $s, t \in S$ and $\mu \in A$:

$$\text{if } sRt \text{ and } s \Rightarrow^{\mu} s' \text{ then } \exists t' \in S\, [t \Rightarrow^{\mu} t' \wedge s'Rt']$$

Two states are *weakly bisimilar,* notation $s =_{wb} t$, if there exists a weak bisimulation relation R with sRt.

The above notion is also called *observational equivalence* ([Mi83]) or *τ-bisimulation* ([BK85]).

Next, we would like to characterize, similarly to Theorem 3.3 above, the equivalence classes of weak bisimilarity. A first naive attempt, which does not work, might be to define a model $\mathcal{M}_{wb}: S \rightarrow P$ by

$$\mathcal{M}_{wb}[\![s]\!] = \{<\mu, \mathcal{M}_{wb}[\![s']\!]>: s \Rightarrow^{\mu}_{+} s'\}$$

The reason why this model fails to characterize weak bisimilarity is illustrated by a very simple example. The states a and $a \cdot \tau$ are weakly bisimilar, but have different meanings under $\mathcal{M}_{wb}$:

$$\mathcal{M}_{wb}[\![a]\!] = \{<a, \emptyset>\}$$

$$\mathcal{M}_{wb}[\![a\cdot\tau]\!] = \{<a, \varnothing>, <a, \{<\tau, \varnothing>\}>\}$$

Therefore, a different approach is chosen. As a starting point, it takes the process representation for strong bisimulation equivalence as given by $\mathcal{M}_{sb}$. Then so-called *pruning* and *lifting* operations are recursively performed.

DEFINITION 4.2 (Pruning and lifting): On processes the operations of *pruning* and *lifting* are defined as follows. Let π, λ: $P \rightarrow P$ be given, for all $p \in P$, by

$$\pi(p) = \{<\mu, p'>: p \xrightarrow{\mu} p' \wedge \neg(p - \{<\mu, p'>\} \Rightarrow^{\mu}_{+} p')\}$$

$$\lambda(p) = \begin{cases} p' & \text{if } p = \{<\tau, p'>\} \\ p & \text{otherwise} \end{cases}$$

The lifting operator λ (ruthlessly) removes brotherless initial τ steps. The pruning operator π cuts those branches $<\mu, p'>$ from a given process p whenever the process p' can also be reached inside p via another branch, possibly via one or more extra silent moves. In the definition, this is formally expressed by

$$p - \{<\mu, p'>\} \Rightarrow^{\mu}_{+} p'$$

A simple example is

$$\pi(\{<\tau, \{<a, \varnothing>\}>, <a, \varnothing>\}) = \{<\tau, \{<a, \varnothing>\}>\}$$

Next an abstraction operator on processes α is defined, which consists of alternatingly applying the operations of pruning and lifting.

DEFINITION 4.3 (Abstraction α): Let α: $P \rightarrow P$ be defined, for all $p \in P$, by

$$\alpha(p) = \lambda \circ \pi(\{<\mu, \alpha(p')>: <\mu, p'> \in p\})$$

Intuitively, applying α to a process p amounts to the following. First α is applied to all the subnodes p' (with $<\mu, p'> \in p$) of p. Next the pruning operation is applied to the resulting process, followed by the lifting operation. Computing $\alpha(p)$ thus consists of alternatingly applying π and λ to all the nodes of p, starting with the very lowest ones, and then continuing with the nodes above, until the top node (p itself) is reached.

Now a model for weak bisimulation is obtained by composing $\mathcal{M}_{sb}$ with α.

DEFINITION 4.4 ($\mathcal{M}_{wb}$): The model $\mathcal{M}_{wb}$: $S \rightarrow P$ is defined, for all $s \in S$, by

$$\mathcal{M}_{wb}[\![s]\!] = \alpha(\mathcal{M}_{sb}[\![s]\!])$$

We can define $\mathcal{M}_{wb}$ independently of $\mathcal{M}_{sb}$ as follows.

DEFINITION 4.5 (Alternative definition of $\mathcal{M}_{wb}$): Let $\mathcal{M}_{wb}$: $S \rightarrow P$ be defined, for all $s \in S$, by

$$\mathcal{M}_{wb}[\![s]\!] = \lambda \circ \pi(\{<\mu, \mathcal{M}_{wb}[\![s']\!]>: s \overset{\mu}{\Rightarrow} s'\})$$

THEOREM 4.6: *The two definitions of* $\mathcal{M}_{wb}$ *are equivalent.*

Now we come to the main result of this paper. It says that $\mathcal{M}_{wb}$ assigns the same value to states if and only if they are weakly bisimilar. Therefore we can view $\mathcal{M}_{wb}[\![s]\!]$ as a canonical representative for the weak-bisimulation-equivalence class of the state $s \in S$.

