Information Processing Letters 80 (2001) 41–49

Information Processing Letters

www.elsevier.com/locate/ipl

Process algebra and conditional composition

Jan A. Bergstra [a,b,*], Alban Ponse [a,c]

[a] University of Amsterdam, Programming Research Group, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands

[b] Utrecht University, Department of Philosophy, Heidelberglaan 8, 3584 CS Utrecht, The Netherlands

[c] CWI, Department of Software Engineering, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands

Received 21 July 2000; received in revised form 30 December 2000

Abstract

We discern three non-classical truth values, and define a five-valued propositional logic. We combine this logic with process algebra via conditional composition (i.e., if-then-else-). In particular, the choice operation (+) is regarded as a special case of conditional composition. We present an operational semantics in SOS-style and some completeness results.

Keywords: Process algebra; Conditional composition; Five-valued logic

1. Introduction

Assume P represents some program (or algorithm). Then the initial behavior of the *conditional* program

`if` ϕ `then` P `else` P

depends on evaluation of the condition ϕ: either it yields an immediate error, or it starts with performing P, or it diverges in evaluation of ϕ. The following three non-classical truth values for ϕ are sufficient to accommodate these intuitions:

Meaningless, notation M. Typical examples are errors that are detectable during execution such as a type-clash or division by zero.

Choice or *undetermined*, notation C. This value represents 'being either true or false'. An example is as above: `if` ϕ `then` P `else` P represents the same behavior as P.

Divergent or *undefined*, notation D. Typically, evaluation of a partial predicate can diverge.

We describe a five-valued propositional logic that incorporates these three non-classical truth values next to *true* (notation T) and *false* (notation F). Furthermore, we define a generalization of process algebra that is based on conditional composition over this logic.

This paper is a successor of [6], in which ACP with five-valued conditions is introduced. In Section 5 we discuss the main differences with [6].

2. Five-valued logic

The five truth values discerned above can be arranged in the partial ordering given in Fig. 1. Let $x \sqcup y$ stand for the least upper bound of x and y. So, $\mathsf{T} \sqcup \mathsf{F} = \mathsf{F} \sqcup \mathsf{T} = \mathsf{C}$, and $x \sqcup y \in \{x, y\}$ for all other pairs. Furthermore, each truth value can be described with $\sqcup$ and the *deterministic* truth values M, T, F and D.

* Corresponding author.

E-mail address: janb@science.uva.nl (J.A. Bergstra).

0020-0190/01/$ – see front matter
PII: S0020-0190(01)00216-2

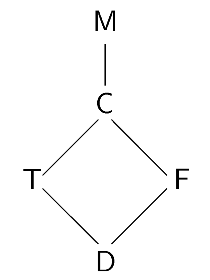

Fig. 1. Five ordered truth values.

We first consider a single, ternary operation on these five truth values: *conditional composition*, notation $x \triangleleft y \triangleright z$ (this notation stems from [10], modeling `if` y `then` x `else` z). Conditional composition is defined as follows:

$x \triangleleft \mathsf{M} \triangleright y = \mathsf{M}$,

$x \triangleleft \mathsf{C} \triangleright y = x \sqcup y$,

$x \triangleleft \mathsf{T} \triangleright y = x$,

$x \triangleleft \mathsf{F} \triangleright y = y$,

$x \triangleleft \mathsf{D} \triangleright y = \mathsf{D}$.

Notice that $x \triangleleft \mathsf{C} \triangleright y$ (as a binary operation) is idempotent, commutative, and associative. Furthermore, we have the following convenient distributivity property:

Proposition 1. *Conditional composition distributes over* $\sqcup$: *let* $\boldsymbol{v}$ *abbreviate* $v_1 \sqcup v_2$, *then*

$$\begin{aligned} \boldsymbol{x} \triangleleft \boldsymbol{y} \triangleright \boldsymbol{z} &= (x_1 \triangleleft \boldsymbol{y} \triangleright \boldsymbol{z}) \sqcup (x_2 \triangleleft \boldsymbol{y} \triangleright \boldsymbol{z}) \\ &= (\boldsymbol{x} \triangleleft y_1 \triangleright \boldsymbol{z}) \sqcup (\boldsymbol{x} \triangleleft y_2 \triangleright \boldsymbol{z}) \\ &= (\boldsymbol{x} \triangleleft \boldsymbol{y} \triangleright z_1) \sqcup (\boldsymbol{x} \triangleleft \boldsymbol{y} \triangleright z_2). \end{aligned}$$

As a consequence, conditional composition is monotonic.

Next to conditional composition, we consider the following logical operations (cf. [2,6]): *negation*, *left-sequential conjunction* and *symmetric* (or *strict parallel*) *conjunction*. Negation on the newly added non-classical values can be explained from the intuitions provided earlier: $\neg\mathsf{M} = \mathsf{M}$ because the negation of an immediate error is one as well. Since C means "being either true or false", so does its negation, thus $\neg\mathsf{C} = \mathsf{C}$. Furthermore, as D represents divergence, so does $\neg\mathsf{D}$, hence $\neg\mathsf{D} = \mathsf{D}$. With $\mathbin{{}_{\circ}\!\wedge}$ we denote left-sequential conjunction, i.e., McCarthy's left to right conjunction [12], adopting the asymmetric notation from [2]. First the left argument is evaluated, and depending on the result of this, possibly the right argument. This yields $x \mathbin{{}_{\circ}\!\wedge} y = x$ for $x \in \{\mathsf{M}, \mathsf{F}, \mathsf{D}\}$, and $\mathsf{T} \mathbin{{}_{\circ}\!\wedge} x = x$. The values of $\mathsf{C} \mathbin{{}_{\circ}\!\wedge} x$ are given below. Finally, symmetric conjunction on the newly added truth values appears to be captured by

$x \wedge y = (x \mathbin{{}_{\circ}\!\wedge} y) \sqcup (y \mathbin{{}_{\circ}\!\wedge} x)$.

