Centrum voor Wiskunde en Informatica

REPORT*RAPPORT*

Time-stamped Actions in pCRL Algebras

M.B. van der Zwaag

Software Engineering (SEN)

SEN-R0002 February 28, 2000

Report SEN-R0002
ISSN 1386-369X

CWI
P.O. Box 94079
1090 GB Amsterdam
The Netherlands

Time-stamped Actions in pCRL Algebras

M.B. van der Zwaag

CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Mark.van.der.Zwaag@cwi.nl

ABSTRACT

We present two extensions of pCRL with time-stamped actions: $p\mathrm{CRL}_\rho$ for absolute time and $p\mathrm{CRL}_{r\rho}$ for relative time. We define timed bisimilarity for both versions and prove that the given axiomatisations are ground complete, provided that the data types have built-in equality and Skolem functions. We base the completeness proofs on the completeness results for untimed pCRL.

1991 Mathematics Subject Classification: 68Q45; 68Q60; 68Q70

1991 ACM Computing Classification System: D.3.3; F.1.1; F.3.1; F.3.2

Keywords & Phrases: timed process algebras, pCRL, completeness, real time

Note: Work carried out under project SEN2.1 "Process Specification and Analysis".

1 Introduction

The specification language μCRL was introduced by Groote and Ponse [5]. It is based on the Algebra of Communicating Processes (Bergstra and Klop [1]), and it combines the specification of data types with the specification of processes. The sublanguage pCRL combines the specification of data types with Basic Process Algebra [1]; it is used for the description of sequential processes.

Groote and Luttik [6] presented an axiomatisation for pCRL that they proved ground complete with respect to strong bisimilarity, under the condition that the data types have *built-in equality* and Skolem functions. Their treatment of the summation over data types, or *alternative quantification* as they call it, differs from [5]. Also, they use so-called *generalised* equational logic as proof theory. We follow them in both respects.

Timed versions of ACP were studied earlier by Baeten and Bergstra [2, 3, 4]. The approach followed here closely resembles the theory of real-time BPA_δ [2]. Actions are parameterized with a time element, that indicates the moment of execution. This parameter, or *time-stamp*, is either an absolute reference to time, in which case the corresponding theory is written $\mathrm{BPA}_{\rho\delta}$, or a relative reference to time, in which case the theory is written $\mathrm{BPA}_{r\rho\delta}$.

Since in pCRL actions can be parameterized with data and since time can be treated as a data type, we can naturally incorporate time-stamps in pCRL. Also, the alternative quantification over data and the conditionals of pCRL provide a powerful means to describe time-dependent processes.

Section 1.1 presents untimed pCRL in the style of [6]. Section 2 introduces processes with time-stamped actions and $p\mathrm{CRL}_\rho$ algebras. We define timed bisimilarity of timed processes, that gives an absolute interpretation of the time-stamps. We show that every timed process is timed bisimilar with a process that can be regarded as an untimed process, i.e. as a process in a set for which timed bisimilarity and (untimed) strong bisimilarity coincide. This result is used in Section 2.3, where we base the completeness proof of the theory $p\mathrm{CRL}_\rho$ on the completeness results of untimed pCRL [6]. We follow a similar strategy in the completeness proof for the relative time variant $p\mathrm{CRL}_{r\rho}$, that is presented in Section 3.

We remark that as time domain for $p\mathrm{CRL}_\rho$ we allow any totally ordered nonempty set, whereas the time domain for $\mathrm{BPA}_{\rho\delta}$ is the set of non-negative real numbers. In particular, we may choose a discrete time domain. Consequently, we do not use the adjective real time for our framework. We write the subscript ρ in the name of the theory, because it clearly stands in the real time tradition.

For relative time, we make a restriction on the possible time domains. If not, then we get counter-intuitive results. E.g. if we take the real numbers as time domain, then it is not clear what it means to execute an action at some negative relative time. Therefore, we take the positive real numbers as time domain. Particularly, we do not have time element zero. In $\mathrm{BPA}_{r\rho\delta}$, zero is a time element as well, but actions cannot be executed at relative time zero. In $p\mathrm{CRL}_{r\rho}$, as in $p\mathrm{CRL}_\rho$, actions can be executed at any of the time elements.

1.1 Untimed pCRL

We present the theory pCRL in the style of [6]. The timed theories in later sections are extensions of pCRL.

The data types A data signature is determined by a set S of sort symbols and a set F of function declarations. We have disjoint infinite sets of variables V_s for the sort symbols s. Let $V = \bigcup_{s\in S} V_s$. We write T_s for the set of terms of sort s. The data signature contains at least the sort symbol $\mathbb{b}$ for the booleans, and the usual function declarations for $\mathtt{t}$, $\mathtt{f}$, $\neg$, $\wedge$ and $\vee$.

For each sort symbol s, we assume a data algebra with universe D_s. The set $\mathrm{D}_{\mathbb{b}}$ has two elements: the interpretation of $\mathtt{t}$ and the interpretation of $\mathtt{f}$.

An assignment f is a function from V to $\bigcup_{s\in S} \mathrm{D}_s$ such that $f(v) \in \mathrm{D}_s$ if and only if $v \in V_s$. We write Ass for the set of assignments. For assignments f and g and variable v, we write $f[v]g$, if for all $u \in V$, $u \neq v$ implies $f(u) = g(u)$. Given an assignment f, we write $[\![t]\!]^f$ for the interpretation of a term t.

For the rest of this paper, we assume that the data types have complete equational axiomatisations.

A data signature, with equational theory E, has *built-in equality* if, for every sort s, it has a function declaration $eq{:}ss \to \mathbb{b}$, such that for all terms t_1, t_2 of sort s, it holds that $E \vdash t_1 = t_2$ if and only if $E \vdash eq(t_1, t_2) = \mathtt{t}$.

Signature A pCRL signature is determined by a data signature and a set of *action declarations*. It has a sort symbol $\mathbb{p}$ that is not a data sort symbol, for *process terms*, and a set $V_{\mathbb{p}}$ of process variables that is disjoint from the set V of data variables. A process term is $\mathbb{p}$-ground, if it has no occurrences of process variables.

The action declarations are of the form $\mathsf{a}{:}s_1 \cdots s_n$, where the s_i are data sort symbols and $n \geq 0$. If $\mathsf{a}{:}s_1 \cdots s_n$ is an action declaration and d_i is a data term of sort s_i for every $i \leq n$, then $\mathsf{a}(d_1, \ldots, d_n)$ is an *action term*. Action terms are of sort $\mathbb{p}$. We write A for the set of action terms.

A pCRL signature has a function declaration $\delta{:}\mathbb{p}$ for the *deadlock* process term, declarations $\cdot, +{:}\mathbb{p}\mathbb{p} \to \mathbb{p}$ for the *sequential* and *alternative* composition respectively, a declaration $\lhd\ \rhd{:}\mathbb{p}\mathbb{b}\mathbb{p} \to \mathbb{p}$ for the *then-if-else* construct, and a binder $\sum{:}\mathbb{p}$. If v is a data variable, i.e. if $v \in V$, and p is a process term, then $\sum_v p$ is a process term. The binder $\sum_v$ binds all free occurrences of v in p. We consider process terms modulo α-conversion. So we may implicitly assume that in an arbitrary process term p, no variables occur both bound and free, and, if $\sum_v p$ is a process term with a subterm $\sum_u q$, then $v \neq u$. A process term $\sum_v p$ represents the *alternative quantification* of p over v, i.e. the choice between the processes p for any value of v. We abbreviate a process term $\sum_{v_1} \cdots \sum_{v_n} p$, with $n \geq 0$, by $\sum_{\vec{v}} p$. We adopt the binding convention that $\cdot$ binds strongest and $\lhd\ \rhd$ binds stronger than $\sum$, which binds stronger than $+$. Process terms $p \cdot q$ are usually written pq.