THEOREM 4.7: *For all* $s, t \in S$,

$$\mathcal{M}_{wb}[\![s]\!] = \mathcal{M}_{wb}[\![t]\!] \Leftrightarrow s =_{wb} t$$

Reading the theorem from left to right gives the soundness of our model $\mathcal{M}_{wb}$: if $\mathcal{M}_{wb}$ maps two states onto the same value, then they must be weakly bisimilar. This will be fairly easy to prove. The arrow from right to left could be called the completeness part of the theorem: all weakly bisimilar states will be mapped onto the same value. Although intuitively quite clear, this will be more difficult to prove. In all, the theorem tells us that $\mathcal{M}_{wb}$ maps every state to a processes that can be viewed as a canonical representative of its weak-bisimulation-equivalence class.

Before we formally proof this theorem in the next section, let us first try to give the reader some confidence in its truth by looking at a few examples.

It is not difficult to see that the following two states are weakly bisimilar: $\tau \cdot a + a =_{wb} \tau \cdot a$. The reader will probably recognize this as an instantiation of Milner's second τ-law. According to the above theorem, they should have the same meaning under $\mathcal{M}_{wb}$. Fortunately, they do:

$$\begin{aligned}
\mathcal{M}_{wb}[\![\tau \cdot a + a]\!] &= \alpha(\mathcal{M}_{sb}[\![\tau \cdot a + a]\!]) \\
&= \lambda \circ \pi(\{<\tau, \{<a, \varnothing>\}>, <a, \varnothing>\}) \\
&= \lambda(\{<\tau, \{<a, \varnothing>\}>\}) \\
&= \lambda \circ \pi(\{<\tau, \{<a, \varnothing>\}>\}) \\
&= \alpha(\mathcal{M}_{sb}[\![\tau \cdot a]\!]) \\
&= \mathcal{M}_{wb}[\![\tau \cdot a]\!]
\end{aligned}$$

The following two states are weakly bisimilar, as can be readily seen using Milner's third τ-law: $(a \cdot (b + \tau \cdot c) + a \cdot c) =_{wb} (a \cdot (b + \tau \cdot c))$. Again, we have

$$\mathfrak{M}_{wb}[\![a\cdot(b+\tau\cdot c)+a\cdot c]\!] = \alpha(\mathfrak{M}_{sb}[\![a\cdot(b+\tau\cdot c)+a\cdot c]\!])$$
$$= \lambda\circ\pi(\{<a,\{<b,\varnothing>,<\tau,\{<c,\varnothing>\}>\}>,\ \ <a,\{<c,\varnothing>\}>\})$$
$$= \lambda(\{<a,\{<b,\varnothing>,<\tau,\{<c,\varnothing>\}>\}>\})$$
$$= \{<a,\{<b,\varnothing>,<\tau,\{<c,\varnothing>\}>\}>\}$$
$$= \mathfrak{M}_{wb}[\![a\cdot(b+\tau\cdot c)]\!]$$

5. Proving the main theorem

The goal of this section is to prove the following theorem, from which Theorem 4.7 of the previous section follows immediately. In fact it is a more general version of Theorem 4.7 and therefore is called the Main Theorem.

MAIN THEOREM 5.1: *For all $p, q \in P$,*

$$\alpha(p) = \alpha(q) \Leftrightarrow p =_{wb} q$$

Here the notion of weak bisimulation is straightforwardly generalised to processes. (Recall that the same was done for the notion of strong bisimulation, see Definition 3.4.) It is formally defined as follows.

DEFINITION 5.2 (Weak bisimulation for processes): A relation $R \subseteq P \times P$ is called a *weak process bisimulation* if it is symmetric and for all $p,q \in P$ and $\mu \in A$:

$$\text{if } pRq \text{ and } p \Rightarrow^{\mu} p' \text{ then } \exists q' \in P\ [q \Rightarrow^{\mu} q' \wedge p'Rq']$$

Two processes are *weakly bisimilar,* notation $p =_{wb} q$, if there exists a weak process bisimulation relation R with pRq.

The following fact is immediate.

LEMMA 5.3: *For all $s, t \in S$,*

$$s =_{wb} t \Leftrightarrow \mathfrak{M}_{sb}[\![s]\!] =_{wb} \mathfrak{M}_{sb}[\![t]\!]$$

Now the Theorem 4.7 follows immediately:

PROOF OF THEOREM 4.7: For all $s, t \in S$,

$$\mathcal{M}_{wb}[\![s]\!] = \mathcal{M}_{wb}[\![t]\!] \Leftrightarrow \alpha(\mathcal{M}_{sb}[\![s]\!]) = \alpha(\mathcal{M}_{sb}[\![t]\!])$$

$$\Leftrightarrow [\text{ Theorem 5.1 }]$$

$$\mathcal{M}_{sb}[\![s]\!] =_{wb} \mathcal{M}_{sb}[\![t]\!]$$

$$\Leftrightarrow [\text{ Lemma 5.3 }]$$

$$s =_{wb} t$$

Before we prove Theorem 5.1, we first collect a number of basic properties of π and α.