Left sequential disjunction, notation $\mathbin{{}^{\circ}\!\vee}$, and *symmetric disjunction* ($\vee$) are defined as expected:

$x \mathbin{{}^{\circ}\!\vee} y = \neg(\neg x \mathbin{{}_{\circ}\!\wedge} \neg y)$,

$x \vee y = \neg(\neg x \wedge \neg y)$.

The complete truth tables for $\neg$, $\mathbin{{}_{\circ}\!\wedge}$, and $\wedge$ are the following:

	M	C	T	F	D
¬	M	C	F	T	D

$\mathbin{{}_{\circ}\!\wedge}$	M	C	T	F	D
M	M	M	M	M	M
C	M	C	C	F	F
T	M	C	T	F	D
F	F	F	F	F	F
D	D	D	D	D	D

$\wedge$	M	C	T	F	D
M	M	M	M	M	M
C	M	C	C	F	F
T	M	C	T	F	D
F	M	F	F	F	F
D	M	F	D	F	D

These truth tables were also presented in [6], and, when omitting C, coincide with the definitions given in [2]. Note that $\mathbin{{}_{\circ}\!\wedge}$ and its dual $\mathbin{{}^{\circ}\!\vee}$ are idempotent and associative.

In the following we establish the relation between conditional composition and the operations just discussed.

Proposition 2. *The operations* $\neg$, $\mathbin{{}_{\circ}\!\wedge}$ *and* $\wedge$ *are definable from conditional composition*:

$$\begin{aligned} \neg x &= \mathsf{F} \triangleleft x \triangleright \mathsf{T}, \\ x \mathbin{{}_{\circ}\!\wedge} y &= y \triangleleft x \triangleright \mathsf{F}, \\ x \wedge y &= (x \mathbin{{}_{\circ}\!\wedge} y) \triangleleft \mathsf{C} \triangleright (y \mathbin{{}_{\circ}\!\wedge} x). \end{aligned}$$

Furthermore,

$$\begin{aligned} x \triangleleft y \triangleright z &= z \triangleleft \neg y \triangleright x, \\ \neg(x \triangleleft y \triangleright z) &= \neg x \triangleleft y \triangleright \neg z. \end{aligned}$$

Corollary 3. *The operations* ${}^{\circ}\!\!\vee$ *and* $\vee$ *can be defined by*:

$$x \,{}^{\circ}\!\!\vee\, y = \mathsf{T} \lhd x \rhd y,$$
$$x \vee y = (x \,{}^{\circ}\!\!\vee\, y) \lhd \mathsf{C} \rhd (y \,{}^{\circ}\!\!\vee\, x).$$

Furthermore, $\neg$, ${}_{\circ}\!\!\wedge$, $\wedge$, ${}^{\circ}\!\!\vee$ *and* $\vee$ *distribute over* $\sqcup$, *and all these operations are monotonic.*

Conversely, $x \lhd \mathsf{C} \rhd y$ can be defined by $(\mathsf{C} \wedge x) \vee (\mathsf{C} \wedge y) \vee (x \wedge y)$. This leads to the following result:

Proposition 4. *Conditional composition* $x \lhd y \rhd z$ *can be defined from* $\neg$, ${}_{\circ}\!\!\wedge$ *and* $\wedge$ *by*

$$x \lhd y \rhd z = \mathcal{E} \lhd \mathsf{C} \rhd \mathcal{F},$$

where $x \lhd \mathsf{C} \rhd y$ *is given above, and*

$$\mathcal{E} = (y \vee \mathsf{D}) \,{}_{\circ}\!\!\wedge\, (x \,{}^{\circ}\!\!\vee\, \mathcal{G}),$$
$$\mathcal{F} = (\neg y \vee \mathsf{D}) \,{}_{\circ}\!\!\wedge\, (z \,{}_{\circ}\!\!\wedge\, \mathcal{H}),$$
$$\mathcal{G} = (y \,{}_{\circ}\!\!\wedge\, x) \vee (\neg y \,{}_{\circ}\!\!\wedge\, z),$$
$$\mathcal{H} = (\neg y \,{}^{\circ}\!\!\vee\, x) \wedge (y \,{}^{\circ}\!\!\vee\, z).$$

We denote the resulting five-valued logic by

$$\mathcal{L}_5(\neg, {}_{\circ}\!\!\wedge, \wedge) \quad \text{or} \quad \mathcal{L}_5(\_ \lhd \_ \rhd \_),$$

or shortly $\mathcal{L}_5$ whenever we do not care which operations are considered primitive.

Following McCarthy and Hayes [13], let $f, g, \ldots$ be names for *fluents*, i.e., objects that in any state (i.e., at each instance of time) may take a deterministic value, thus a value in $\{\mathsf{M}, \mathsf{T}, \mathsf{F}, \mathsf{D}\}$. Let $\mathbb{P}_4$ be a set of fluents. We write $\mathcal{L}_5(\mathbb{P}_4)$ for the extension of $\mathcal{L}_5$ with the fluents in $\mathbb{P}_4$. In order to equate propositions in $\mathcal{L}_5(\mathbb{P}_4)$ we use substitution of fluents: for $f, g \in \mathbb{P}_4$,

$$[\phi/f]f \stackrel{\triangle}{=} \phi, \qquad [\phi/f]g \stackrel{\triangle}{=} g,$$
$$[\phi/f]c \stackrel{\triangle}{=} c \quad \text{for } c \in \{\mathsf{M}, \mathsf{C}, \mathsf{T}, \mathsf{F}, \mathsf{D}\},$$
$$[\phi/f](\psi_1 \lhd \psi_2 \rhd \psi_3) \stackrel{\triangle}{=}$$
$$[\phi/f]\psi_1 \lhd [\phi/f]\psi_2 \rhd [\phi/f]\psi_3\,,$$

and as a proof rule the *excluded fifth rule* (cf. [5]):

$$\frac{[c/f]\phi = [c/f]\psi \quad \text{for } c \in \{\mathsf{M}, \mathsf{T}, \mathsf{F}, \mathsf{D}\}}{\phi = \psi}.$$

By Proposition 2 it follows that substitution distributes over the other logical operations in the expected way. Together with the identities generated by the truth tables this yields a complete evaluation system for equations over $\mathcal{L}_5(\mathbb{P}_4)$. We write $\mathcal{L}_5(\mathbb{P}_4) \models \phi = \psi$ or shortly $\models \phi = \psi$, if $\phi = \psi$ follows from the system defined above and the truth tables for $\mathcal{L}_5(\mathbb{P}_4)$.