$\mathsf{a} \xrightarrow{\mathsf{a}} \surd$	$\dfrac{\mathsf{p} \xrightarrow{\mathsf{a}} \surd}{\mathsf{p}\cdot\mathsf{q} \xrightarrow{\mathsf{a}} \mathsf{q}}$	$\dfrac{\mathsf{p} \xrightarrow{\mathsf{a}} \mathsf{p}'}{\mathsf{p}\cdot\mathsf{q} \xrightarrow{\mathsf{a}} \mathsf{p}'\cdot\mathsf{q}}$	$\dfrac{\mathsf{p} \xrightarrow{\mathsf{a}} \surd}{\sum\{\mathsf{p}\}\cup P \xrightarrow{\mathsf{a}} \surd}$	$\dfrac{\mathsf{p} \xrightarrow{\mathsf{a}} \mathsf{p}'}{\sum\{\mathsf{p}\}\cup P \xrightarrow{\mathsf{a}} \mathsf{p}'}$

Table 1: Transition rules for pCRL algebras. $\mathsf{a} \in \mathrm{A}$ and $\mathsf{p},\mathsf{q} \in \mathrm{P}$ and $P \subseteq \mathrm{P}$.

Theory The pCRL axioms are listed in Table 2. As proof theory we use generalised equational logic with a congruence rule for binders. For pCRL, this rule says that we may infer $\sum_v p = \sum_v q$ from $p = q$. Also, substitutions are adjusted to the use of binders. E.g. one may, in axiom SUM1, substitute for the process variable x any process term without free occurrences of the data variable v. We refer to [6] for a precise exposition of the proof theory.

Semantics We interpret $\mathbb{p}$-ground pCRL process terms as *processes* in a pCRL algebra. A pCRL algebra $\mathfrak{P}$ has a universe P of processes. The algebra has a set A of constants that are called *actions*, a constant $\delta^{\mathfrak{p}} \notin \mathrm{A}$, a binary operator $\cdot^{\mathfrak{p}} : \mathrm{P}^2 \to \mathrm{P}$ and a unary operator $\sum^{\mathfrak{p}} : 2^{\mathrm{P}} \setminus \emptyset \to \mathrm{P}$. The set P is the smallest set that can be generated from these operators.

We write $\mathsf{a},\mathsf{b},\ldots$ for actions and $\mathsf{p},\mathsf{q},\ldots$ for processes. We may write $\mathsf{p} +^{\mathfrak{p}} \mathsf{q}$ for the process $\sum^{\mathfrak{p}}\{\mathsf{p},\mathsf{q}\}$, and pq for the process $\mathsf{p} \cdot^{\mathfrak{p}} \mathsf{q}$. We write $\mathfrak{P}(\mathrm{A})$ for a pCRL algebra $\mathfrak{P}$ with set of actions A. We shall often ommit the superscript on operations, thus using the same notation for operation symbols and their interpretation.

Given a pCRL signature with set A of action terms, $\mathbb{p}$-ground process terms are interpreted in the pCRL algebra with set of actions

$$\mathrm{A} = \{[\![a]\!]^f \mid f \in Ass,\ a \in A\}.$$

This interpretation is defined as follows (given an assignment f).

$$\begin{aligned}
[\![\mathsf{a}(d_1,\ldots,d_n)]\!]^f &= \mathsf{a}([\![d_1]\!]^f,\ldots,[\![d_n]\!]^f)\\
[\![\delta]\!]^f &= \delta^{\mathfrak{p}}\\
[\![p+q]\!]^f &= [\![p]\!]^f +^{\mathfrak{p}} [\![q]\!]^f\\
[\![p\cdot q]\!]^f &= [\![p]\!]^f \cdot^{\mathfrak{p}} [\![q]\!]^f\\
[\![\textstyle\sum_v p]\!]^f &= \textstyle\sum^{\mathfrak{p}}\{[\![p]\!]^g \mid f[v]g\}\\
[\![p \lhd b \rhd q]\!]^f &= \begin{cases} [\![p]\!]^f & \text{if } [\![b]\!]^f = [\![\mathtt{t}]\!]\\ [\![q]\!]^f & \text{if } [\![b]\!]^f = [\![\mathtt{f}]\!]\end{cases}
\end{aligned}$$

The transition relations $\_ \xrightarrow{\_} \_ \subseteq (\mathrm{P}\times\mathrm{A}\times\mathrm{P})$ and $\_ \xrightarrow{\_} \surd \subseteq (\mathrm{P}\times\mathrm{A})$ are defined by the transition rules in Table 1.

Definition 1.1 A binary relation $\mathcal{R}$ on P is a *(strong) bisimulation* if it is symmetric and whenever $\mathsf{p}\mathcal{R}\mathsf{q}$, then $\mathsf{p} \xrightarrow{\mathsf{a}} \surd$ implies $\mathsf{q} \xrightarrow{\mathsf{a}} \surd$, and $\mathsf{p} \xrightarrow{\mathsf{a}} \mathsf{p}'$ implies $\mathsf{q} \xrightarrow{\mathsf{a}} \mathsf{q}'$ for some q' with $\mathsf{p}'\mathcal{R}\mathsf{q}'$.

Processes p and q are *(strongly) bisimilar*, notation $\mathsf{p} \leftrightarroweq \mathsf{q}$, if there is a bisimulation that relates p and q.

It is proved in [6] that bisimilarity is a congruence. We write $\mathfrak{P}/\leftrightarroweq$ for the quotient algebra $\mathfrak{P}$ modulo bisimilarity. If $\mathbb{p}$-ground process terms p and q are interpreted in $\mathfrak{P}$, then we write $\mathfrak{P}/\leftrightarroweq \models p = q$, if $[\![p]\!]^f \leftrightarroweq [\![q]\!]^f$ for all assignments f.

Completeness We assume that the data types have *built-in equality* and Skolem functions (cf. [6]). This means that the first-order theory of the data is decidable. For completeness we need the following extra axioms. First, for every action declaration $\mathsf{a}{:}s_1 \cdots s_n$ with $n > 0$ an axiom

$$(\text{AE}_{\mathsf{a}}) \qquad \mathsf{a}(\bar{x}) \lhd eq(\bar{x}, \bar{y}) \rhd \delta \;=\; \mathsf{a}(\bar{y}) \lhd eq(\bar{x}, \bar{y}) \rhd \delta,$$

where by $\bar{x}$ and $\bar{y}$ we abbreviate parameter lists $x_1, \ldots, x_n$ and $y_1, \ldots, y_n$, respectively. By $eq(\bar{x}, \bar{y})$ we abbreviate the boolean term $eq(x_1, y_1) \land \cdots \land eq(x_n, y_n)$.

Finally, we need the following axiom that is called the *static condition axiom*.

$$(\text{SCA}) \qquad (x \lhd b \rhd \delta)(y \lhd b \rhd \delta) \;=\; xy \lhd b \rhd \delta.$$

The proof of the completeness theorem below may be found in [6].

Theorem 1.1 (Completeness) If the data types have built-in equality and Skolem functions, and E is the equational theory of the data types, then $\mathfrak{P}/\leftrightarrow \models p = q$, if and only if $p\text{CRL} + E + \text{AE} + \text{SCA} \vdash p = q$, for all $\mathbb{p}$-ground process terms p and q.