LEMMA 5.4: *For all* $p \in P$,

$$\text{if } p \xrightarrow{\mu} p' \text{ then } \pi(p) \Rightarrow^{\mu}_{+} p'$$

PROOF: Let $p \xrightarrow{\mu} p'$. If $<\mu, p'> \notin \pi(p)$ then the definition of π implies

$$p - \{<\mu, p'>\} \Rightarrow^{\mu}_{+} p'$$

Since $\neg(p - \{<\mu, p'>\} \xrightarrow{\mu} p')$ this implies

$$p - \{<\mu, p'>\} \xrightarrow{\tau} p_1 \Rightarrow^{\mu}_{+} p'$$

for some p_1 with $<\tau, p_1> \in p$. Note that $d(p_1) > d(p')$.

Suppose, again, that $<\tau, p_1> \notin \pi(p)$. We repeat the above argument. Proceeding in this way, a sequence $p_1, \ldots, p_n$ is constructed satisfying

$$p - \{<\tau, p_{n-1}>\} \xrightarrow{\tau} p_n \Rightarrow^{\tau}_{+} p_{n-1} \Rightarrow^{\tau}_{+} \cdots \Rightarrow^{\tau}_{+} p_1 \Rightarrow^{\mu}_{+} p'$$

and $d(p') < d(p_1) < \cdots < d(p_n)$. Since $d(p_n) \leqslant d(p)$ this construction must at a certain moment end. That is, either $<\mu, p'> \in \pi(p)$ or there exists $n \geqslant 1$ such that $<\tau, p_n> \in \pi(p)$. In both cases we are ready since the latter implies

$$\pi(p) \xrightarrow{\tau} p_n \Rightarrow^{\mu}_{+} p'$$

and hence $\pi(p) \Rightarrow^{\mu}_{+} p'$. (End of proof.)

The fact that we consider only *finite* processes is essential, as is apparent from the proof. If one would consider an infinite process of the form

$$p = \{<a, \emptyset>,$$

$$<\tau, \{<a, \emptyset>\}>,$$

$$<\tau, \{<\tau, \{<a, \emptyset>\}>\}>, \cdots \}$$

then the above theorem does not apply: $\alpha(p) = \emptyset$.

As a convenient shorthand, we introduce the following.

DEFINITION 5.5 (V): The function $V: P \rightarrow P$ is defined, for all $p \in P$, by

$$V(p) = \{<\mu, \alpha(p')>: <\mu, p'> \in p\}$$

Note that for all $p \in P$, $\alpha(p) = \lambda \circ \pi(V)$.

The next lemma states that when one computes $\alpha(p)$ for a given $p \in P$, one can find a process p' with $\alpha(p) = \alpha(p')$ such that the application of α to p' does not involve lambda lifting at the outermost level.

LEMMA 5.6: *For all $p \in P$ there exists $p' \in P$ with $p \Rightarrow^{\tau} p'$ such that*

$$\alpha(p) = \alpha(p') \text{ and } \alpha(p') = \pi(V(p'))$$

PROOF: We proceed by induction on the depth $d(p)$ of p. Let $p \in P$ and suppose the theorem holds for all p' with $d(p') < d(p)$. We prove that the lemma holds for p. If $\alpha(p) = \pi(V(p))$ then we can take $p' = p$. Now suppose $\alpha(p) \neq \pi(V(p))$. Then $\pi(V(p)) = \{<\tau, \alpha(p')>\}$, for some $p' \in P$ with $<\tau, p'> \in p$, and $\alpha(p) = \lambda \circ \pi(V(p)) = \alpha(p')$. Since $<\tau, p'> \in p$, we have $d(p') < d(p)$. By induction there exists p'' with $p' \Rightarrow^{\tau} p''$, $\alpha(p'') = \pi(V(p''))$ and $\alpha(p') = \alpha(p'')$. Thus $\alpha(p) = \alpha(p') = \alpha(p'')$ and $p \Rightarrow^{\tau} p''$. (End of proof.)

The lemma below is a necessary condition for the soundness of our Main Theorem.