3. A generalization of BPA with five-valued conditions

Let A be a set of constants $a, b, c, \ldots$ denoting atomic actions (atoms), i.e., processes that are not subject to further division, and that execute in finite time. We consider a generalized version of $\mathsf{BPA}_{\delta,\mu}(A)$, i.e., Basic Process Algebra (see, e.g., [3,1,8]) extended with $\delta \notin A$ (*inaction* or *deadlock*) and with $\mu \notin A$. The *meaningless* process μ represents the operational contents of M, and is introduced in [4,5]. We use the notation $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ for a generalization of $\mathsf{BPA}_{\delta,\mu}(A)$ in which alternative composition is a special case of conditional composition over $\mathcal{L}_5(\mathbb{P}_4)$ (various other generalizations are conceivable). The operations of $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ are:

Sequential composition: $X \cdot Y$ denotes the process that performs X, and upon completion of X starts with Y.

Conditional composition: $X +_\phi Y$ with $\phi \in \mathcal{L}_5(\mathbb{P}_4)$ denotes the process that either performs X or Y, or represents δ or μ, depending on the value of ϕ (which may depend on some valuation). (Conditional composition $X +_\phi Y$ is often denoted $X \lhd \phi \rhd Y$, cf. [10].)

We mostly suppress the $\cdot$ in process expressions, and brackets according to the rule that $\cdot$ binds strongest. Accommodating to classical process algebra, we shall often use the *abbreviation* $+$ for $+_{\mathsf{C}}$ (modeling 'alternative composition' or 'choice'), thus $X + Y$ is short for $X +_{\mathsf{C}} Y$.

In Table 1 we give the rule of equivalence (ROE) and the axioms of $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$.

Example 5. In $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ one easily derives

$$X + \delta = X,$$
$$X + X = X,$$
$$X + \mu = \mu.$$

Table 1
Rule of equivalence and axioms of $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$

(ROE)	$\mathcal{L}_5(\mathbb{P}_4) \models \phi = \psi \quad \Rightarrow \quad X +_\phi Y = X +_\psi Y$		
(CM)	$X +_{\mathsf{M}} Y = \mu$	(GA1)	$X +_\phi (Y +_\phi Z) = (X +_\phi Y) +_\phi Z$
(CC)	$(X +_\phi X') +_{\mathsf{C}} (Y +_\phi Y') =$	(GA2)	$X +_\phi Y = Y +_{\neg\phi} X$
	$(X +_{\mathsf{C}} Y) +_\phi (X' +_{\mathsf{C}} Y')$	(GA3)	$(X +_\phi Y) +_\psi (X +_\chi Y) = X +_{(\phi \lhd \psi \rhd \chi)} Y$
(CT)	$X +_{\mathsf{T}} Y = X$	(GA4)	$(X +_\phi Y)Z = XZ +_\phi YZ$
(CD)	$X +_{\mathsf{D}} Y = \delta$	(GA5)	$(XY)Z = X(YZ)$

(Use axiom GA3 with $\phi = \mathsf{T}$, $\psi = \mathsf{C}$, and $\chi = \mathsf{D}, \mathsf{T}, \mathsf{M}$, respectively).

Furthermore, with $\models \mathsf{T} \lhd \phi \rhd \phi = \phi$ and GA3 it follows that

$$X +_\phi (X +_\phi Y) = X +_\phi Y.$$

Finally, Proposition 2 implies that

$$X +_{(\phi \mathbin{{}_\circ\!\wedge} \psi)} Y = (X +_\psi Y) +_\phi Y.$$

With δ, the *conditional guard construct* from [7] (called *guarded command* in that paper, and roughly expressing an `if_then_`construct) can be defined as a special case of conditional composition:

$$\phi :\rightarrow X \triangleq X +_\phi \delta.$$

Example 6. With the axioms CC, GA2, and $X + \delta = X$ we find

$$X +_\phi Y = (\phi :\rightarrow X) + (\neg\phi :\rightarrow Y).$$

An intricate identity is $(\phi \vee \psi) :\rightarrow X = \phi :\rightarrow X + \psi :\rightarrow X$. First we derive

$$\begin{aligned} X +_{(\phi \vee \psi)} \delta &= (X +_{(\phi \vee \psi)} \delta) + (X +_{\mathsf{F}} \delta) \\ &= X +_{((\phi \vee \psi) \lhd \mathsf{C} \rhd \mathsf{F})} \delta. \end{aligned}$$

In a similar way it follows that

$$X +_{(\phi \lhd \mathsf{C} \rhd \psi)} \delta = X +_{((\phi \lhd \mathsf{C} \rhd \psi) \lhd \mathsf{C} \rhd \mathsf{F})} \delta.$$

Because $\models (\phi \vee \psi) \lhd \mathsf{C} \rhd \mathsf{F} = (\phi \lhd \mathsf{C} \rhd \psi) \lhd \mathsf{C} \rhd \mathsf{F}$, we can apply the rule of equivalence (ROE).

Closed terms over $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ will be further called *process terms*. We provide an operational semantics for process terms. Given a (non-empty) set $\mathbb{P}_4$ of fluents, let w range over $\mathcal{W}$, the *valuations* (interpretations) of $\mathbb{P}_4$ in $\{\mathsf{M}, \mathsf{T}, \mathsf{F}, \mathsf{D}\}$. Valuations are extended to propositions in the usual way. In Table 2 we define for each $w \in \mathcal{W}$ a unary predicate *meaningless*, notation $\mu(w, \_)$, over process terms in $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$. This predicate defines whether a process term represents the meaningless process μ under valuation w.