(A1)	$x + y$	$=$	$y + x$
(A2)	$x + (y + z)$	$=$	$(x + y) + z$
(A3)	$x + x$	$=$	x
(A4)	$(x + y)z$	$=$	$xz + yz$
(A5)	$(xy)z$	$=$	$x(yz)$
(A6)	$x + \delta$	$=$	x
(A7)	δx	$=$	δ
(SUM1)	$\sum_v x$	$=$	x
(SUM3)	$\sum_v p$	$=$	$\sum_v p + p$
(SUM4)	$\sum_v (p + q)$	$=$	$\sum_v p + \sum_v q$
(SUM5)	$(\sum_v p)x$	$=$	$\sum_v px$
(SUM12)	$(\sum_v p) \lhd b \rhd \delta$	$=$	$\sum_v p \lhd b \rhd \delta$
(CND1)	$x \lhd \mathtt{t} \rhd y$	$=$	x
(CND2)	$x \lhd \mathtt{f} \rhd y$	$=$	y
(CND3)	$x \lhd b \rhd y$	$=$	$x \lhd b \rhd \delta + y \lhd \neg b \rhd \delta$
(CND4)	$(x \lhd b_1 \rhd \delta) \lhd b_2 \rhd \delta$	$=$	$x \lhd b_1 \land b_2 \rhd \delta$
(CND5)	$(x \lhd b_1 \rhd \delta) + (x \lhd b_2 \rhd \delta)$	$=$	$x \lhd b_1 \lor b_2 \rhd \delta$
(CND6)	$(x \lhd b \rhd \delta)y$	$=$	$xy \lhd b \rhd \delta$
(CND7)	$(x + y) \lhd b \rhd \delta$	$=$	$x \lhd b \rhd \delta + y \lhd b \rhd \delta$

Table 2: pCRL axioms. $x, y, z \in V_{\mathbb{p}}$. $b, b_1, b_2 \in V_{\mathbb{b}}$. p, q range over $\mathbb{p}$-ground process terms.

2 Absolute time

We first introduce $p\text{CRL}_\rho$ algebras and timed bisimilarity. The theory $p\text{CRL}_\rho$ follows in Section 2.2.

Let a totally ordered set T of time elements be given. In this section we consider algebras that model the behaviour of processes *in time*. The basic ingredient is that actions are executed at some time. We let actions be *time-stamped*; they are parameterised with a time element. An action $\mathsf{a}(\mathsf{t})$ is executed at (absolute) time t. The execution of $\mathsf{a}(\mathsf{t})$ terminates at time t and no other actions may be executed at t subsequently. We shall see that the process $\mathsf{a}(\mathsf{t})\mathsf{b}(\mathsf{t})$ is equivalent with the process $\mathsf{a}(\mathsf{t})\delta$.

$$\mathsf{a} \xrightarrow{\mathsf{a}}_\rho \surd \qquad \frac{\mathsf{p} \xrightarrow{\mathsf{a(t)}}_\rho \surd}{\mathsf{p}\cdot\mathsf{q} \xrightarrow{\mathsf{a(t)}}_\rho \mathsf{t}\gg\mathsf{q}} \qquad \frac{\mathsf{p} \xrightarrow{\mathsf{a}}_\rho \mathsf{p}'}{\mathsf{p}\cdot\mathsf{q} \xrightarrow{\mathsf{a}}_\rho \mathsf{p}'\cdot\mathsf{q}} \qquad \frac{\mathsf{p} \xrightarrow{\mathsf{a}}_\rho \surd}{\sum\{\mathsf{p}\}\cup P \xrightarrow{\mathsf{a}}_\rho \surd}$$

$$\frac{\mathsf{p} \xrightarrow{\mathsf{a}}_\rho \mathsf{p}'}{\sum\{\mathsf{p}\}\cup P \xrightarrow{\mathsf{a}}_\rho \mathsf{p}'} \qquad \frac{\mathsf{p} \xrightarrow{\mathsf{a(u)}}_\rho \surd \quad \mathsf{t}<\mathsf{u}}{\mathsf{t}\gg\mathsf{p} \xrightarrow{\mathsf{a(u)}}_\rho \surd} \qquad \frac{\mathsf{p} \xrightarrow{\mathsf{a(u)}}_\rho \mathsf{p}' \quad \mathsf{t}<\mathsf{u}}{\mathsf{t}\gg\mathsf{p} \xrightarrow{\mathsf{a(u)}}_\rho \mathsf{p}'}$$

$$\frac{U_\mathsf{u}(\mathsf{p}) \quad \mathsf{t}<\mathsf{u}}{U_\mathsf{t}(\mathsf{p})} \qquad \frac{U_\mathsf{t}(\mathsf{p})}{U_\mathsf{t}(\mathsf{p}\cdot\mathsf{q})} \qquad \frac{U_\mathsf{t}(\mathsf{p})}{U_\mathsf{t}(\sum\{\mathsf{p}\}\cup P)} \qquad \frac{U_\mathsf{u}(\mathsf{p})}{U_\mathsf{u}(\mathsf{t}\gg\mathsf{p})}$$

$$U_\mathsf{t}(\mathsf{a(t)}) \qquad U_\mathsf{t}(\delta(\mathsf{t})) \qquad U_\mathsf{t}(\mathsf{t}\gg\mathsf{p})$$

Table 3: Transition rules for $\_ \xrightarrow{-}_\rho \_ \subseteq (\mathrm{P}\times\mathrm{A}\times\mathrm{P})$ and $\_ \xrightarrow{-}_\rho \surd \subseteq (\mathrm{P}\times\mathrm{A})$, and delay predicates $U_\mathsf{t} \subseteq \mathrm{P}$ for the elements t of T. ($\mathsf{a}\in\mathrm{A}$ and $\mathsf{p},\mathsf{q}\in\mathrm{P}$ and $P\subseteq\mathrm{P}$ and $\mathsf{t},\mathsf{u}\in\mathrm{T}$)

The algebras we consider here have, for every time element t, a so-called *time-deadlock* process $\delta(\mathsf{t})$ and a unary *initialisation* operator $\mathsf{t}\gg$. A process $\mathsf{t}\gg\mathsf{p}$ behaves like p, except that initial actions before or at t are blocked.

An important notion is the *existence* of a process in time. The time-deadlock $\delta(\mathsf{t})$ does not have behaviour in the sense of the performance of actions, but it does have behaviour with respect to its existence in time: it *exists* at any moment before t and at t, or, equivalently, it can let time pass until t. Thus, our notion of timed bisimilarity will distinguish between processes $\delta(\mathsf{t})$ and $\delta(\mathsf{u})$ if $\mathsf{t}\neq\mathsf{u}$. The process δ does not exist at any time. A process $\mathsf{t}\gg\mathsf{p}$ can let time pass at least until t.

Definition 2.1 (pCRL$_\rho$ algebras) Let a totally ordered time domain T and a set A of time-stamped actions be given. A pCRL$_\rho$ algebra $\mathfrak{P}$ with universe P has set of constants A, a constant $\delta^\mathfrak{p}\notin\mathrm{A}$ and, for every $\mathsf{t}\in\mathrm{T}$, a constant $\delta(\mathsf{t})^\mathfrak{p}\notin\mathrm{A}$. It has a binary operator $\cdot^\mathfrak{p} : \mathrm{P}^2\to\mathrm{P}$, a unary operator $\sum^\mathfrak{p} : 2^\mathrm{P}\setminus\emptyset\to\mathrm{P}$, and, for every $\mathsf{t}\in\mathrm{T}$, a unary operator $\mathsf{t}\gg^\mathfrak{p} : \mathrm{P}\to\mathrm{P}$. The set of processes P is defined as the smallest set that can be generated from these operators.

If A is a set of time-stamped actions, then we write A_{td} for the set $\mathrm{A}\cup\{\delta(\mathsf{t})\mid\mathsf{t}\in\mathrm{T}\}$. We use $\mathsf{a},\mathsf{b},\ldots$ as names for arbitrary elements of A_{td}. We may write $\mathsf{a(t)}$ to refer to an action a with time-stamp t.

