Proposition Algebra

JAN A. BERGSTRA and ALBAN PONSE, University of Amsterdam

Sequential propositional logic deviates from conventional propositional logic by taking into account that during the sequential evaluation of a propositional statement, atomic propositions may yield different Boolean values at repeated occurrences. We introduce "free valuations" to capture this dynamics of a propositional statement's environment. The resulting logic is phrased as an equationally specified algebra rather than in the form of proof rules, and is named "proposition algebra." It is strictly more general than Boolean algebra to the extent that the classical connectives fail to be expressively complete in the sequential case. The four axioms for free valuation congruence are then combined with other axioms in order define a few more valuation congruences that gradually identify more propositional statements, up to static valuation congruence (which is the setting of conventional propositional logic).

Proposition algebra is developed in a fashion similar to the process algebra ACP and the program algebra PGA, via an algebraic specification which has a meaningful initial algebra for which a range of coarser congruences are considered important as well. In addition, infinite objects (i.e., propositional statements, processes and programs respectively) are dealt with by means of an inverse limit construction which allows the transfer of knowledge concerning finite objects to facts about infinite ones while reducing all facts about infinite objects to an infinity of facts about finite ones in return.

21

Categories and Subject Descriptors: F.3.2 [**Logics and Meanings of Programs**]: Semantics of Programming Languages—*Algebraic approaches to semantics*

General Terms: Theory; Verification

Additional Key Words and Phrases: Conditional composition, propositional statement, reactive valuation, satisfiability, sequential connective, short-circuit evaluation, side effect

ACM Reference Format:
Bergstra, J. A. and Ponse, A. 2011. Proposition algebra. ACM Trans. Comput. Logic 12, 3, Article 21 (May 2011), 36 pages.
DOI = 10.1145/1929954.1929958 http://doi.acm.org/10.1145/1929954.1929958

1. INTRODUCTION

A propositional statement is a composition of atomic propositions made by means of one or more (proposition) composition mechanisms, usually called connectives. Atomic propositions are considered to represent facts about an environment (execution environment, execution architecture, operating context) that are used by the logical mechanism contained in the propositional statement which aggregates these facts for presentation to the propositional statement's user. Different occurrences of the same

J. A. Bergstra acknowledges support from NWO (project Thread Algebra for Strategic Interleaving).
Authors' address: J. A. Bergstra and A. Ponse, Section Theory of Computer Science, Informatics Institute, University of Amsterdam, Amsterdam, The Netherlands, email: {J.A.Bergstra,A.Ponse}@uva.nl.

DOI 10.1145/1929954.1929958 http://doi.acm.org/10.1145/1929954.1929958

atomic propositions represent different queries (measurements, issued information requests) at different moments in time.

A valuation that may return different Boolean values for the same atomic proposition during the sequential evaluation of a single propositional statement is called *free*, or in the case the evaluation result of an atomic proposition can have effect on subsequent evaluation, it is called *reactive*. This is in contrast to a "static" valuation, which always returns the same value for the same atomic proposition. Free valuations are thus semantically at the opposite end of static valuations, and are *observation based* in the sense that they capture the identity of a propositional statement as a pattern of queries followed by a Boolean value.

Many classes of valuations can be distinguished. Given a class K of valuations, two propositional statements are K-equivalent if they evaluate to the same Boolean value for each valuation in K. Given a family of proposition connectives, K-equivalence need not be a congruence, and K-congruence is the largest congruence that is contained in K-equivalence. It is obvious that with larger K more propositional statements can be distinguished and the one we consider most distinguishing is named *free valuation congruence*. It is this congruence that plays the role of an initial algebra for the proposition algebras developed in this article. The axioms of proposition algebra specify free valuation congruence in terms of the single ternary connective *conditional composition* (in computer science terminology: *if-then-else*) and constants for truth and falsity, and their soundness and completeness (for closed equations) is easily shown. Additional axioms are given for static valuation congruence, and for some reactive valuation congruences in between.

Sequential versions of the well-known binary connectives of propositional logic and negation can be expressed in terms of conditional composition. We prove that these connectives have insufficient expressive power at this level of generality and that a ternary connective is needed (in fact, this holds for any collection of binary connectives definable by conditional composition.)