Table 2
Rules for the $\mu(w, \_)$ predicate

$$\mu(w,\mu) \qquad \mu(w, X +_\phi Y) \quad \text{if } w(\phi) = \mathsf{M}$$

$$\frac{\mu(w, X)}{\left\{ \begin{array}{ll} \mu(w, X +_\phi Y) & \text{if } w(\phi) = \mathsf{C}, \mathsf{T}, \\ \mu(w, Y +_\phi X) & \text{if } w(\phi) = \mathsf{C}, \mathsf{F}, \\ \mu(w, X \cdot Y) & \end{array} \right\}}$$

The axioms and rules for $\mu(w, \_)$ given in Table 2 are extended by those given in Table 3, which define transitions $\_ \xrightarrow{w,a} \_$ as a binary relation on process terms, and unary "tick-predicates" or "termination transitions" $\_ \xrightarrow{w,a} \surd$, where w ranges over $\mathcal{W}$ and a over A. Transitions characterize under which interpretations a process term defines the possibility to execute an atomic action, and what remains to be executed (if anything, otherwise $\surd$ symbolizes successful termination). Note that if a process term P has a transition $P \xrightarrow{w,a} \cdots$, then $\neg\mu(w, P)$.

The axioms and rules in Tables 2 and 3 yield a structured operational semantics (SOS) with negative premises in the style of [9]. Moreover, they satisfy the so called *panth-format* [15]. Using [9,15], it is easy to establish that the meaningless instances and transitions defined by these rules are uniquely determined, and go with the following notion of bisimulation equivalence:

Table 3
Transition rules in *panth*-format

$a \in A$	$a \xrightarrow{w,a} \surd$	
$+_\phi$	$\dfrac{X \xrightarrow{w,a} \surd,\quad \neg\mu(w,Y)}{\left\{\begin{array}{ll} X +_\phi Y \xrightarrow{w,a} \surd & \text{if } w(\phi)=\mathsf{C},\\ Y +_\phi X \xrightarrow{w,a} \surd & \text{if } w(\phi)=\mathsf{C}\end{array}\right\}}$	$\dfrac{X \xrightarrow{w,a} X',\quad \neg\mu(w,Y)}{\left\{\begin{array}{ll} X +_\phi Y \xrightarrow{w,a} X' & \text{if } w(\phi)=\mathsf{C},\\ Y +_\phi X \xrightarrow{w,a} X' & \text{if } w(\phi)=\mathsf{C}\end{array}\right\}}$
$\cdot, +_\phi$	$\dfrac{X \xrightarrow{w,a} \surd}{\left\{\begin{array}{ll} X \cdot Y \xrightarrow{w,a} Y, & \\ X +_\phi Y \xrightarrow{w,a} \surd & \text{if } w(\phi)=\mathsf{T},\\ Y +_\phi X \xrightarrow{w,a} \surd & \text{if } w(\phi)=\mathsf{F}\end{array}\right\}}$	$\dfrac{X \xrightarrow{w,a} X'}{\left\{\begin{array}{ll} X \cdot Y \xrightarrow{w,a} X' \cdot Y, & \\ X +_\phi Y \xrightarrow{w,a} X' & \text{if } w(\phi)=\mathsf{T},\\ Y +_\phi X \xrightarrow{w,a} X' & \text{if } w(\phi)=\mathsf{F}\end{array}\right\}}$

Definition 7. A binary relation B over process terms is a *bisimulation* if for all $P,\ Q$ with PBQ the following conditions hold for all $w \in \mathcal{W}$ and $a \in A$:

- $\mu(w,P) \ \Leftrightarrow\ \mu(w,Q)$,
- $P \xrightarrow{w,a} \surd \ \Leftrightarrow\ Q \xrightarrow{w,a} \surd$,
- $\forall P'\,(P \xrightarrow{w,a} P' \ \Rightarrow\ \exists Q'(Q \xrightarrow{w,a} Q' \wedge P'BQ'))$,
- $\forall Q'\,(Q \xrightarrow{w,a} Q' \ \Rightarrow\ \exists P'(P \xrightarrow{w,a} P' \wedge P'BQ'))$.

Two processes $P,\ Q$ are *bisimilar*, notation $P \leftrightarrow Q$, if there exists a bisimulation containing the pair (P,Q).

By the main result in [15] it follows that bisimilarity is a *congruence* relation for all operations involved. Note that conditional composition constructs are considered binary operations: for each $\phi \in \mathcal{L}_5(\mathbb{P}_4)$ there is an operation $+_\phi$.

We write $G/_{\leftrightarrow} \models P = Q$ whenever $P \leftrightarrow Q$ according to the notions just defined, and for $\vec{X} = X_1,\ldots,X_n$, $G/_{\leftrightarrow} \models t(\vec{X}) = t'(\vec{X})$ if for all $\vec{P} = P_1,\ldots,P_n$ it holds that $G/_{\leftrightarrow} \models t(\vec{P}) = t'(\vec{P})$. It is not difficult to show that in the bisimulation model thus obtained all equations of Table 1 are true. Hence we conclude:

Lemma 8 (Soundness). *If* $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A)) \vdash t(\vec{X}) = t'(\vec{X})$, *then* $G/_{\leftrightarrow} \models t(\vec{X}) = t'(\vec{X})$.

Finally, we provide a completeness result for $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$. Our proof refers to the completeness result in [5], which is based on a representation of process terms for which bisimilarity implies derivability in a straightforward way.

Definition 9. A process term P over $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ is a *generalized basic term* if it is of the form

$$P ::= \delta \mid \mu \mid a \mid aP \mid P +_\phi P,$$

where $a \in A$ and $\phi \in \mathcal{L}_5(\mathbb{P}_4)$.

Lemma 10. *Each process term over* $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ *is provably equal to a generalized basic term.*

In the following we relate process terms over $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ with terms over $\mathsf{BPA}_{\delta,\mu}(A)$ extended with conditional guard constructs, of which the conditions are in

$$\mathcal{L}_4(\mathbb{P}_4) \stackrel{\triangle}{=} \mathcal{L}_{\{\mathsf{M},\mathsf{T},\mathsf{F},\mathsf{D}\}}(\mathbb{P}_4, \neg, \mathbin{\overset{\circ}{\wedge}}, \wedge),$$

thus $\mathcal{L}_5(\mathbb{P}_4)$ without C. The only operations of $\mathsf{BPA}_{\delta,\mu}(A)$ are sequential composition and the choice operation $+$, i.e., the operation $+_{\mathsf{C}}$. In the following, finite sums $P_1 + P_2 + \cdots + P_n$ are abbreviated by $\sum_{i=1}^n P_i$.