LEMMA 5.7: $\alpha \circ \alpha = \alpha$

PROOF: Let $p \in P$. We use induction on $d(\alpha(p))$. Suppose the theorem holds for all p' with $d(\alpha(p')) < d(\alpha(p))$. We prove : $\alpha(\alpha(p)) = \alpha(p)$. By Lemma 5.6 we may assume that $\alpha(p) = \pi(V(p))$. Because $d(\alpha(p')) < d(\alpha(p))$ for all p' with $<\mu, \alpha(p')> \in V(p)$, we have $\alpha(\alpha(p')) = \alpha(p')$. The same holds for all p' with $<\mu, \alpha(p')> \in \pi(V(p))$, because $\pi(V(p)) \subseteq V(p)$. Therefore the following equalities hold:

$$\begin{aligned}
\alpha(\alpha(p)) &= \alpha(\pi(V(p))) \\
&= \lambda \circ \pi(\{<\mu, \alpha(\alpha(p'))>: <\mu, \alpha(p')> \in \pi(V(p))\}) \\
&= [\text{ induction }] \\
&\quad \lambda \circ \pi(\{<\mu, \alpha(p')>: <\mu, \alpha(p')> \in \pi(V(p))\}) \\
&= \lambda \circ \pi(\pi(V(p))
\end{aligned}$$

$$= [\ \pi\circ\pi\ =\ \pi\ ]$$

$$\lambda\circ\pi(V(p))$$

$$= \alpha(p)$$

(End of proof.)

The next two lemma's and theorem establish the soundness of the Main Theorem. This will be stated as a corollary.

LEMMA 5.8: *For all* $p, p' \in P$ *and* $\mu \in A$,

$$\text{if } p \Rightarrow^{\mu} p' \text{ then } \alpha(p) \Rightarrow^{\mu} \alpha(p')$$

PROOF: Suppose $p \Rightarrow^{\mu} p'$. If $p = p'$ then we are ready. So suppose $p \Rightarrow^{\mu}_{+} p'$. Let us first consider the case that $p \xrightarrow{\mu} p'$. By the definition of V we have $V(p) \xrightarrow{\mu} \alpha(p')$. Applying Lemma 5.4 yields

$$\pi(V(p)) \Rightarrow^{\mu}_{+} \alpha(p')$$

If $\alpha(p) = \pi(V(p))$ then we are ready. If, on the other hand, $\pi(V(p)) = \{<\tau, \alpha(q)>\}$ for some q with $<\tau, q> \in p$, then

$$\alpha(p) = \lambda(\pi(V(p)))$$

$$= \lambda(\{<\tau, \alpha(q)>\})$$

$$= \alpha(q)$$

Since $\{<\tau, \alpha(q)>\} \Rightarrow^{\mu}_{+} \alpha(p')$ we have

$$\{<\tau, \alpha(q)>\} \xrightarrow{\tau} \alpha(q) \Rightarrow^{\mu} \alpha(p')$$

Because $\alpha(p) = \alpha(q)$ this implies $\alpha(p) \Rightarrow^{\mu} \alpha(p')$.

Next consider

$$p \xrightarrow{\mu_1} p_1 \xrightarrow{\mu_2} \cdots \xrightarrow{\mu_n} p_n = p'$$

with $\mu_1 \cdots \mu_n \equiv \mu$. The above implies

$$\alpha(p) \Rightarrow^{\mu_1} \alpha(p_1) \Rightarrow^{\mu_2} \cdots \Rightarrow^{\mu_n} \alpha(p_n) = \alpha(p')$$

and thus $\alpha(p) \Rightarrow^{\mu} \alpha(p')$. (End of proof.)

LEMMA 5.9: *For all* $p, p' \in P$ *and* $\mu \in A$,

$$\text{if } \alpha(p) \Rightarrow^{\mu} p' \text{ then } \exists q \in P\ [p \Rightarrow^{\mu} q \wedge \alpha(q) = p']$$

PROOF: Let $\alpha(p) \Rightarrow^{\mu} p'$. If $\alpha(p) = p'$ then we are ready. So suppose $\alpha(p) \Rightarrow^{\mu}_{+} p'$. First we consider $\alpha(p) \xrightarrow{\mu} p'$. By Lemma 5.6 there exists $q \in P$ with

$$p \Rightarrow^{\tau} q, \quad \alpha(p) = \alpha(q), \quad \alpha(q) = \pi(V(q))$$

Since $<\mu, p'> \in \alpha(p) = \alpha(q) = \pi(V(q)) \subseteq V(q)$ there exists by the definition of V a process q' with $q \xrightarrow{\mu} q'$ and $\alpha(q') = p'$. Thus $p \Rightarrow^{\mu} q'$ and $\alpha(q') = p'$.