The transition relations and the *delay predicates* U_t are defined in Table 3. A process p exists at time t if and only if $U_\mathsf{t}(\mathsf{p})$. Let for instance p be the process $\mathsf{a(t)b(u)}$ with $\mathsf{t}<\mathsf{u}$. It holds that

$$\mathsf{p} \xrightarrow{\mathsf{a(t)}}_\rho \mathsf{t}\gg\mathsf{b(u)} \xrightarrow{\mathsf{b(u)}}_\rho \surd$$

and $U_\mathsf{t}(\mathsf{p})$, but not $U_\mathsf{u}(\mathsf{p})$.

Definition 2.2 A binary relation $\mathcal{R}$ on P is a ρ*-bisimulation*, if it is symmetric, and whenever $\mathsf{p}\mathcal{R}\mathsf{q}$ and $\mathsf{a}\in\mathrm{A}$, then

1. if $\mathsf{p} \xrightarrow{\mathsf{a}}_\rho \surd$, then $\mathsf{q} \xrightarrow{\mathsf{a}}_\rho \surd$,
2. if $\mathsf{p} \xrightarrow{\mathsf{a}}_\rho \mathsf{p}'$, then $\mathsf{q} \xrightarrow{\mathsf{a}}_\rho \mathsf{q}'$, for some q' with $\mathsf{p}'\mathcal{R}\mathsf{q}'$,
3. if $U_\mathsf{t}(\mathsf{p})$ for some t, then $U_\mathsf{t}(\mathsf{q})$.

Processes p and q are *ρ-bisimilar*, notation $\mathsf{p} \leftrightarrows_\rho \mathsf{q}$, if there is a ρ-bisimulation that relates p and q.

Theorem 2.1 The relation $\leftrightarrows_\rho$ is a congruence on $\mathfrak{P}$.

Proof. It is straightforward to prove that $\leftrightarrows_\rho$ is an equivalence. We show that the substitution property holds for $\cdot$, $\sum$ and $\gg$. We use implicitly that the union of ρ-bisimulations is itself a ρ-bisimulation.

Suppose that $\mathcal{R}$ is a ρ-bisimulation with $\mathsf{p}\mathcal{R}\mathsf{q}$. It is straightforward to prove that the relation

$$\{\langle \mathsf{t}\gg\mathsf{p}, \mathsf{t}\gg\mathsf{q}\rangle, \langle \mathsf{t}\gg\mathsf{q}, \mathsf{t}\gg\mathsf{p}\rangle\} \cup \mathcal{R}$$

is a ρ-bisimulation that relates $\mathsf{t}\gg\mathsf{p}$ and $\mathsf{t}\gg\mathsf{q}$ for any $\mathsf{t} \in \mathrm{T}$.

Suppose that $\mathcal{R}$ is a ρ-bisimulation with $\mathsf{p}_1\mathcal{R}\mathsf{q}_1$ and $\mathsf{p}_2\mathcal{R}\mathsf{q}_2$. It is straightforward to prove that the relation

$$\{\langle \mathsf{p}, \mathsf{q}\rangle,\ \langle \mathsf{t}\gg\mathsf{p}, \mathsf{t}\gg\mathsf{q}\rangle,\ \langle \mathsf{p}\cdot\mathsf{p}', \mathsf{q}\cdot\mathsf{q}'\rangle \mid \mathsf{p}\mathcal{R}\mathsf{q},\ \mathsf{p}'\mathcal{R}\mathsf{q}',\ \mathsf{t}\in\mathrm{T}\}$$

is a ρ-bisimulation that relates $\mathsf{p}_1 \cdot \mathsf{p}_2$ and $\mathsf{q}_1 \cdot \mathsf{q}_2$.

Let P and Q be nonempty sets of processes and let $\mathcal{R}$ be a ρ-bisimulation such that for all p in P there exists a q in Q with $\mathsf{p}\mathcal{R}\mathsf{q}$, and for all q in Q there exists a p in P with $\mathsf{q}\mathcal{R}\mathsf{p}$. It is straightforward to prove that the relation

$$\{\langle \textstyle\sum P, \sum Q\rangle, \langle \sum Q, \sum P\rangle\} \cup \mathcal{R}$$

is a ρ-bisimulation that relates $\sum P$ and $\sum Q$. □

2.1 Strongly bisimilar timed processes

The subset P_{if} of P is the set of *initialisation-free* processes, i.e. the processes that do not have occurrences of initialisation operators. An important observation that we will exploit below, is that the initialisation-free processes are exactly the processes of the untimed pCRL algebra based on the set of actions A_{td}. Note that in that algebra the time-deadlocks are transition labels. So, if a process is initialisation-free, then we can compare its behaviour *as an untimed* process, with its behaviour as a timed process. For instance, a process $\mathsf{a}(\mathsf{t})\mathsf{p} \in \mathrm{P}_{if}$ with $\mathsf{a}(\mathsf{t}) \in \mathrm{A}$ has transitions

$$\mathsf{a}(\mathsf{t})\mathsf{p} \xrightarrow{\mathsf{a}(\mathsf{t})} \mathsf{p} \quad\text{and}\quad \mathsf{a}(\mathsf{t})\mathsf{p} \xrightarrow{\mathsf{a}(\mathsf{t})}_\rho \mathsf{t}\gg\mathsf{p}.$$

For another example, let $\mathsf{p} \in \mathrm{P}_{if}$ and observe that processes $\delta(\mathsf{t})\mathsf{p}$ and $\delta(\mathsf{t})$ are ρ-bisimilar. However, these processes are not strongly bisimilar, because

$$\delta(\mathsf{t})\mathsf{p} \xrightarrow{\delta(\mathsf{t})} \mathsf{p}, \quad\text{whereas}\quad \delta(\mathsf{t}) \xrightarrow{\delta(\mathsf{t})} \surd.$$

From now on it is left implicit that the labels of the untimed transitions of an initialisation-free process may be time-deadlocks.

In Definition 2.3 we define three subsets of P_{if}. We shall see that for processes that are in the intersection of these three sets, ρ-bisimilarity coincides with strong bisimilarity.

Definition 2.3 A process $\mathsf{p} \in \mathrm{P}_{if}$ is *well-timed*, if whenever $\mathsf{p} \xrightarrow{\mathsf{a}(\mathsf{t})} \mathsf{p}'$ with $\mathsf{a} \in \mathrm{A}_{td}$, then $\mathsf{t}\gg\mathsf{p}' \leftrightarrows_\rho \mathsf{p}'$ and p' is well-timed.

A process $\mathsf{p} \in \mathrm{P}_{if}$ is *deadlock-saturated with respect to time* t, if

1. whenever $U_\mathsf{u}(\mathsf{p})$ with $\mathsf{t} \leq \mathsf{u}$, then $\mathsf{p} \xrightarrow{\delta(\mathsf{u})} \surd$ or $\mathsf{p} \xrightarrow{\delta(\mathsf{u})} \mathsf{p}'$ for some p', and

2. whenever $\mathsf{p} \xrightarrow{\mathsf{a(u)}} \mathsf{p}'$ with $\mathsf{a} \in \mathrm{A}$, then p' is deadlock-saturated with respect to u.

A process is *deadlock-saturated*, if it is deadlock-saturated with respect to every $\mathsf{t} \in \mathrm{T}$.

A process $\mathsf{p} \in \mathrm{P}_{\mathit{if}}$ is *deadlock-terminating*, if whenever $\mathsf{p} \xrightarrow{\mathsf{a}} \mathsf{p}'$, then $\mathsf{a} \in \mathrm{A}$ and p' is deadlock-terminating.