Let the symbol $\equiv$ denote syntactic equivalence, and let $\mathcal{L} \subseteq \mathcal{L}_5(\mathbb{P}_4)$.

Definition 11. A process term P over $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ is called an $\mathcal{L}$-*basic term* if $P \equiv$

$\sum_{i\in I}\phi_i :\to Q_i$, where I is a finite, non-empty index set, $\phi_i \in \mathcal{L}$, and $Q_i \in \{\delta, a, aR \mid a \in A,\ R$ an $\mathcal{L}$-basic term$\}$.

Lemma 12. *Each process term over $G_{\mathcal{L}_5}(\mathsf{BPA}_{\delta,\mu}(A))$ is provably equal to an $\mathcal{L}_4(\mathbb{P}_4)$-basic term.*

Proof. By Lemma 10 it is sufficient to consider generalized basic terms. Then, representation easily follows for $\mathcal{L}_5(\mathbb{P}_4)$-basic terms by induction (where axiom CC is needed, cf. footnotes 5, 6 in Section 5). It remains to be shown that each $\mathcal{L}_5(\mathbb{P}_4)$-basic term is provably equal to one in which C does not occur in any conditional guard construct. As $\mathsf{C} :\to X = X$, this follows easily by induction on the complexity of the guard ϕ in $\phi :\to X$. □

Theorem 13. *The system $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ is complete with respect to bisimulation equivalence.*

Proof. By Lemmas 12 and 8 it is sufficient to prove that bisimilarity between $\mathcal{L}_4(\mathbb{P}_4)$-basic terms implies their provable equality. A detailed (inductive) proof is spelled out in [5], which is also sufficient as all axioms of Basic Process Algebra with four-valued logic are derivable from $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ (the less trivial ones were derived in Examples 5 and 6). □

4. A generalization of ACP with five-valued conditions

We extend $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ to a generalized version of $\mathsf{ACP}(A,|)$ (Algebra of Communicating Processes, see, e.g., [3,1,8]) by including encapsulation and parametrized merge operations ${}_\phi\!\diamond_\psi$. In the latter, ϕ covers the choice between interleaving and synchronization, and ψ determines the order of interleaving and synchronization:

Parametrized merge: $X\,{}_\phi\|_\psi\,Y$ denotes the parallel execution of X and Y under conditions ϕ and ψ.

Parametrized left merge, an auxiliary operator: $X\,{}_\phi⫗_\psi\,Y$ denotes $X\,{}_\phi\|_\psi\,Y$ with the restriction that the first action stems from X.

Parametrized communication merge, an auxiliary operator: $X\,{}_\phi|_\psi\,Y$ denotes $X\,{}_\phi\|_\psi\,Y$ with the restriction that the first action is a synchronization of both X and Y.

Parametrized left communication merge, an auxiliary operator: $X\,{}_\phi\lfloor_\psi\,Y$ is used to define $X\,{}_\phi|_\psi\,Y$.

Encapsulation: $\partial_H(X)$ (where $H \subseteq A$) renames atoms in H to δ.

In $\mathsf{ACP}(A,|)$, the commutative and associative communication function $| : A \times A \to A \cup \{\delta\}$ is given (and extended to process terms). The axioms of our generalization of $\mathsf{ACP}(A,|)$ are those of $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{BPA}_{\delta,\mu}(A))$ (including ROE) and those in Table 4. We adopt the convention that $+_\phi$ binds weakest and $\cdot$ binds strongest, and denote the resulting system by $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{ACP}(A,|))$. We note that the $\|$ operation of $\mathsf{ACP}(A,|)$ equals ${}_\mathsf{C}\|_\mathsf{C}$. Furthermore, the operation ${}_\mathsf{T}\|_\mathsf{C}$ restricts $\|$ to interleaving only, while ${}_\mathsf{F}\|_\diamond$ for $\diamond \in \{\mathsf{C},\mathsf{T},\mathsf{F}\}$ defines "synchronous ACP" and ${}_\mathsf{T}\|_\mathsf{T}$ represents sequential composition. Some typical $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{ACP}(A,|))$ identities are:

$$X\,{}_\phi\|_\psi\,Y = Y\,{}_\phi\|_{\neg\psi}\,X,$$
$$X\,{}_\phi|_\psi\,Y = Y\,{}_\phi|_{\neg\psi}\,X,$$
$$\mu\,{}_\phi|_\psi\,\delta = \mu +_\psi \delta,$$
$$\mu\,{}_\phi|_\psi\,a = \mu +_\psi \mu \quad (a \in A),$$
$$\delta\,{}_\phi|_\psi\,a = \delta +_\psi \delta \quad (a \in A).$$

In Table 5 we give additional rules for the meaningless predicate defined in Table 2 and the transition rules defined in Table 3. We stick to bisimulation equivalence as defined in Definition 7, and as before it follows that bisimilarity is a congruence for all operations involved. It is not difficult (but tedious) to establish that in the bisimulation model thus obtained all equations of Table 4 are true. Furthermore, each process term over $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{ACP}(A,|))$ is provably equal to a generalized basic term (see Definition 9). Hence:

Theorem 14. *The system $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{ACP}(A,|))$ is complete with respect to bisimulation equivalence.*

5. Conclusions

In this paper we have shown that process algebra can be viewed from a logical perspective that comprises the truth values *choice* C and *divergent*

Table 4
Additional axioms of $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{ACP}(A,|))$, $a,b,c \in A$ and $H \subseteq A$