Repeated use of the same atomic proposition is meaningful in free or reactive valuation semantics, and as a consequence infinite propositions are meaningful and may be more expressive than finite ones. Infinite propositions are defined by means of an inverse limit construction which allows the transfer of knowledge concerning finite objects to facts about infinite ones while reducing all facts about infinite objects to an infinity of facts about finite ones in return. This construction was applied in giving standard semantics for the process algebra ACP by Bergstra and Klop [1984] (see Baeten and Weijland [1990] for a more recent overview). In doing so, the design of proposition algebra is very similar to the thread algebra of Bergstra and Middelburg [2007] which is based on a similar ternary connective but which features constants for termination and deadlock rather than for truth and falsity. Whereas thread algebra focuses on multi-threading and concurrency, proposition algebra has a focus on sequential mechanisms.

The article is structured as follows: In the next section, we discuss some motivation for proposition algebra. In Section 3, we define the signature and equations of proposition algebra, and in Section 4, we formally define valuation algebras. In Section 5, we consider some observation based equivalences and congruences generated by valuations, and in Sections 6-9, we provide complete axiomatizations of these congruences. Definable (binary) connectives are formally introduced in Section 10. In Section 11, we briefly consider some complexity issues concerning satisfiability. The expressiveness (functional incompleteness) of binary connectives is discussed in Section 12. In Section 13, we introduce projection and projective limits for defining potentially infinite propositions, and in Section 14, we discuss recursive specifications of infinite propositions. The article is ended with some conclusions in Section 15.

2. MOTIVATION FOR PROPOSITION ALGEBRA

Proposition algebra is proposed as a preferred way of viewing the data type of propositional statements, at least in a context of sequential systems. Here are some arguments in favor of that thesis:

In a sequential program a test, which is a conjunction of P and Q will be evaluated in a sequential fashion, beginning with P and not evaluating Q unless the evaluation of P led to a positive outcome. The sequential form of evaluation takes precedence over the axioms or rules of conventional propositional logic or Boolean algebra. For instance, neither conjunction nor disjunction are commutative when evaluated sequentially in the presence of side-effects, errors or exceptions. The absence of these latter features is never claimed for imperative programming and thus some extension or modification of ordinary two-valued logic is necessary to understand the basics of propositional logic as it occurs in the context of imperative programs. Three-, four- or more sophisticated many-valued logics may be used to explain the logic in this case (see, e.g., Bergstra et al. [1995], Bergstra and Ponse [1998b], and Hähnle [2005]). The noncommutative, sequential reading of conjunction mentioned above can be traced back to the seminal work on computation theory by McCarthy [1963], in which a specific value for undefinedness (e.g., a divergent computation) is considered that in conjunction with falsity results in the value that was evaluated first. In many explanations of the semantics of Boolean operators in programming languages, this form of sequential evaluation is called *short-circuit evaluation* (and in some, *minimal evaluation* or *McCarthy evaluation*).

Importing noncommutative conjunction to two-valued propositional logic means that the sequential order of events is significant, and that is what proposition algebra is meant to specify and analyze in the first place. As a simple example, consider the propositional statement that a pedestrian evaluates just before crossing a road with two-way traffic driving on the right:

$$\textit{look-left-and-check} \mathbin{{}_{\circ}\!\wedge} \textit{look-right-and-check} \mathbin{{}_{\circ}\!\wedge} \textit{look-left-and-check}. \tag{1}$$

Here ${}_{\circ}\!\wedge$ is *left-sequential conjunction*, which is similar to conjunction but the left argument is evaluated first and upon F ("false"), evaluation finishes with result F. A valuation associated with this example is (should be) a free valuation: also in the case that the leftmost occurrence of *look-left-and-check* evaluates to T ("true"), its second evaluation might very well evaluate to F. However, the order of events (or their amount) needs not to be significant in all circumstances and one may still wish or require that in particular cases conjunction is idempotent or even commutative. A most simple example is perhaps

$$a \mathbin{{}_{\circ}\!\wedge} a = a$$

with a an atomic proposition, which is not valid in free valuation semantics (and neither is the falsity of $a \mathbin{{}_{\circ}\!\wedge} \neg a$). For this reason we distinguish a number of restricted forms of reactive valuation equivalences and congruences that validate this example or variations thereof, but still refine static valuation congruence. It is evident that many more such refinements can be distinguished.