We state without proof that for every process p there is a process $\mathsf{q} \in \mathrm{P}_{\mathit{if}}$ that is well-timed, deadlock-saturated and deadlock-terminating, such that $\mathsf{p} \leftrightarroweq_\rho \mathsf{q}$, and that for all $\mathsf{p}, \mathsf{q} \in \mathrm{P}_{\mathit{if}}$ it holds that $\mathsf{p} \leftrightarroweq \mathsf{q}$ implies $\mathsf{p} \leftrightarroweq_\rho \mathsf{q}$.

Lemma 2.1 For all $\mathsf{p}, \mathsf{q} \in \mathrm{P}$ and $\mathsf{t} \in \mathrm{T}$, it holds that $(\mathsf{t} \gg \mathsf{p})\mathsf{q} \leftrightarroweq_\rho \mathsf{t} \gg (\mathsf{pq})$.

Lemma 2.2 For all $\mathsf{p}, \mathsf{q} \in \mathrm{P}_{\mathit{if}}$ that are well-timed, deadlock-saturated and deadlock-terminating, it holds that $\mathsf{p} \leftrightarroweq_\rho \mathsf{q}$ implies $\mathsf{p} \leftrightarroweq \mathsf{q}$.

Proof. Suppose that $\mathcal{R}$ witnesses $\mathsf{p} \leftrightarroweq_\rho \mathsf{q}$. We show that $\mathsf{p} \leftrightarroweq \mathsf{q}$ by induction on the complexity of p.

1. if $\mathsf{p} \xrightarrow{\mathsf{a(t)}} \surd$, then, if $\mathsf{a(t)} \neq \delta(\mathsf{t})$, it follows from $\mathsf{p} \leftrightarroweq_\rho \mathsf{q}$ that $\mathsf{q} \xrightarrow{\mathsf{a(t)}} \surd$.

   Now suppose that $\mathsf{a(t)} = \delta(\mathsf{t})$. It follows from $U_\mathsf{t}(\mathsf{p})$ and $\mathsf{p} \leftrightarroweq_\rho \mathsf{q}$, that $U_\mathsf{t}(\mathsf{q})$. From $U_\mathsf{t}(\mathsf{q})$ and the deadlock-saturatedness of q it follows that $\mathsf{q} \xrightarrow{\mathsf{a(t)}} \surd$.

2. if $\mathsf{p} \xrightarrow{\mathsf{a(t)}} \mathsf{p}'$, then, since p is deadlock-terminating, we have that $\mathsf{a(t)} \neq \delta(\mathsf{t})$. We see that $\mathsf{p} \xrightarrow{\mathsf{a(t)}}_\rho \mathsf{p}^*$, where either $\mathsf{p}^* = \mathsf{t} \gg \mathsf{p}'$, or $\mathsf{p}^* = (\mathsf{t} \gg \mathsf{p}_1')\mathsf{p}_2'$ and $\mathsf{p}' = \mathsf{p}_1'\mathsf{p}_2'$. By the well-timedness of p and Lemma 2.1, we find $\mathsf{p}' \leftrightarroweq_\rho \mathsf{p}^*$.

   By $\mathsf{p} \leftrightarroweq_\rho \mathsf{q}$ we find that $\mathsf{q} \xrightarrow{\mathsf{a(t)}}_\rho \mathsf{q}^*$ for some q^* with $\mathsf{p}^* \mathcal{R} \mathsf{q}^*$. We see that either $\mathsf{q}^* = \mathsf{t} \gg \mathsf{q}'$ or $\mathsf{q}^* = (\mathsf{t} \gg \mathsf{q}_1')\mathsf{q}_2'$, for some q' with in the second case $\mathsf{q}' = \mathsf{q}_1'\mathsf{q}_2'$. We also see that $\mathsf{q} \xrightarrow{\mathsf{a(t)}} \mathsf{q}'$. By the well-timedness of q and Lemma 2.1, we find $\mathsf{q}' \leftrightarroweq_\rho \mathsf{q}^*$.

   Since $\leftrightarroweq_\rho$ is an equivalence we have $\mathsf{p}' \leftrightarroweq_\rho \mathsf{q}'$. By induction it follows that $\mathsf{p}' \leftrightarroweq \mathsf{q}'$. □

2.2 The theory pCRL$_\rho$

We present the signature and theory of pCRL$_\rho$. We define *basic terms* and show that every $\mathbb{p}$-ground process term is derivably equal to a basic term. In Section 2.3 we prove that pCRL$_\rho$ is complete, provided that the data types have built-in equality and Skolem functions.

Assume that the data signature has a sort symbol $\mathbb{t}$ and a binary function symbol $\leq : \mathbb{t}\mathbb{t} \to \mathbb{b}$. The only requirement on the set $\mathrm{D}_\mathbb{t}$ is that it be totally ordered. We write $u < t$ for the boolean term $u \leq t \wedge \neg(t \leq u)$.

A pCRL$_\rho$ signature is a pCRL signature extended with declarations $\delta{:}\mathbb{t} \to \mathbb{p}$ and $\gg{:}\mathbb{t}\mathbb{p} \to \mathbb{p}$. Furthermore, for every action declaration $\mathsf{a}{:}s_1 \cdots s_n$ it holds that $n > 0$ and $s_n = \mathbb{t}$. We write A for the set of action terms and write A_{td} for the set $A \cup \{\delta(t) \mid t \in T_\mathbb{t}\}$. We use $a, b, \ldots$ as names for arbitrary elements of A_{td}. The last parameter of an action term a is refered to as its time-stamp. The term t is called the time-stamp of the time-deadlock term $\delta(t)$. If t is the time-stamp of a, then we also write $a(t)$ to refer to a.

We extend the interpretation function given in Section 1.1 with

$$[\![\delta(t)]\!]^f = \delta([\![t]\!]^f) \text{ and } [\![t \gg p]\!]^f = [\![t]\!]^f \gg [\![p]\!]^f.$$

Thus, we interpret $\mathbb{p}$-ground pCRL$_\rho$ process terms as processes in a pCRL$_\rho$ algebra with time domain $\mathrm{D}_\mathbb{t}$, where every action is of the form $\mathsf{a}(\mathsf{d}_1, \ldots, \mathsf{d}_n)$ with d_n its time-stamp.

The theory pCRL$_\rho$ consists of the axioms of pCRL plus the axioms in Table 4. The soundness proof is a tedious, but straightforward exercise.

(AT1)	$a(t)$	$=$	$a(t) + \sum_u \delta(u) \lhd u \leq t \rhd \delta$
(AT2)	$a(t)x$	$=$	$a(t)(t \gg x)$
(AT3)	$\delta(t)x$	$=$	$\delta(t)$
(AT4)	$t \gg a(u)$	$=$	$a(u) \lhd t < u \rhd \delta(t)$
(AT5)	$t \gg (x+y)$	$=$	$t \gg x + t \gg y$
(AT6)	$t \gg xy$	$=$	$(t \gg x)y$
(AT7)	$t \gg \sum_v p$	$=$	$\sum_v t \gg p$
(AT8)	$t \gg (x \lhd b \rhd \delta)$	$=$	$t \gg x \lhd b \rhd \delta(t)$

Table 4: $p\mathrm{CRL}_\rho$ axioms. $a \in A_{td}$.

Definition 2.4 (Basic terms) The set of $p\mathrm{CRL}_\rho$ basic terms is defined inductively as follows.

1. Every term $\sum_{\vec{v}} a \lhd b \rhd \delta$, with $a \in A_{td}$ and $b \in T_{\mathbb{b}}$, is a basic term.
2. If p is a basic term, then $\sum_{\vec{v}} ap \lhd b \rhd \delta$ with $a \in A$ and $b \in T_{\mathbb{b}}$ is a basic term.
3. If p and q are basic terms, then $p + q$ is a basic term.