(C1)	$a \mid b = b \mid a$	(GD1)	$\partial_H(a) = a \quad \text{if } a \notin H$
(C2)	$(a \mid b) \mid c = a \mid (b \mid c)$	(GD2)	$\partial_H(a) = \delta \quad \text{if } a \in H$
		(GD3)	$\partial_H(X +_\phi Y) = \partial_H(X) +_\phi \partial_H(Y)$
		(GD4)	$\partial_H(XY) = \partial_H(X)\partial_H(Y)$

(GCM1)	$X \,{}_\phi\|_\psi\, Y = (X \,{}_\phi⫫_\psi\, Y +_\psi Y \,{}_\phi⫫_{\neg\psi}\, X) +_\phi X \,{}_\phi|_\psi\, Y$
(GCM2)	$a \,{}_\phi⫫_\psi\, X = aX$
(GCM3)	$aX \,{}_\phi⫫_\psi\, Y = a(X \,{}_\phi\|_\psi\, Y)$
(GCM4)	$(X +_\phi Y) \,{}_\psi⫫_\chi\, Z = X \,{}_\psi⫫_\chi\, Z +_\phi Y \,{}_\psi⫫_\chi\, Z$
(GCMC)	$X \,{}_\phi|_\psi\, Y = X \,{}_\phi\lfloor_\psi\, Y +_\psi Y \,{}_\phi\lfloor_{\neg\psi}\, X$
(GCM5)	$aX \,{}_\phi\lfloor_\psi\, Y = a \,{}_\phi\lfloor_\psi\, (Y \,{}_\phi⫫_{\neg\psi}\, X)$
(GCM6)	$a \,{}_\phi\lfloor_\psi\, b = a \mid b$
(GCM7)	$a \,{}_\phi\lfloor_\psi\, bX = (a \mid b)X$
(GCM8)	$a \,{}_\phi\lfloor_\psi\, (X +_\chi Y) = a \,{}_\phi\lfloor_\psi\, X +_\chi a \,{}_\phi\lfloor_\psi\, Y$
(GCM9)	$(X +_\phi Y) \,{}_\psi\lfloor_\chi\, Z = X \,{}_\psi\lfloor_\chi\, Z +_\phi Y \,{}_\psi\lfloor_\chi\, Z$

D, and the basic operations *conditional composition* and *sequential composition*. For instance, the axiom $X +_{\mathsf{D}} Y = \delta$ expresses that δ is associated with "divergence". This may seem incompatible with the usual "deadlock" interpretation (modeled by the standard axioms $X + \delta = X$ and $\delta X = \delta$), but can be clarified as follows: in order to support an axiomatic approach to the interleaving hypothesis, [1] the operation $+$ models "optimistic choice" in the sense that δ-alternatives are discarded ($X + \delta = X$). E.g., the derivation $ab\|\delta = a(b\delta + \delta b) + \delta ab = ab\delta$ shows that δ has an aspect of divergence: the deadlock in $ab\|\delta$ is postponed until all concurrent behavior has been executed.

In the following we shortly discuss the main differences between this paper and [6]. Taking four-valued logic over {M, T, F, D} [2,14] and its combination with process algebra [5] as a point of departure, the contribution of [6] can be characterized as follows:

- The introduction of C as a 'natural' truth value [2] and the associated logic $\mathcal{L}_5(\mathbb{P}_4)$.
- The introduction of conditional composition as a definable operation in $\mathcal{L}_5(\mathbb{P}_4)$.
- An axiomatization of $\mathsf{ACP}(A,|)$ with conditional guard construct over $\mathcal{L}_5(\mathbb{P}_4)$, [3] going with an operational semantics and a completeness result.
- A generalization of $\mathsf{ACP}(A,|)$: the $+$ and merge operators can be *parameterized* with propositions over $\mathcal{L}_5(\mathbb{P}_4)$ (or one of its sublogics containing C, T and F). [4]

The present paper records a non-trivial extension of our understanding of $\mathcal{L}_5(\mathbb{P}_4)$, and of its combination with process algebra:

- We show that $\mathcal{L}_5(\mathbb{P}_4, \neg, \,_{\circ}\!\wedge, \wedge)$ and $\mathcal{L}_5(\mathbb{P}_4, \_ \lhd \_ \rhd \_)$ are interdefinable (Propositions 2 and 4).
- We provide operational semantics and (ground complete) axiomatizations for our $\mathcal{L}_5(\mathbb{P}_4)$-generalizations of $\mathsf{BPA}_\delta(A)$ and $\mathsf{ACP}(A,|)$ (Theorems 13 and 14). [5]

Inspired by one of the referees, we end with some considerations about a six-valued logic. By symmetry

[1] I.e., concurrency can be analyzed in terms of all possible interleavings.

[2] This establishes a second intuition for Kleene's third truth value (D modeling the first). We note that a complete axiomatization of Kleene's three-valued logic [11] admits exactly *two* non-classical truth values, the conjunction of which must equal F.

[3] These axioms are derivable in $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{ACP}(A,|))$.

[4] For the case {C, T, F} we give the axioms. For the proposed generalization to $\mathcal{L}_5(\mathbb{P}_4)$ it must be required that $a \neq \delta$ in axiom GCM8.

[5] In [6] we have a less general version of axiom GA3, which requires separate axioms $X + \delta = X$ and $X + \mu = \mu$. Moreover, in [6] the axiom CC (see Table 1) is neither present, nor derivable.