We take the point of departure that the very purpose of any action taken by a program under execution is to change the state of a system. If no change of state results with certainty the action can just as well be skipped. This holds for tests as well as for any action mainly performed because of its intended side-effects. The common intuition that the state is an external matter not influenced by the evaluation of a test, justifies ignoring side-effects of tests and for that reason it justifies an exclusive focus on static valuations to a large extent, thereby rendering the issue of reactivity pointless as well. But there are some interesting cases where this intuition is not necessarily convincing.

We mention three such issues, all of which also support the general idea of considering propositional statements under sequential evaluation.

(1) It is common to accept that in a mathematical text an expression x/x is admissible only after a test $x \neq 0$ has been performed. One might conceive this test as an action changing the state of mind of the reader thus influencing the evaluation of further assertions such as $x/x = 1$.
(2) A well-known dogma on computer viruses introduced by Cohen in 1984 (journal publication in [1987]) states that a computer cannot decide whether or not a program that it is running is itself a virus. The proof involves a test that is hypothetically enabled by a decision mechanism that is supposed to have been implemented on a modified instance of the machine under consideration. It seems fair to say that the property of a program being viral is not obviously independent of the state of the program. So here is a case where performing the test might (in principle at least) result in a different state from which the same test would lead to a different outcome.

    This matter has been analyzed in detail in Bergstra and Ponse [2005] and Bergstra et al. [2007] with the conclusion that the reactive nature of valuations gives room for criticism of Cohen's original argument. In the didactic literature on computer security Cohen's viewpoint is often repeated and it can be found on many websites and in the introduction of many texts. But there is a remarkable lack of secondary literature on the matter; an exception is the discussion in Cohen [2001] and the papers cited therein.
(3) The online halting problem is about execution environments that allow a running program to acquire information about its future halting or divergence. This information is supposed to be provided by means of a forecasting service. In Bergstra and Ponse [2007], that feature is analyzed in detail in a setting of thread algebra and the impossibility of sound and complete forecasting of halting is established. In particular, calling a forecasting service may have side-effects that leads to different replies in future calls (see, e.g., Ponse and van der Zwaag [2008]).

Our account of proposition algebra is based on the ternary operator *conditional composition* (or *if-then-else*). This operator has a sequential form of evaluation as its natural semantics, and thus combines naturally with free and reactive valuation semantics. Furthermore, proposition algebra constitutes a simple setting for constructing infinite propositions by means of an inverse limit construction. The resulting *projective limit model* can be judged as one that didactically precedes (prepares for) technically more involved versions for process algebra and thread algebra, and as such provides by itself a motivation for proposition algebra.

3. PROPOSITION ALGEBRA

In this section, we introduce the signature and equational axioms of proposition algebra. Let A be a countable set of atomic propositions $a, b, c, \ldots$. The elements of A serve as atomic (i.e., nondivisible) queries that will produce a Boolean reply value.

We assume that $|A| > 1$. The case that $|A| = 1$ is described in detail in Regenboog [2010]. We come back to this point in Section 15.

The signature of proposition algebra consists of the constants T and F (representing true and false), a constant a for each $a \in A$, and, following Hoare [1985b], the ternary operator *conditional composition*

$$\_ \triangleleft \_ \triangleright \_ .$$

Table I. The Set CP of Axioms for Proposition Algebra

(CP1)	$x \triangleleft T \triangleright y = x$
(CP2)	$x \triangleleft F \triangleright y = y$
(CP3)	$T \triangleleft x \triangleright F = x$
(CP4)	$x \triangleleft (y \triangleleft z \triangleright u) \triangleright v = (x \triangleleft y \triangleright v) \triangleleft z \triangleright (x \triangleleft u \triangleright v)$

We write $\Sigma_{\text{CP}}(A)$ for the signature introduced here. Terms are subject to the equational axioms in Table I. We further write CP for this set of axioms (where CP abbreviates conditional propositions).