Next consider

$$\alpha(p) \xrightarrow{\mu_1} p_1 \xrightarrow{\mu_2} \cdots \xrightarrow{\mu_n} p_n = p'$$

with $\mu_1 \cdots \mu_n \equiv \mu$. Since $\alpha(p) \xrightarrow{\mu_1} p_1$ there exists by the above a process q_1 such that $p \Rightarrow^{\mu_1} q_1$ and $\alpha(q_1) = p_1$. Because $\alpha(q_1) = p_1 \xrightarrow{\mu_2} p_2$ there exists, again by the above, a process q_2 satisfying $q_1 \Rightarrow^{\mu_2} q_2$ and $\alpha(q_2) = p_2$. Continuing in this way we find $q_1, \ldots, q_n$ such that

$$p \Rightarrow^{\mu_1} q_1 \Rightarrow^{\mu_2} \cdots \Rightarrow^{\mu_n} q_n \text{ and } \alpha(q_n) = p_n = p'$$

Thus $p \Rightarrow^{\mu} q_n$ and $\alpha(q_n) = p'$. (End of proof.)

THEOREM 5.10: *For all* $p \in P$,

$$\alpha(p) =_{wb} p$$

PROOF: We show that the relation $R \subseteq P \times P$, defined by

$$R = \{(p, \alpha(p)):\ p \in P\} \cup \{(\alpha(p), p):\ p \in P\}$$

is a weak process bisimulation. Let $p \in P$ with $pR\,\alpha(p)$. (The case that $\alpha(p)Rp$ is symmetric.) Suppose $p \Rightarrow^{\mu} p'$. Then also $\alpha(p) \Rightarrow^{\mu} \alpha(p')$, by Lemma 5.8. Note that $p'R\,\alpha(p')$.

Next suppose that $\alpha(p) \Rightarrow^{\mu} p'$. By Lemma 5.9 there exists $q \in P$ with $p \Rightarrow^{\mu} q$ and $\alpha(q) = p'$. Note that $p'R\,q$ since $p' = \alpha(q)$. (End of proof.)

The soundness of the Main Theorem is now immediate.

COROLLARY 5.11 (Soundness): *For all* $p, q \in P$,

$$\text{if } \alpha(p) = \alpha(q) \text{ then } p =_{wb} q$$

PROOF: Suppose $\alpha(p) = \alpha(q)$. Since $\alpha(p) =_{wb} p$ and $\alpha(q) =_{wb} q$, by Theorem 5.10, this implies $p =_{wb} q$.

The next lemma states that all nodes occurring in $\alpha(p)$, for any $p \in P$, are invariant under application of α. (For the top node $\alpha(p)$ itself this is immediate from Lemma 5.7, stating that $\alpha \circ \alpha = \alpha$.)

LEMMA 5.12: *For all* $p, q \in P$, $n \geq 0$ *and* $\mu_1, \ldots, \mu_n$ *in* A,

$$\text{if } \alpha(p) \Rightarrow^{\mu_1} \cdots \Rightarrow^{\mu_n} q \quad \text{then } \alpha(q) = q$$

PROOF: The proof uses induction on n. If $\alpha(p) = q$, i.e., $n = 0$, then $\alpha(q) = \alpha(\alpha(p)) = \alpha(p)$ (by Lemma 5.7) $= q$. Next suppose $n \geqslant 1$. Suppose

$$\alpha(p) \Rightarrow^{\mu_1} \cdots \Rightarrow^{\mu_{n-1}} q'$$

for some $\mu_1, \ldots, \mu_{n-1}$, and q' with $q' \Rightarrow^{\mu_n} q$ for some $\mu_n \in A$. By induction we have $\alpha(q') = q'$. Thus $\alpha(q') \Rightarrow^{\mu_n} q$. By Lemma 5.9 there exists a process q'' with $q' \Rightarrow^{\mu_n} q''$ and $\alpha(q'') = q$. Applying Lemma 5.7 yields

$$\alpha(q) = \alpha(\alpha(q'')) = \alpha(q'') = q$$

(End of proof.)

Finally we come to the completeness part of the Main Theorem.

THEOREM 5.13 (Completeness): *For all* $p, q \in P$,

$$\text{if } \alpha(p) =_{wb} \alpha(q) \text{ then } \alpha(p) = \alpha(q)$$

PROOF: We use induction on

$$\omega(p, q) = \max\{d(\alpha(p)), d(\alpha(q))\}$$

Let $p, q \in P$ and suppose the theorem holds for all $p', q' \in P$ with $\omega(p', q') < \omega(p, q)$. Suppose $\alpha(p) =_{wb} \alpha(q)$. We show $\alpha(p) = \alpha(q)$ by proving $\alpha(p) \subseteq \alpha(q)$ and $\alpha(q) \subseteq \alpha(p)$. It will be convenient to assume that $\alpha(p) = \pi(V(p))$ and $\alpha(q) = \pi(V(q))$; note that we can do so without loss of generality by Lemma 5.6.