Table 5
Additional meaningless and transition rules for $G_{\mathcal{L}_5(\mathbb{P}_4)}(\mathsf{ACP}(A,|))$ in *panth*-format

$\mu(w, X\,{}_{\phi}\|_{\psi}\,Y)\quad$ if $w(\psi \lhd \phi \rhd \psi) = \mathsf{M}$	$\mu(w, X\,{}_{\phi}|_{\psi}\,Y)\quad$ if $w(\psi) = \mathsf{M}$

$$\frac{\mu(w, X)}{\left\{\begin{array}{l} \mu(w, X\,{}_{\phi}\lfloor\!\lfloor_{\psi}\,Y),\ \mu(w, X\,{}_{\phi}\lfloor_{\psi}\,Y),\\ \mu(w, a\,{}_{\phi}\lfloor_{\psi}\,X),\ \mu(w, aY\,{}_{\phi}\lfloor_{\psi}\,X),\\ \mu(w, \partial_H(X)) \end{array}\right\}}
\qquad
\frac{\mu(w, X),\quad w(\psi) = \mathsf{C}}{\left\{\begin{array}{ll} \mu(w, X\,{}_{\phi}\|_{\psi}\,Y) & \text{if } w(\phi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ \mu(w, Y\,{}_{\phi}\|_{\psi}\,X) & \text{if } w(\phi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ \mu(w, X\,{}_{\phi}|_{\psi}\,Y),\ \mu(w, Y\,{}_{\phi}|_{\psi}\,X) & \end{array}\right\}}$$

$$\frac{\mu(w, X),\quad w(\psi) = \mathsf{T}}{\left\{\begin{array}{ll} \mu(w, X\,{}_{\phi}\|_{\psi}\,Y) & \text{if } w(\phi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ \mu(w, a\,{}_{\phi}\|_{\psi}\,X) & \text{if } w(\phi) = \mathsf{C},\mathsf{F},\\ \mu(w, aY\,{}_{\phi}\|_{\psi}\,X) & \text{if } w(\phi) = \mathsf{C},\mathsf{F},\\ \mu(w, X\,{}_{\phi}|_{\psi}\,Y), & \\ \mu(w, a\,{}_{\phi}|_{\psi}\,X),\ \mu(w, aY\,{}_{\phi}|_{\psi}\,X) & \end{array}\right\}}
\qquad
\frac{\mu(w, X),\quad w(\psi) = \mathsf{F}}{\left\{\begin{array}{ll} \mu(w, Y\,{}_{\phi}\|_{\psi}\,X) & \text{if } w(\phi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ \mu(w, X\,{}_{\phi}\|_{\psi}\,a) & \text{if } w(\phi) = \mathsf{C},\mathsf{F},\\ \mu(w, X\,{}_{\phi}\|_{\psi}\,aY) & \text{if } w(\phi) = \mathsf{C},\mathsf{F},\\ \mu(w, Y\,{}_{\phi}|_{\psi}\,X), & \\ \mu(w, X\,{}_{\phi}|_{\psi}\,a),\ \mu(w, X\,{}_{\phi}|_{\psi}\,aY) & \end{array}\right\}}$$

$$\frac{X \xrightarrow{w,a} \surd,\ \neg\mu(w,Y),\ w(\phi) = \mathsf{C},\mathsf{T}}{\left\{\begin{array}{ll} X\,{}_{\phi}\|_{\psi}\,Y \xrightarrow{w,a} Y & \text{if } w(\psi) = \mathsf{C},\mathsf{T},\\ Y\,{}_{\phi}\|_{\psi}\,X \xrightarrow{w,a} Y & \text{if } w(\psi) = \mathsf{C},\mathsf{F} \end{array}\right\}}
\qquad
\frac{X \xrightarrow{w,a} X',\ \neg\mu(w,Y),\ w(\phi) = \mathsf{C},\mathsf{T}}{\left\{\begin{array}{ll} X\,{}_{\phi}\|_{\psi}\,Y \xrightarrow{w,a} X'\,{}_{\phi}\|_{\psi}\,Y & \text{if } w(\psi) = \mathsf{C},\mathsf{T},\\ Y\,{}_{\phi}\|_{\psi}\,X \xrightarrow{w,a} Y\,{}_{\phi}\|_{\psi}\,X' & \text{if } w(\psi) = \mathsf{C},\mathsf{F} \end{array}\right\}}$$

$$\frac{X \xrightarrow{w,a} \surd,\ Y \xrightarrow{w,b} \surd,\ a \mid b = c}{\left\{\begin{array}{ll} X\,{}_{\phi}\|_{\psi}\,Y \xrightarrow{w,c} \surd & \text{if } w(\phi) = \mathsf{C},\mathsf{F}\\ & \text{and } w(\psi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ X\,{}_{\phi}|_{\psi}\,Y \xrightarrow{w,c} \surd & \text{if } w(\psi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ X\,{}_{\phi}\lfloor_{\psi}\,Y \xrightarrow{w,c} \surd & \end{array}\right\}}
\qquad
\frac{X \xrightarrow{w,a} \surd,\ Y \xrightarrow{w,b} Y',\ a \mid b = c}{\left\{\begin{array}{ll} X\,{}_{\phi}\|_{\psi}\,Y \xrightarrow{w,c} Y' & \text{if } w(\phi) = \mathsf{C},\mathsf{F}\\ & \text{and } w(\psi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ X\,{}_{\phi}|_{\psi}\,Y \xrightarrow{w,c} Y' & \text{if } w(\psi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ X\,{}_{\phi}\lfloor_{\psi}\,Y \xrightarrow{w,c} Y' & \end{array}\right\}}$$

$$\frac{X \xrightarrow{w,a} X',\ Y \xrightarrow{w,b} \surd,\ a \mid b = c}{\left\{\begin{array}{ll} X\,{}_{\phi}\|_{\psi}\,Y \xrightarrow{w,c} X' & \text{if } w(\phi) = \mathsf{C},\mathsf{F}\\ & \text{and } w(\psi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ X\,{}_{\phi}|_{\psi}\,Y \xrightarrow{w,c} X' & \text{if } w(\psi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ X\,{}_{\phi}\lfloor_{\psi}\,Y \xrightarrow{w,c} X' & \end{array}\right\}}
\qquad
\frac{X \xrightarrow{w,a} X',\ Y \xrightarrow{w,b} Y',\ a \mid b = c}{\left\{\begin{array}{ll} X\,{}_{\phi}\|_{\psi}\,Y \xrightarrow{w,c} X'\,{}_{\phi}\|_{\psi}\,Y' & \text{if } w(\phi) = \mathsf{C},\mathsf{F}\\ & \text{and } w(\psi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ X\,{}_{\phi}|_{\psi}\,Y \xrightarrow{w,c} X'\,{}_{\phi}\|_{\psi}\,Y' & \text{if } w(\psi) = \mathsf{C},\mathsf{T},\mathsf{F},\\ X\,{}_{\phi}\lfloor_{\psi}\,Y \xrightarrow{w,c} X'\,{}_{\phi}\|_{\psi}\,Y' & \end{array}\right\}}$$