If a basic term is of the first form, then we say it is *of type 1*. Similarly for forms 2 and 3.

Lemma 2.3 For every $\mathbb{p}$-ground process term p there is a basic term q such that $p\mathrm{CRL} + \mathrm{AT}3\text{-}8 \vdash p = q$.

Proof. We apply induction on the number of symbols in p. If $p \equiv \delta$, then p equals the basic term $a \lhd \mathtt{f} \rhd \delta$ by CND2, for any $a \in A_{td}$. If $p \in A_{td}$, then p equals the basic term $p \lhd \mathtt{t} \rhd \delta$ by axiom CND1. If $p \equiv p_1 + p_2$, then p is derivably equal to a basic term by induction hypothesis.

If $p \equiv \sum_v p'$, then $p = \sum_v q$, for some basic p'' with $p' = p''$ by induction. We apply induction on p''. If p'' is of type 1 or 2, then $\sum_v p''$ is basic. If p'' is of type 3, then we use axiom SUM4 and induction.

If $p \equiv p_1 \lhd b \rhd p_2$, then $p = p'_1 \lhd b \rhd p'_2$, for some basic p'_1, p'_2 with $p_1 = p'_1$ and $p_2 = p'_2$ by induction. By CND3, we find that p equals $p'_1 \lhd b \rhd \delta + p'_2 \lhd \neg b \rhd \delta$. We show that the first summand is derivably equal to a basic term by induction on p'_1. The case of the second summand is similar. If p'_1 is of type 1 or 2, then we use axioms CND4 and SUM12. If p'_1 is of type 3, then we use CND7 and induction.

If $p \equiv p_1 p_2$, then $p = p'_1 p'_2$, for some basic p'_1, p'_2 with $p_1 = p'_1$ and $p_2 = p'_2$ by induction. We apply induction on p'_1. If p'_1 is of type 1, then we use axioms SUM5, CND6 and, occasionally, AT3. If p'_1 is of type 2, then we use SUM5, CND6, A5, AT3 and induction. If p'_1 is of type 3, then we use A4 and induction.

If $p \equiv t \gg p'$, then $p = t \gg p''$, for some basic p'' with $p' = p''$ by induction. We apply induction on p''. If p'' is of type 1, then let $p'' \equiv \sum_{\vec{v}} a(u) \lhd b \rhd \delta$ and derive by axioms AT4,7,8 that p equals

$$\textstyle\sum_{\vec{v}} (a(u) \lhd t < u \rhd \delta(t)) \lhd b \rhd \delta(t),$$

and by axioms CND3,4,5,7 and SUM4, that this term equals

$$\textstyle\sum_{\vec{v}} a(u) \lhd t < u \wedge b \rhd \delta + \sum_{\vec{v}} \delta(t) \lhd (\neg(t < u) \wedge b) \vee \neg b \rhd \delta,$$

which is basic. If p'' is of type 2, then we use a similar argument. If p'' is of type 3, then we use AT5 and induction. □

2.3 Completeness

We show that every $\mathbb{p}$-ground process term is derivably equal to a term that is interpreted as a well-timed, deadlock-saturated, deadlock-terminating initialisation-free process. Then completeness follows from Lemma 2.2 and Theorem 1.1.

We call a $\mathbb{p}$-ground process term p well-timed, if $[\![p]\!]^f$ is well-timed for every f. Deadlock-saturated process terms and deadlock-terminating process terms are defined similarly. Observe that every basic term is deadlock-terminating. We first show that every basic term is derivably equal to a deadlock-saturated basic term (Lemma 2.6). This result is based on axiom AT1, which says that for every $a(t) \in A_{td}$ it holds that

$$a(t) = a(t) + \sum_u \delta(u) \lhd u \leq t \rhd \delta.$$

Observe that the right-hand side of this identity is deadlock-saturated.

Lemma 2.4 If $\mathbb{p}$-ground process term p is deadlock-saturated and $p\mathrm{CRL} + \mathrm{AT2\text{-}8} \vdash p = q$, for some $\mathbb{p}$-ground q, then q is deadlock-saturated.

Proof. Straightforward.

Lemma 2.5 A $\mathbb{p}$-ground process term $p + q$ is deadlock-saturated, if both p and q are deadlock-saturated.

Proof. Straightforward.

Lemma 2.6 Every $\mathbb{p}$-ground process term p is derivably equal to a deadlock-saturated basic term.

Proof. By Lemma 2.3, we may assume that p is basic. We apply induction on the structure of p. If p is of type 3, then we use induction and Lemma 2.5. If p is of type 1, then let $p \equiv \sum_{\vec{v}} a(t) \lhd b \rhd \delta$ and derive using axiom AT1, that p equals

$$\textstyle\sum_{\vec{v}}(a(t) + \sum_u \delta(u) \lhd u \leq t \rhd \delta) \lhd b \rhd \delta,$$

which is deadlock-saturated. By Lemmas 2.3 and 2.4 we see that it is derivably equal to a deadlock-saturated basic term.

If p is of type 2, then let $p \equiv \sum_{\vec{v}} a(t)p' \lhd b \rhd \delta$ with p' deadlock-saturated, and derive using axiom AT1, that p equals

$$\textstyle\sum_{\vec{v}}(a(t) + \sum_u \delta(u) \lhd u \leq t \rhd \delta)p' \lhd b \rhd \delta,$$

which is deadlock-saturated. By Lemmas 2.3 and 2.4 we see that it is derivably equal to a deadlock-saturated basic term. □

Lemma 2.7 A basic term $p + q$ is well-timed, if both p and q are well-timed.

Proof. Straightforward.

Lemma 2.8 If basic term p is well-timed, then $a(t)(t \gg p)$ is well-timed.

Proof. Straightforward.

Lemma 2.9 If $\mathbb{p}$-ground process term p is well-timed and $p\mathrm{CRL} + \mathrm{AT3\text{-}8} \vdash p = q$, for some $\mathbb{p}$-ground q, then q is well-timed.

$\mathsf{a} \xrightarrow{\mathsf{a}}_{r\rho} \surd$	$\dfrac{\mathsf{p} \xrightarrow{\mathsf{a}}_{r\rho} \surd}{\mathsf{p}\cdot\mathsf{q} \xrightarrow{\mathsf{a}}_{r\rho} \mathsf{q}}$	$\dfrac{\mathsf{p} \xrightarrow{\mathsf{a}}_{r\rho} \mathsf{p}'}{\mathsf{p}\cdot\mathsf{q} \xrightarrow{\mathsf{a}}_{r\rho} \mathsf{p}'\cdot\mathsf{q}}$	$\dfrac{\mathsf{p} \xrightarrow{\mathsf{a}}_{r\rho} \surd}{\sum\{\mathsf{p}\}\cup P \xrightarrow{\mathsf{a}}_{r\rho} \surd}$	$\dfrac{\mathsf{p} \xrightarrow{\mathsf{a}}_{r\rho} \mathsf{p}'}{\sum\{\mathsf{p}\}\cup P \xrightarrow{\mathsf{a}}_{r\rho} \mathsf{p}'}$
$U_{\mathsf{t}}(\mathsf{a}[\mathsf{t}])$	$U_{\mathsf{t}}(\delta[\mathsf{t}])$	$\dfrac{U_{\mathsf{u}}(\mathsf{p}) \quad \mathsf{t}<\mathsf{u}}{U_{\mathsf{t}}(\mathsf{p})}$	$\dfrac{U_{\mathsf{t}}(\mathsf{p})}{U_{\mathsf{t}}(\mathsf{p}\cdot\mathsf{q})}$	$\dfrac{U_{\mathsf{t}}(\mathsf{p})}{U_{\mathsf{t}}(\sum\{\mathsf{p}\}\cup P)}$

Table 5: Transition rules for $\_ \xrightarrow{-}_{r\rho} \_ \subseteq (\mathrm{P}\times\mathrm{A}\times\mathrm{P})$ and $\_ \xrightarrow{-}_{r\rho} \surd \subseteq (\mathrm{P}\times\mathrm{A})$, and delay predicates $U_{\mathsf{t}} \subseteq \mathrm{P}$ for the elements t of T. ($\mathsf{a}\in\mathrm{A}$ and $\mathsf{p},\mathsf{q}\in\mathrm{P}$ and $P\subseteq\mathrm{P}$ and $\mathsf{t},\mathsf{u}\in\mathrm{T}$)

Proof. Straightforward.