Suppose that for some $\mu \in A$ and $p' \in P$ we have

$$\alpha(p) \xrightarrow{\mu} p'$$

We set out to show that also $\alpha(q) \xrightarrow{\mu} p'$. Because $\alpha(p) =_{wb} \alpha(q)$ there exists $q' =_{wb} p'$ with

$$\alpha(q) \Rightarrow^{\mu} q'$$

Suppose $\alpha(q) = q'$. Then $\alpha(p) =_{wb} \alpha(q) = q' =_{wb} p'$, hence $\alpha(p) =_{wb} p'$. We derive a contradiction.

First note that $\mu = \tau$, since otherwise an infinite transition sequence would be derivable from $\alpha(p)$. Next consider an arbitrary transition $\alpha(p) \xrightarrow{\nu} r$, for $r \in P$ and $\nu \in A$. Because $\alpha(p) =_{wb} p'$ there exists $r' =_{wb} r$ with $p' \Rightarrow^{\nu} r'$. Since $\omega(r, r') < \omega(p, q)$ and $\alpha(r) = r$ and $\alpha(r') = r'$, by Lemma 5.12, we have by induction $\alpha(r) = \alpha(r')$, hence $r = r'$. It must be the case that $\langle \nu, r \rangle = \langle \tau, p' \rangle$ because if they are different, then

$$V(p) - \{\langle \nu, r \rangle\} \xrightarrow{\tau} p' \Rightarrow^{\nu} r'$$

which contradicts the fact that $<\nu, r> \in \alpha(p) = \pi(V(p))$. Thus $\alpha(p) = \pi(V(p)) = \{<\tau, p'>\}$. On the other hand,

$$\alpha(p) = \lambda \circ \pi(V(p)) = \lambda(\{<\tau, p'>\}) = p'$$

which yields a contradiction.

We see that $\alpha(q) = q'$ implies $\alpha(p) =_{wb} p$, which yields a contradiction. Thus we may conclude

$$\alpha(q) \Rightarrow^{\mu}_{+} q'$$

Since $\alpha(p') = p'$ and $\alpha(q') = q'$ by Lemma 5.12, and $\omega(p', q') < \omega(p, q)$, we have by induction $p' = q'$. Thus there exist $q_1, q_2 \in P$ with

$$\alpha(q) \Rightarrow^{\tau} q_1 \xrightarrow{\mu} q_2 \Rightarrow^{\tau} p'$$

We proceed by showing $\alpha(q) = q_1$ and $q_2 = p'$.

First, suppose $\alpha(q) \neq q_1$, i.e., $\alpha(q) \Rightarrow^{\tau}_{+} q_1$. Since $\alpha(p) =_{wb} \alpha(q)$ there exists $p_1 \in P$ with $\alpha(p) \Rightarrow^{\tau} p_1$. Suppose $\alpha(p) = p_1$. Then $\alpha(q) =_{wb} \alpha(p) = p_1 =_{wb} q_1$, thus $\alpha(q) =_{wb} q_1$. This is impossible, as can be shown by exactly the same argument that was used above to prove that $\alpha(p) =_{wb} p'$ leads to a contradiction. Thus $\alpha(p) \Rightarrow^{\tau}_{+} p_1$. Again we can use induction to conclude that $p_1 = q_1$. Since $q_1 \Rightarrow^{\mu}_{+} p'$ we have

$$\alpha(p) \Rightarrow^{\tau}_{+} q_1 \Rightarrow^{\mu}_{+} p'$$

and thus

$$V(p) - \{<\mu, p'>\} \Rightarrow^{\mu}_{+} p'$$

This contradicts the fact that $<\mu, p'> \in \alpha(p) = \pi(V(p))$. We see that $\alpha(q) \neq q_1$ yields a contradiction. Hence, $\alpha(q) = q_1$.

Second, we show that also the assumption that $q_2 \neq p'$ leads to a contradiction. So suppose $q_2 \Rightarrow^{\tau}_{+} p'$. The facts that $\alpha(q) \Rightarrow^{\mu}_{+} q_2$ and $\alpha(p) =_{wb} \alpha(q)$ imply that there exists a process $p_2 =_{wb} q_2$ such that $\alpha(p) \Rightarrow^{\tau} p_2$. As in the case of p_1 above, we have $\alpha(p) \Rightarrow^{\tau}_{+} p_2$. By induction and Lemma 5.12 we have that $p_2 = q_2$. Thus

$$\alpha(p) \Rightarrow^{\mu}_{+} q_2 \Rightarrow^{\tau}_{+} p'$$

whichs contradicts, as above, the fact that $<\mu, p'> \in \alpha(p) = \pi(V(p))$. And so $q_2 = p'$.

The above implies $\alpha(q) \xrightarrow{\mu} p'$. Thus we have shown: $\alpha(p) \subseteq \alpha(q)$. Similarly, one proves $\alpha(q) \subseteq \alpha(p)$ and so we can conclude $\alpha(p) = \alpha(q)$. (End of proof.)