An alternative name for the conditional composition

$$y \triangleleft x \triangleright z$$

is if x then y else z: the axioms CP1 and CP2 model that its central condition x is evaluated first, and depending on the reply either its leftmost or rightmost argument is evaluated. Axiom CP3 establishes that a term can be extended to a larger conditional composition by adding T as a leftmost argument and F as a rightmost one, and CP4 models the way a nonatomic central condition distributes over the outer arguments. We note that the expression

$$F \triangleleft x \triangleright T$$

can be seen as defining the negation of x:

$$\text{CP} \vdash z \triangleleft (F \triangleleft x \triangleright T) \triangleright y = (z \triangleleft F \triangleright y) \triangleleft x \triangleright (z \triangleleft T \triangleright y) = y \triangleleft x \triangleright z, \tag{2}$$

which illustrates that "if $\neg x$ then z else y" and "if x then y else z" are considered equal.

We introduce the abbreviation

$$x \circ y \quad \text{for} \quad y \triangleleft x \triangleright y,$$

and we name this expression x and then y. It follows easily that $\circ$ is associative:

$$(x \circ y) \circ z = z \triangleleft (y \triangleleft x \triangleright y) \triangleright z = (z \triangleleft y \triangleright z) \triangleleft x \triangleright (z \triangleleft y \triangleright z) = x \circ (y \circ z).$$

We take the and-then operator $\circ$ to bind stronger than conditional composition. At a later stage, we will formally add negation, the "and then" connective $\circ$, and some other binary connectives to proposition algebra (i.e., add their function symbols to $\Sigma_{\text{CP}}(A)$ and their defining equations to CP).

Closed terms over $\Sigma_{\text{CP}}(A)$ are called *propositional statements*, with typical elements $P, Q, R, \ldots$.

Definition 3.1. A propositional statement P is a *basic form* if

$$P ::= T \mid F \mid P_1 \triangleleft a \triangleright P_2$$

with $a \in A$, and P_1 and P_2 basic forms.

So, basic forms can be seen as binary trees of which the leaves are labeled with either T or F, and the internal nodes with atomic propositions. Following Baeten and Weijland [1990], we use the name basic form instead of normal form because we associate the latter with a term rewriting setting.

LEMMA 3.2. *Each propositional statement can be proved equal to one in basic form using the axioms in Table I.*

PROOF. We first show that if P, Q, R are basic forms, then $P \triangleleft Q \triangleright R$ can be proved equal to a basic form by structural induction on Q. If $Q = T$ or $Q = F$, this follows immediately, and if $Q = Q_1 \triangleleft a \triangleright Q_2$, then

$$\begin{aligned}\mathrm{CP} \vdash P \triangleleft Q \triangleright R &= P \triangleleft (Q_1 \triangleleft a \triangleright Q_2) \triangleright R \\ &= (P \triangleleft Q_1 \triangleright R) \triangleleft a \triangleright (P \triangleleft Q_2 \triangleright R)\end{aligned}$$

and by induction there are basic forms P_i for $i = 1, 2$ such that $\mathrm{CP} \vdash P_i = P \triangleleft Q_i \triangleright R$, hence $\mathrm{CP} \vdash P \triangleleft Q \triangleright R = P_1 \triangleleft a \triangleright P_2$ and $P_1 \triangleleft a \triangleright P_2$ is a basic form.

Next, we prove the lemma's statement by structural induction on the form that propositional statement P may take. If $P = T$ or $P = F$, then P is a basic form, and if $P = a$, then $\mathrm{CP} \vdash P = T \triangleleft a \triangleright F$. For the case $P = P_1 \triangleleft P_2 \triangleright P_3$, it follows by induction that there are basic forms Q_1, Q_2, Q_3 with $\mathrm{CP} \vdash P_i = Q_i$, so $\mathrm{CP} \vdash P = Q_1 \triangleleft Q_2 \triangleright Q_3$. Now apply the first result. □

We write

$$P \equiv Q$$

to denote that propositional statements P and Q are syntactically equivalent. In Section 6, we prove that basic forms constitute a convenient representation:

Proposition 3.3. *If* $\mathrm{CP} \vdash P = Q$ *for basic forms* P *and* Q*, then* $P \equiv Q$.

4. VALUATION ALGEBRAS

In this section, we define valuation algebras. Let B be the sort of the Booleans with constants T and F. The signature $\Sigma_{Val}(A)$ of valuation algebras contains the sort B and a sort Val of valuations. The sort Val has two constants

$$T_{Val} \quad \text{and} \quad F_{Val},$$

which represent the valuations that assign to each atomic proposition the value T respectively F, and for each $a \in A$ a function

$$y_a : Val \to B$$

called the *yield* of a, and a function

$$a \bullet \ : Val \to Val$$

called the *a-derivative*. Given a valuation H we write $a \bullet H$ (instead of $a \bullet (H)$) to denote the transformation of H after a has been evaluated.