Lemma 2.10 Every (deadlock-saturated) basic term p is derivably equal to a (deadlock-saturated) well-timed basic term.

Proof. We apply induction on the structure of p. If p is of type 3, then it is well-timed by induction and Lemma 2.7. If p is of type 1, then it is trivially well-timed. Let p be of type 2, and write $p \equiv \sum_v a(t)p' \lhd b \rhd \delta$, where p' is a well-timed basic term. By AT2, we find $p = \sum_v a(t)(t \gg p') \lhd b \rhd \delta$. The righthand-side is well-timed by Lemma 2.8. By Lemmas 2.3 and 2.9 it is derivably equal to a well-timed basic term q. By Lemma 2.4, we find that q is deadlock-saturated, if p is deadlock-saturated. □

Theorem 2.2 (Completeness) If the data types have built-in equality and Skolem functions, and E is the equational theory of the data types, then $\mathfrak{P}/\leftrightarrow_\rho \models p = q$ implies $p\mathrm{CRL} + E + \mathrm{AE} + \mathrm{SCA} + \mathrm{AT} \vdash p = q$, for all $\mathbb{p}$-ground process terms p and q.

Proof. We may assume that p, q are well-timed deadlock-saturated deadlock-terminating basic terms by Lemmas 2.6 and 2.10. From the assumption $\mathfrak{P}/\leftrightarrow_\rho \models p = q$, we know by Lemma 2.2 that $\mathfrak{P}(\mathrm{A}_{td})/\leftrightarrow \models p = q$. Then by Theorem 1.1, it holds that $p\mathrm{CRL} + E + \mathrm{AE} + \mathrm{SCA} \vdash p = q$. □

3 Relative time

We define $p\mathrm{CRL}_{r\rho}$ algebras as $p\mathrm{CRL}_\rho$ algebras without initialisation operators. As before, we let A_{td} be the set $\mathrm{A}\cup\{\delta(\mathsf{t}) \mid \mathsf{t}\in\mathrm{T}\}$, and use $\mathsf{a},\mathsf{b},\ldots$ as names for arbitrary elements of A_{td}. We may write $\mathsf{a}[\mathsf{t}]$ to refer to an action a with time-stamp t.

We let the time stamp be a relative reference to time. For example, a process $\mathsf{a}[\mathsf{t}]\cdot\mathsf{b}[\mathsf{u}]$ executes the action $\mathsf{a}[\mathsf{t}]$ at the moment t time after its moment of initialisation (that is determined by its environment). Upon the termination of $\mathsf{a}[\mathsf{t}]$ at this moment, the action $\mathsf{b}[\mathsf{u}]$ is executed u time later.

We take the positive real numbers as time domain, i.e. for $p\mathrm{CRL}_{r\rho}$ algebras it is assumed that T is the set of positive real numbers.

The transition relations and the delay predicates are defined in Table 5. The transition rules are those of untimed pCRL (The processes $\delta[\mathsf{t}]$ do not have outgoing transitions). The rules for the delay predicates are the same as in the absolute time variant (without rule for the initialisation operators).

Definition 3.1 A binary relation $\mathcal{R}$ on P is a *$r\rho$-bisimulation*, if it is symmetric, and whenever $\mathsf{p}\mathcal{R}\mathsf{q}$ and $\mathsf{a}\in\mathrm{A}$, then

1. if $\mathsf{p} \xrightarrow{\mathsf{a}}_{r\rho} \surd$, then $\mathsf{q} \xrightarrow{\mathsf{a}}_{r\rho} \surd$,
2. if $\mathsf{p} \xrightarrow{\mathsf{a}}_{r\rho} \mathsf{p}'$, then $\mathsf{q} \xrightarrow{\mathsf{a}}_{r\rho} \mathsf{q}'$, for some q' with $\mathsf{p}'\mathcal{R}\mathsf{q}'$,
3. if $U_{\mathsf{t}}(\mathsf{p})$ for some t, then $U_{\mathsf{t}}(\mathsf{q})$.

Processes p and q are *$r\rho$-bisimilar*, notation $\mathsf{p} \leftrightarrow_{r\rho} \mathsf{q}$, if there is a $r\rho$-bisimulation $\mathcal{R}$ with $\mathsf{p}\mathcal{R}\mathsf{q}$.

We state that $\leftrightarrow_{r\rho}$ is a congruence. The proof is similar to the proof of Theorem 2.1.

3.1 Strongly bisimilar timed processes

The processes of a $p\mathrm{CRL}_{r\rho}$ algebra are precisely the processes of the untimed pCRL algebra based on the set of actions A_{td}. As we did for $p\mathrm{CRL}_{\rho}$, we shall exploit this fact in the completeness proof. This proof is easier than it was for the absolute time variant, because the well-timedness of processes with relative time-stamps is immediate.

Definition 3.2 A process p is *deadlock-saturated*, if whenever $U_{\mathsf{t}}(\mathsf{p})$, then $\mathsf{p} \xrightarrow{\delta[\mathsf{t}]} \surd$ or $\mathsf{p} \xrightarrow{\delta[\mathsf{t}]} \mathsf{p}'$ for some p', and whenever $\mathsf{p} \xrightarrow{\mathsf{a}}_{r\rho} \mathsf{p}'$, then p' is deadlock-saturated.

A process p is *deadlock-terminating*, if whenever $\mathsf{p} \xrightarrow{\mathsf{a}} \mathsf{p}'$, then $\mathsf{a} \in \mathrm{A}$ and p' is deadlock-terminating.

We state without proof that for every process p, there is a process q that is deadlock-saturated and deadlock-terminating, such that $\mathsf{p} \leftrightarrow_{r\rho} \mathsf{q}$, and that for all p and q it holds that $\mathsf{p} \leftrightarrow \mathsf{q}$ implies $\mathsf{p} \leftrightarrow_{r\rho} \mathsf{q}$.

Lemma 3.1 If processes p and q are deadlock-saturated and deadlock-terminating, then $\mathsf{p} \leftrightarrow_{r\rho} \mathsf{q}$ implies $\mathsf{p} \leftrightarrow \mathsf{q}$.

Proof. Suppose that $\mathsf{p} \leftrightarrow_{r\rho} \mathsf{q}$. We apply induction on the complexity of p.