COROLLARY 5.14 (Theorem 5.1): *For all $p, q \in P$,*

$$\alpha(p) = \alpha(q) \Leftrightarrow p =_{wb} q$$

PROOF:
$\Rightarrow$: This is Corollary 5.11.
$\Leftarrow$: Suppose $p =_{wb} q$. By Theorem 5.10 we have $\alpha(p) =_{wb} p$ and $\alpha(q) =_{wb} q$. Hence $\alpha(p) =_{wb} \alpha(q)$. Theorem 5.13 now entails $\alpha(p) = \alpha(q)$.

6. OBSERVATIONAL CONGRUENCE

It is well-known that weak bisimulation equivalence is not a congruence. More specifically, it is not substitutive with respect to the $+$ operator on states. The familiar example is as follows. The states a and $\tau \cdot a$ are weakly bisimilar, but when substituted in the context $\cdots + b$, they are not:

$$\neg(a + b =_{wb} \tau \cdot a + b)$$

Therefore the notion of *observational congruence* ([Mi83]) has been invented. (It is also called *rooted τ-bisimulation* ([BK85]).)

DEFINITION 6.1 (Observational congruence): Let $=_{oc} \subseteq S \times S$ be a symmetric relation defined as follows. For all $s,t \in S$, we put $s =_{oc} t$ if and only if, for all $\mu \in A$,

$$\text{if } s \xrightarrow{\mu} s' \text{ then } \exists t' \in S \; [\, t \Rightarrow^{\mu}_{+} t' \wedge s' =_{wb} t' \,]$$

The states s and t are then called *observationally congruent.*

Again we want to define a model that maps a state to a canonical representative of its equivalence class. As in the case of weak bisimulation, it will be obtained by composing $\mathfrak{M}_{sb}$ with a suitable abstraction operator. This we introduce next.

DEFINITION 6.2 (Abstraction β): Let $\beta: P \rightarrow P$ be defined, for all $p \in P$, by

$$\beta(p) = \pi(\{<\mu, \alpha(p')>: <\mu, p'> \in p\})$$

Note that β is almost the same as α (Definition 4.3), but for the outermost application of λ. In other words: $\alpha = \lambda \circ \beta$.

Now a model for observational congruence is obtained by composing $\mathfrak{M}_{sb}$ with β.

DEFINITION 6.3 ($\mathfrak{M}_{oc}$): The model $\mathfrak{M}_{oc}$: $S \rightarrow P$ is defined, for all $s \in S$, by

$$\mathfrak{M}_{oc}[\![s]\!] = \beta(\mathfrak{M}_{sb}[\![s]\!])$$

The model $\mathfrak{M}_{oc}$ assigns the same value to states if and only if they are observationally congruent. That is the content of the following.

THEOREM 6.4 : *For all s, t∈S,*

$$\mathcal{M}_{oc}[\![s]\!] = \mathcal{M}_{oc}[\![t]\!] \Leftrightarrow s =_{oc} t$$

Again this theorem is a direct consequence of another one, which makes use of the notion of observational congruence for processes. This notion is introduced next.

DEFINITION 6.5 (Observational congruence for processes): Let $=_{oc} \subseteq P \times P$ be a symmetric relation defined as follows. For all $p,q \in P$, we put $p =_{oc} q$ if and only if, for all $\mu \in A$,

$$\text{if } p \xrightarrow{\mu} p' \text{ then } \exists q' \in P\ [q \Rightarrow^{\mu}_{+} q' \wedge p' =_{wb} q']$$

The processes p and q are then called *observationally congruent.*

We have the following fact.

THEOREM 6.6: *For all p, q∈P,*

$$\beta(p) = \beta(q) \Leftrightarrow p =_{oc} q$$

It can be proved along the lines of the previous section. As a matter of fact, it is much easier, since its proof can use many of the results concerning weak bisimulation.

7. DISCUSSION

There are many other weak equivalences. Among the more recently invented ones are *branching-bisimulation equivalence* ([GW89]) and *dynamic-bisimulation equivalence* ([MSa90]). In this paper we have focussed on weak-bisimulation equivalence, because this is found to be the most difficult one to deal with. Without giving any details here we observe that similar results can be obtained for the two above equivalences as minor variations (in fact simplifications) of the definitions and proofs presented here.

The same holds for the notion of *NMS-bisimulation* ([DM87], [DDM87], [MSg89]). In order to describe canonical representatives for that equivalence, it is convenient to use a domain of *node*-labelled, rather than *arc*-labelled trees, which are used here. The operations of pruning and lifting can be straightforwardly adapted for node-labelled trees, and similarly the abstraction operator α (Definition 4.3).