Definition 4.1. A $\Sigma_{Val}(A)$-algebra $\mathbb{A}$ is a *valuation algebra* (VA) if for all $a \in A$, it satisfies the axioms

$$\begin{aligned}&y_a(T_{Val}) = T,\\ &y_a(F_{Val}) = F,\\ &a \bullet T_{Val} = T_{Val},\\ &a \bullet F_{Val} = F_{Val}.\end{aligned}$$

Given a valuation algebra $\mathbb{A}$, a valuation H (of sort Val) in $\mathbb{A}$, and a propositional statement P, we now define both the evaluation of P over H, notation

$$P/H$$

and the P-derivative of H, a generalized notion of the a-derivative of H, notation

$$P \bullet H.$$

Definition 4.2. Let $\mathbb{A}$ be a valuation algebra and H a valuation in $\mathbb{A}$, and let P, Q, R be propositional statements. Evaluation and (valuation) derivatives are defined by the following case distinctions:

$$\begin{aligned}
T/H &= T,\\
F/H &= F,\\
a/H &= y_a(H),\\
(P \triangleleft Q \triangleright R)/H &= \begin{cases} P/(Q \bullet H) & \text{if } Q/H = T,\\ R/(Q \bullet H) & \text{if } Q/H = F,\end{cases}
\end{aligned}$$

and

$$\begin{aligned}
T \bullet H &= H,\\
F \bullet H &= H,\\
(P \triangleleft Q \triangleright R) \bullet H &= \begin{cases} P \bullet (Q \bullet H) & \text{if } Q/H = T,\\ R \bullet (Q \bullet H) & \text{if } Q/H = F.\end{cases}
\end{aligned}$$

Some explanation: whenever in a conditional composition the central condition is an atomic proposition, say c, then a valuation H distributes over the outer arguments as $c \bullet H$, thus

$$(P \triangleleft c \triangleright Q)/H = \begin{cases} P/(c \bullet H) & \text{if } y_c(H) = T,\\ Q/(c \bullet H) & \text{if } y_c(H) = F.\end{cases} \tag{3}$$

If, in a conditional composition the central condition is not atomic, valuation decomposes further in accordance with the previous equations, for example,

$$\begin{aligned}
(a \triangleleft (b \triangleleft c \triangleright d) \triangleright e)/H &= \begin{cases} a/((b \triangleleft c \triangleright d) \bullet H) & \text{if } (b \triangleleft c \triangleright d)/H = T,\\ e/((b \triangleleft c \triangleright d) \bullet H) & \text{if } (b \triangleleft c \triangleright d)/H = F,\end{cases}\\
&= \begin{cases} a/(b \bullet (c \bullet H)) & \text{if } y_c(H) = T \text{ and } y_b(c \bullet H) = T,\\ a/(d \bullet (c \bullet H)) & \text{if } y_c(H) = F \text{ and } y_d(c \bullet H) = T,\\ e/(b \bullet (c \bullet H)) & \text{if } y_c(H) = T \text{ and } y_b(c \bullet H) = F,\\ e/(d \bullet (c \bullet H)) & \text{if } y_c(H) = F \text{ and } y_d(c \bullet H) = F.\end{cases}
\end{aligned} \tag{4}$$

We compare the last example with

$$((a \triangleleft b \triangleright e) \triangleleft c \triangleright (a \triangleleft d \triangleright e))/H, \tag{5}$$

which is a particular instance of (3). For the case $y_c(H) = T$, we find from (3) that

$$(5) = (a \triangleleft b \triangleright e)/(c \bullet H) = \begin{cases} a/(b \bullet (c \bullet H)) & \text{if } y_b(c \bullet H) = T,\\ e/(b \bullet (c \bullet H)) & \text{if } y_b(c \bullet H) = F,\end{cases}$$

and for the case $y_c(H) = F$ we find the other two right-hand sides of (4). In a similar way, it follows that

$$((a \triangleleft b \triangleright e) \triangleleft c \triangleright (a \triangleleft d \triangleright e)) \bullet H = (a \triangleleft (b \triangleleft c \triangleright d) \triangleright e) \bullet H,$$

thus providing a prototypical example of the soundness of axiom CP4 of CP.