1. if $\mathsf{p} \xrightarrow{\mathsf{a}[\mathsf{t}]} \surd$, then, if $\mathsf{a}[\mathsf{t}] \neq \delta[\mathsf{t}]$, it follows from $\mathsf{p} \leftrightarrow_{r\rho} \mathsf{q}$ that $\mathsf{q} \xrightarrow{\mathsf{a}[\mathsf{t}]} \surd$. Now suppose that $\mathsf{a}[\mathsf{t}] = \delta[\mathsf{t}]$. From $U_{\mathsf{t}}(\mathsf{p})$ and $\mathsf{p} \leftrightarrow_{r\rho} \mathsf{q}$, it follows that $U_{\mathsf{t}}(\mathsf{q})$. From $U_{\mathsf{t}}(\mathsf{q})$ and the deadlock-saturatedness of q it follows that $\mathsf{q} \xrightarrow{\mathsf{a}[\mathsf{t}]} \surd$.
2. if $\mathsf{p} \xrightarrow{\mathsf{a}[\mathsf{t}]} \mathsf{p}'$, then, since p is deadlock-terminating, we have that $\mathsf{a}[\mathsf{t}] \neq \delta[\mathsf{t}]$. Therefore, by $\mathsf{p} \leftrightarrow_{r\rho} \mathsf{q}$ we find that $\mathsf{q} \xrightarrow{\mathsf{a}[\mathsf{t}]} \mathsf{q}'$ for some q' with $\mathsf{p}'\mathcal{R}\mathsf{q}'$. So $\mathsf{p}' \leftrightarrow_{r\rho} \mathsf{q}'$, and then by induction $\mathsf{p}' \leftrightarrow \mathsf{q}'$. □

3.2 The theory $p\mathrm{CRL}_{r\rho}$

A $p\mathrm{CRL}_{r\rho}$ signature is a $p\mathrm{CRL}_{\rho}$ signature without the declaration for $\gg$. We write square brackets around the time stamp, i.e. if a is an action term with time stamp t, then we may write $a[t]$ to refer to a. We write $\delta[t]$ for process terms $\delta(t)$. The relative time axioms are those of pCRL plus the two axioms in Table 6. The soundness proof is a straightforward exercise.

The interpretation of $\mathbb{p}$-ground process terms is the same as the interpretation of absolute time process terms without initialisation.

The definition of the set of $p\mathrm{CRL}_{r\rho}$ basic terms is exactly as the definition of the set of $p\mathrm{CRL}_{\rho}$ basic terms.

Lemma 3.2 For every $\mathbb{p}$-ground process term p there is a basic term q such that $p\mathrm{CRL}+\mathrm{RT2} \vdash p = q$.

Proof. Similar to the proof of Lemma 2.3.

(RT1)	$a[t]$	$=$	$a[t] + \sum_u \delta[u] \lhd u \leq t \rhd \delta$
(RT2)	$\delta[t]x$	$=$	$\delta[t]$

Table 6: pCRL$_{r\rho}$ axioms. $a \in A_{td}$.

3.3 Completeness

We show that every $\mathbb{p}$-ground process term is derivably equal to a term that is interpreted as a deadlock-saturated deadlock-terminating process. Then completeness follows from Lemma 3.1 and Theorem 1.1.

A $\mathbb{p}$-ground process term p is deadlock-saturated, if $[\![p]\!]^f$ is deadlock-saturated for every f. Deadlock-terminating process terms are defined similarly. Every basic term is deadlock-terminating.

Lemma 3.3 If p is deadlock-saturated and $p\text{CRL} + \text{RT2} \vdash p = q$, for some $\mathbb{p}$-ground q, then q is deadlock-saturated.

Proof. Straightforward.

Lemma 3.4 If $\mathbb{p}$-ground process terms p and q are deadlock-saturated, then $p + q$ is deadlock-saturated.

Proof. Straightforward.

Lemma 3.5 Every $\mathbb{p}$-ground process term p is derivably equal to a deadlock-saturated basic term.

Proof. By Lemma 3.2, we may assume that p is basic. We apply induction on the structure of p. If p is of type 3, then we use induction and Lemma 3.4. If p is of type 1, then let $p \equiv \sum_{\vec{v}} a[t] \lhd b \rhd \delta$ and derive using axiom RT1 that p equals

$$\textstyle\sum_{\vec{v}}(a[t] + \sum_u \delta[u] \lhd u \leq t \rhd \delta) \lhd b \rhd \delta,$$

which is deadlock-saturated. It is derivably equal to a deadlock-saturated basic term by Lemmas 3.2 and 3.3. If p is of type 2, then let $p \equiv \sum_{\vec{v}} a[t]p' \lhd b \rhd \delta$ with p' deadlock-saturated, and derive using RT1 that p equals

$$\textstyle\sum_{\vec{v}}(a[t] + \sum_u \delta[u] \lhd u \leq t \rhd \delta)p' \lhd b \rhd \delta,$$

which is deadlock-saturated. It is derivably equal to a deadlock-saturated basic term by Lemmas 3.2 and 3.3. □

Theorem 3.1 (Completeness) If the data types have built-in equality and Skolem functions, and E is the equational theory of the data types, then $\mathfrak{P}/\underline{\leftrightarrow}_{r\rho} \models p = q$ implies $p\text{CRL}+E+\text{AE}+\text{SCA}+\text{RT1,2} \vdash p = q$, for all $\mathbb{p}$-ground process terms p and q.

Proof. By Lemma 3.5, we may assume that p, q are deadlock-saturated deadlock-terminating basic terms. From the assumption $\mathfrak{P}/\underline{\leftrightarrow}_{r\rho} \models p = q$ we know by Lemma 3.1 that $\mathfrak{P}(\text{A}_{td})/\underline{\leftrightarrow} \models p = q$. Then by Theorem 1.1, it holds that $p\text{CRL} + E + \text{AE} + \text{SCA} \vdash p = q$. □

4 Conclusions

We have presented two extensions of pCRL with time-stamped actions: $p\mathrm{CRL}_\rho$ for absolute time and $p\mathrm{CRL}_{r\rho}$ for relative time. We defined timed bisimilarity for both versions and proved that the given axiomatisations are ground complete. We based the completeness proofs on the completeness results for untimed pCRL [6]. We inherited from [6] the proviso that the data types have built-in equality and Skolem functions.

We conclude that the integration of data and processes already present in pCRL, makes pCRL very suitable for the extension with time, as time is treated as just another data type. The alternative quantification over data and the conditional construct allow a simple yet powerful and well-understood means to describe time-dependent processes. Furthermore, most results of the various studies into timed ACP translate directly to our framework. Therefore, we think that the presented theories provide an elegant basis for the further study of timed process algebras.

Finally, we remark that the fact that deadlock-saturated, deadlock-terminating and – in the absolute time variant – well-timed and initialisation-free processes can be regarded as untimed processes, makes it possible to use the verification tools that exist for μCRL, for the analysis of timed processes.

Acknowledgements The author thanks Wan Fokkink, Jan Friso Groote, Bas Luttik, Alban Ponse and Michel Reniers for comments. Research supported by the Netherlands Organisation for Scientific Research (NWO) under contract SION 612-61-002.

References

[1] J.A. Bergstra and J.W. Klop. Process algebra for synchronous communication. *Information and Computation*, 60(1/3):109–137, 1984.

[2] J.C.M. Baeten and J.A. Bergstra. Real time process algebra. *Formal Aspects of Computing*, 3(2):142–188, 1991.

[3] J.C.M. Baeten and J.A. Bergstra. Real space process algebra. *Formal Aspects of Computing*, 5(6):481–592, 1993.

[4] J.C.M. Baeten and J.A. Bergstra. Discrete time process algebra. *Formal Aspects of Computing*, 8(2):188–208, 1996.

[5] J.F. Groote and A. Ponse. The syntax and semantics of μCRL. In A. Ponse, C. Verhoef and S.F.M. van Vlijmen, Editors, *Algebra of Communicating Processes '94*, Workshops in Computing Series, Springer Verlag, p. 26–62, 1995.

[6] J.F. Groote and S.P. Luttik. Undecidability and completeness results for process algebras with alternative quantification over data. Technical Report SEN-R9806, CWI, Amsterdam, 1998.