An obvious question is how the results of this paper generalize to infinite behaviour. The domain of finite trees should be extended in such a framework that it contains also infinite elements. Here several ways are open. Apart from the traditional world of complete partial orderings, solutions can be obtained in the form of complete metric spaces ([BZ82], [AR89]). Another framework is offered by the universe of nonwellfounded sets, recently very elegantly presented in [Ac88]. (See also [Ru90b] for some applications of nonwellfounded sets to programming language semantics.) Since at a number of places in our definitions and proofs it is crucial that the structures under consideration are finite, we are not very optimistic regarding the general case of infinite behaviour. For the more restricted case of so-called *regular* processes ([Mi84], [BK88]) we are confident, however, that a similar result can be obtained.

8. REFERENCES

[Ab90] S. ABRAMSKY, *A domain equation for bisimulation,* Imperial College of Science, Dept. of Computing, London, 1990.

[Ac88] P. ACZEL, *Non-well-founded sets,* CSLI Lecture Notes No. 14, 1988.

[AR89] P. AMERICA, J.J.M.M. RUTTEN, *Solving reflexive domain equations in a category of complete metric spaces,* Journal of Computer and System Sciences, Vol. 39, No. 3, 1989, pp. 343-375.

[BK85] J.A. BERGSTRA, J.W. KLOP, *Algebra of communicating processes with abstraction,* Theoretical Computer Science 37, 1985, pp. 77-121.

[BK88] J.A. BERGSTRA, J.W. KLOP, *A complete inference system for regular processes with silent moves,* Logic Colloquium '86 (F.R. Drake, J.K. Truss, eds.), Elsevier North-Holland, 1988.

[BZ82] J.W. DE BAKKER, J.I. ZUCKER, *Processes and the denotational semantics of concurrency,* Information and Control 54, 1982, pp. 70-120.

[Ca87] I. CASTELLANI, *Bisimulations and abstraction homomorphisms,* Journal of Computer and System Sciences 34, 1987, pp. 210-235.

[DDM87] P. DEGANO, R. DE NICOLA, U. MONTANARI, *Observational equivalences for concurrency models,* in Formal Description of Programming Concepts III (M. Wirsing, ed.), North-Holland, 1987, pp. 105-132.

[DM87] P. DEGANO, U. MONTANARI, *Concurrent histories: a basis for observing distributed systems,* Journal of Computer and System Sciences 34, 1987, pp. 442-461.

[GR89] R.J. VAN GLABBEEK, J.J.M.M. RUTTEN, *The processes of De Bakker and Zucker represent bisimulation equivalence classes,* in: J.W. de Bakker, 25 jaar semantiek, Centre for Mathematics and Computer Science, Amsterdam, 1989.

[GW89] R.J. VAN GLABBEEK, W.P. WEIJLAND, *Branching time and abstraction in bisimulation semantics (extended abstract),* Information Processing 89 (G.X. Ritter, ed.), North-Holland, 1989, pp. 613-618.

[Mi80] R. MILNER, *A calculus of communicating systems,* Lecture Notes in Computer Science 92, Springer-Verlag, 1980.

[Mi83] R. MILNER, *Calculi for synchrony and asynchrony,* Theoretical Computer Science Vol. 25 (3), 1983, pp. 267-310.

[Mi84] R. MILNER, *A complete inference system for a class of regular behaviours,* Journal of Computer and System Sciences 28, 1985, pp. 439-466.

[Mi89] R. MILNER, *Communication and concurrency,* Prentice-Hall International, 1989.

[MSa90] U. MONTANARI, V. SASSONE, *Dynamic Bisimulation,* Technical Report TR-13/90, University of Pisa, 1990.

[MSg89] U. MONTANARI, M. SGAMMA, *Canonical representatives for observational equivalence classes,* Proceedings Colloquium on the resolution of equations in algebraic structures (H. Ait-Kaci, M. Nivat, eds.), Academic Press, Inc., 1989, pp. 293-319.

[Pa81] D.M.R. PARK, *Concurrency and automata on infinite sequences,* in: Proceedings 5th GI conference, Lecture Notes in Computer Science 104, Springer-Verlag, 1981, pp. 15-32.

[Pl81] G.D. PLOTKIN, *A structural approach to operational semantics,* Report DAIMI FN-19, Comp. Sci. Dept., Aarhus Univ. 1981.

[Ru90a] J.J.M.M. RUTTEN, *Deriving denotational models for bisimulation from structured operational semantics,* Technical Report CS-R8955, Centre for Mathematics and Computer Science, Amsterdam, 1989. To appear in: Proceedings IFIP TC2 Working Conference on Programming Concepts and Methods, April 2-5, 1990, Israel.

[Ru90b] J.J.M.M. RUTTEN, *Nonwellfounded sets and programming language semantics,* Technical Report CS-R90.., CWI, Amsterdam, 1990 (to appear).