THEOREM 4.3 (SOUNDNESS). *If for propositional statements P and Q,* CP $\vdash P = Q$, *then for each* VA $\mathbb{A}$ *and each valuation $H \in \mathbb{A}$,*

$$P/H = Q/H \quad \textit{and} \quad P \bullet H = Q \bullet H.$$

Proof. Let $\mathbb{A}$ be some VA and $H \in \mathbb{A}$. It is an easy exercise to show that an arbitrary instance $P = Q$ of one of the axioms in CP satisfies $P/H = Q/H$ and $P \bullet H = Q \bullet H$.

Assume that $\mathrm{CP} \vdash P = Q$ follows from the context rule, thus $P \equiv P_1 \triangleleft P_2 \triangleright P_3$, $Q \equiv Q_1 \triangleleft Q_2 \triangleright Q_3$, and $\mathrm{CP} \vdash P_i = Q_i$. Then, for all $H \in \mathbb{A}$, $P_i/H = Q_i/H$ and $P_i \bullet H = Q_i \bullet H$, and we find the desired result by case distinction: let $H \in \mathbb{A}$ and $P_2/H = Q_2/H = T$, then

$$(P_1 \triangleleft P_2 \triangleright P_3)/H = P_1/(P_2 \bullet H) = Q_1/(Q_2 \bullet H) = (Q_1 \triangleleft Q_2 \triangleright Q_3)/H,$$
$$(P_1 \triangleleft P_2 \triangleright P_3) \bullet H = P_1 \bullet (P_2 \bullet H) = Q \bullet_1 (Q_2 \bullet H) = (Q_1 \triangleleft Q_2 \triangleright Q_3) \bullet H.$$

For the case $P_2/H = Q_2/H = F$, a similar argument applies. □

5. VALUATION VARIETIES AND VALUATION CONGRUENCES

We introduce some specific equivalences and congruences generated by classes of valuations. The class of VAs that satisfy a certain collection of equations over $\Sigma_{Val}(A)$ is called a *valuation* variety. We distinguish the following varieties, where each next one is subvariety of the one defined.

(1) The variety of VAs with *free* valuations is defined by the equations in Definition 4.1.

(2) The variety of VAs with *repetition-proof* valuations: all VAs that satisfy for all $a \in A$,

$$y_a(x) = y_a(a \bullet x).$$

So the reply to a series of consecutive atoms a is determined by the first reply.
Typical example: $(P \triangleleft a \triangleright Q) \triangleleft a \triangleright (R \triangleleft a \triangleright S) = (a \circ P) \triangleleft a \triangleright (a \circ S)$.

(3) The variety of VAs with *contractive* valuations: all repetition-proof VAs that satisfy for all $a \in A$,

$$a \bullet (a \bullet x) = a \bullet x.$$

Each successive atom a is contracted by using the same evaluation result.
Typical example: $(P \triangleleft a \triangleright Q) \triangleleft a \triangleright (R \triangleleft a \triangleright S) = P \triangleleft a \triangleright S$.

(4) The variety of VAs with *weakly memorizing* valuations consists of all contractive VAs that satisfy for all $a, b \in A$,

$$y_b(a \bullet x) = y_a(x) \rightarrow (a \bullet (b \bullet (a \bullet x)) = b \bullet (a \bullet x) \ \wedge \ y_a(b \bullet (a \bullet x)) = y_a(x)).$$

Here the evaluation result of an atom a is memorized in a subsequent evaluation of a if the evaluation of intermediate atoms yields the same result, and this subsequent a can be contracted.
Two typical examples are

$$((P \triangleleft a \triangleright Q) \triangleleft b \triangleright R) \triangleleft a \triangleright S = (P \triangleleft b \triangleright R) \triangleleft a \triangleright S,$$
$$P \triangleleft a \triangleright (Q \triangleleft b \triangleright (R \triangleleft c \triangleright (S \triangleleft a \triangleright V))) = P \triangleleft a \triangleright (Q \triangleleft b \triangleright (R \triangleleft c \triangleright V)).$$