$$\frac{X \xrightarrow{w,a} \surd}{\left\{\begin{array}{ll} X\,{}_{\phi}\lfloor\!\lfloor_{\psi}\,Y \xrightarrow{w,a} Y, & \\ \partial_H(X) \xrightarrow{w,a} \surd & \text{if } a \notin H \end{array}\right\}}
\qquad
\frac{X \xrightarrow{w,a} X'}{\left\{\begin{array}{ll} X\,{}_{\phi}\lfloor\!\lfloor_{\psi}\,Y \xrightarrow{w,a} X'\,{}_{\phi}\|_{\psi}\,Y, & \\ \partial_H(X) \xrightarrow{w,a} \partial_H(X') & \text{if } a \notin H \end{array}\right\}}$$

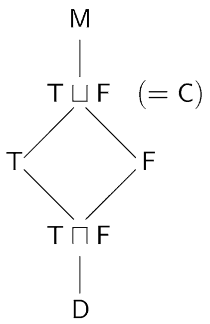

Fig. 2. Six ordered truth values.

one can distinguish a greatest lower bound (notation $\sqcap$) of T and F that majorizes D (see Fig. 2), and extend the definition of conditional composition with $x \lhd (\mathsf{T} \sqcap \mathsf{F}) \rhd y = x \sqcap y$. This yields a six-valued logic in which the identities $\mathsf{F} \lhd (\mathsf{T} \sqcap \mathsf{F}) \rhd \mathsf{F} = \mathsf{F}$ and $\mathsf{F} \lhd \mathsf{D} \rhd \mathsf{F} = \mathsf{D}$ illustrate the difference between $\mathsf{T} \sqcap \mathsf{F}$ and D. Although this logic is simple and elegant (e.g., conditional composition also distributes over $\sqcap$, cf. Proposition 1), we see no algorithmic motive for distinguishing D and $\mathsf{T} \sqcap \mathsf{F}$. We can employ process algebraic conditional composition to support this position: by $x \lhd (\mathsf{T} \sqcap \mathsf{F}) \rhd x = x$ we obtain the associated identity $X +_{\mathsf{T} \sqcap \mathsf{F}} X = X$, and by $x \lhd (\mathsf{T} \sqcap \mathsf{F}) \rhd \mathsf{D} = \mathsf{D}$ we find $X +_{\mathsf{T} \sqcap \mathsf{F}} \delta = \delta$. This illustrates that the operation $+_{\mathsf{T} \sqcap \mathsf{F}}$ models a notion of choice, say “pessimistic choice”, for which we have no useful intuition or application. [6]

Acknowledgements

We thank the referees for useful and inspiring comments.

[6] When combining this six-valued logic with process algebra, the axiom CC (see Table 1) appears to be the only one that should be changed: it allows one to derive undesirable identities, such as $a +_{\mathsf{T} \sqcap \mathsf{F}} b = (a + X) +_{\mathsf{T} \sqcap \mathsf{F}} b$. We note that CC is crucial for Lemma 12, and thereby for our completeness results.

References

[1] J.C.M. Baeten, W.P. Weijland, Process Algebra, Cambridge Tracts in Theoretical Computer Science 18, Cambridge University Press, 1990.

[2] J.A. Bergstra, I. Bethke, P.H. Rodenburg, A propositional logic with 4 values: true, false, divergent and meaningless, J. Appl. Non-Classical Logics 5 (1995) 199–217.

[3] J.A. Bergstra, J.W. Klop, Process algebra for synchronous communication, Inform. and Control 60 (1–3) (1984) 109–137.

[4] J.A. Bergstra, A. Ponse, Bochvar–McCarthy logic and process algebra, Notre Dame J. Formal Logic 39 (4) (1998) 464–484.

[5] J.A. Bergstra, A. Ponse, Process algebra with four-valued logic, J. Appl. Non-Classical Logics 10 (1) (2000) 27–53.

[6] J.A. Bergstra, A. Ponse, Process algebra with five-valued conditions, in: C.S. Calude, M.J. Dinneen (Eds.), Combinatorics, Complexity, and Logic, Australian Computer Science Communications, Vol. 21, Springer, Singapore, 1999, pp. 128–143.

[7] E.W. Dijkstra, Guarded commands, nondeterminacy and formal derivation of programs, Comm. ACM 18 (8) (1975) 453–457.

[8] W.J. Fokkink, Introduction to Process Algebra, Springer, Berlin, 2000.

[9] J.F. Groote, Transition system specifications with negative premises, Theoret. Comput. Sci. 118 (2) (1993) 263–299.

[10] C.A.R. Hoare, I.J. Hayes, H. Jifeng, C.C. Morgan, A.W. Roscoe, J.W. Sanders, I.H. Sorensen, J.M. Spivey, B.A. Sufrin, Laws of programming, Comm. ACM 30 (8) (1987) 672–686.

[11] S.C. Kleene, On a notation for ordinal numbers, J. Symbolic Logic 3 (1938) 150–155.

[12] J. McCarthy, A basis for a mathematical theory of computation, in: P. Braffort, D. Hirshberg (Eds.), Computer Programming and Formal Systems, North-Holland, Amsterdam, 1963, pp. 33–70.

[13] J. McCarthy, P.J. Hayes, Some philosophical problems from the standpoint of artificial intelligence, in: B. Meltzer, D. Michie (Eds.), Machine Intelligence 4, Edinburgh University Press, 1969, pp. 463–502, Reprinted in: V. Lifschitz (Ed.), Formalization of Common Sense, Papers by John McCarthy, Ablex, Norwood, NJ, 1990.

[14] P.H. Rodenburg, A complete system of four-valued logic, Report P9616, University of Amsterdam, 1996; J. Appl. Non-Classical Logics, to appear.

[15] C. Verhoef, A congruence theorem for structured operational semantics with predicates and negative premises, Nordic J. Comput. 2 (2) (1995) 274–302.