(5) The variety of VAs with *memorizing* valuations: all contractive VAs that satisfy for all $a, b \in A$,

$$a \bullet (b \bullet (a \bullet x)) = b \bullet (a \bullet x) \ \wedge \ y_a(b \bullet (a \bullet x)) = y_a(x).$$

Here the evaluation result of an atom a is memorized in all subsequent evaluations of a and all subsequent a's can be contracted.
Typical axiom (right-oriented version):

$$x \triangleleft y \triangleright (z \triangleleft u \triangleright (v \triangleleft y \triangleright w)) = x \triangleleft y \triangleright (z \triangleleft u \triangleright w).$$

Typical counter-example: $a \triangleleft b \triangleright F \neq b \triangleleft a \triangleright F$ (thus, $b \mathbin{\wedge\!\!\!\circ} a \neq a \mathbin{\wedge\!\!\!\circ} b$).

(6) The variety of VAs with *static* valuations: all VAs that satisfy for all $a, b \in A$,

$$y_a(b \bullet x) = y_a(x)$$

This is the setting of conventional propositional logic.
Typical identities: $a = b \circ a$ and $a \triangleleft b \triangleright F = b \triangleleft a \triangleright F$ (thus, $b \mathbin{{}_{\circ}\!\wedge} a = a \mathbin{{}_{\circ}\!\wedge} b$).

Definition 5.1. Let K be a variety of valuation algebras over A. Then, propositional statements P and Q are *K-equivalent*, notation

$$P \equiv_K Q,$$

if $P/H = Q/H$ for all $\mathbb{A} \in K$ and $H \in \mathbb{A}$. Let $=_K$ be the largest congruence contained in $\equiv_K$. Propositional statements P and Q are *K-congruent* if

$$P =_K Q.$$

From this definition, it follows that $P =_K Q$ if $P \equiv_K Q$ and for all propositonal statements R and S, the following three cases are true:

$$P \triangleleft R \triangleright S \equiv_K Q \triangleleft R \triangleright S, \tag{6}$$
$$R \triangleleft P \triangleright S \equiv_K R \triangleleft Q \triangleright S, \tag{7}$$
$$R \triangleleft S \triangleright P \equiv_K R \triangleleft S \triangleright Q. \tag{8}$$

Case (6) follows immediately from $P \equiv_K Q$: for all $\mathbb{A} \in K$ and $H \in \mathbb{A}$,

$$(P \triangleleft R \triangleright S)/H = \begin{cases} P/(R \bullet H) = Q/(R \bullet H) = (Q \triangleleft R \triangleright S)/H & \text{if } R/H = T, \\ S/(R \bullet H) = (Q \triangleleft R \triangleright S)/H & \text{if } R/H = F, \end{cases}$$

and case (8) follows in a similar way. So in order to prove $P =_K Q$ it suffices to prove that case (7) is true (note that with $R = T$ and $S = F$, case (7) implies $P \equiv_K Q$). We will often use the contraposition of this conditional property:

$$P \neq_K Q \implies \exists \mathbb{A} \in K, H \in \mathbb{A}, R, S\ (\ (R \triangleleft P \triangleright S)/H \neq (R \triangleleft Q \triangleright S)/H). \tag{9}$$

By the varieties defined thus far, we distinguish six types of K-equivalence and K-congruence: free, repetition-proof, contractive, weakly memorizing, memorizing and static. We use the following abbreviations for these:

$$K = fr,\ rp,\ cr,\ wm,\ mem,\ st,$$

respectively.

PROPOSITION 5.2. *The inclusions $\equiv_{fr} \subseteq \equiv_{rp} \subseteq \equiv_{cr} \subseteq \equiv_{wm} \subseteq \equiv_{mem} \subseteq \equiv_{st}$, and $=_K \subseteq \equiv_K$ for $K \in \{fr, rp, cr, wm, mem\}$ are all proper.*

PROOF. In this proof, we assume that all VAs we use satisfy $T \neq F$. We first consider the differences between the mentioned equivalences:

(1) $a \equiv_{rp} a \triangleleft a \triangleright F$, but $\equiv_{fr}$ does not hold in this case as is witnessed by a VA with valuation H that satisfies $y_a(H) = T$ and $y_a(a \bullet H) = F$ (yielding $a/H = T$ and $(a \triangleleft a \triangleright F)/H = F$).
(2) $b \triangleleft a \triangleright F \equiv_{cr} b \triangleleft (a \triangleleft a \triangleright F) \triangleright F$, but $\equiv_{rp}$ does not hold in the VA with element H with $y_a(H) = y_b(a \bullet H) = T$ and $y_b(a \bullet (a \bullet H)) = F$.
(3) $(a \triangleleft b \triangleright F) \triangleleft a \triangleright F \equiv_{wm} b \triangleleft a \triangleright F$, but $\equiv_{cr}$ does not hold in the VA with element H with $y_a(H) = y_b(a \bullet H) = T$ and $y_a(b \bullet (a \bullet H)) = F$.
(4) $(T \triangleleft b \triangleright (F \triangleleft a \triangleright T)) \triangleleft a \triangleright F \equiv_{mem} b \triangleleft a \triangleright F$, but $\equiv_{wm}$ does not hold in the VA with element H with $y_a(H) = T$ and $y_b(a \bullet H) = y_a(b \bullet (a \bullet H)) = F$.
(5) $a \equiv_{st} a \triangleleft b \triangleright a$ (distinguish all possible cases), but $\equiv_{mem}$ does not hold as is witnessed by the VA with element H with $y_a(H) = y_b(H) = T$ and $y_a(b \bullet H) = F$ (yielding $a/H = T$ and $(a \triangleleft b \triangleright a)/H = F$).

Finally, observe that for $K \in \{fr, rp, cr, wm, mem\}$ it holds that $T \equiv_K T \triangleleft a \triangleright T$, but $b \triangleleft T \triangleright T \not\equiv_K b \triangleleft (T \triangleleft a \triangleright T) \triangleright T$ as is witnessed by the VA with element H with $y_a(H) = y_b(H) = T$ and $y_b(a \bullet H) = F$. □

The following proposition stems from Regenboog [2010] and can be used to deal with the difference between K-congruence and K-equivalence.

PROPOSITION 5.3. *If $P \equiv_K Q$ and for all $\mathbb{A} \in K$ and $H \in \mathbb{A}$, $P \bullet H = Q \bullet H$, then $P =_K Q$.*

PROOF. Assume $P \equiv_K Q$ and the further requirement in the proposition is satisfied. As argued before (see (7)), $P =_K Q$ if for all propositional statements R and S,

$$R \triangleleft P \triangleright S \equiv_K R \triangleleft Q \triangleright S.$$

This follows easily: let $\mathbb{A} \in K$ and $H \in \mathbb{A}$ and $P \bullet H = Q \bullet H$, then

$$\begin{aligned}(R \triangleleft P \triangleright S)/H &= \begin{cases} R/(P \bullet H) & \text{if } P/H = T,\\ S/(P \bullet H) & \text{if } P/H = F,\end{cases}\\ &= \begin{cases} R/(Q \bullet H) & \text{if } Q/H = T,\\ S/(Q \bullet H) & \text{if } Q/H = F,\end{cases}\\ &= (R \triangleleft Q \triangleright S)/H.\end{aligned}$$

□

As a consequence, the soundness of CP with respect to all valuation varieties K introduced can be phrased as follows: for all propositional statements P and Q,

$$\mathrm{CP} \vdash P = Q \implies P =_K Q.$$

In particular, this holds for free valuation congruence, the most distinguishing congruence we consider, thus $\mathrm{CP} \vdash P = Q \Longrightarrow P =_{fr} Q$.

6. COMPLETENESS FOR THE VARIETIES *fr*, *rp* AND *cr*

In this section, we provide complete axiomatizations of free valuation congruence, and repetition-proof and contractive valuation congruence. We start with a basic result on free valuation congruence of basic forms.

LEMMA 6.1. *For all basic forms P and Q, $P =_{fr} Q$ implies $P \equiv Q$.*

PROOF. The implication of the lemma follows by contraposition and structural induction on P, where in each case we apply structural induction on Q.

If $P \equiv T$, then if $Q \equiv F$, $P \neq_{fr} Q$ is witnessed by each VA in which $T \neq F$. If $Q \equiv Q_1 \triangleleft a \triangleright Q_2$, consider a VA with valuation H that satisfies $y_a(H) = y_b(H) = T$ and for all propositional statements R, $y_b(R \bullet (a \bullet H)) = F$. Then $(P \circ b)/H = T$ while $(Q \circ b)/H = y_b(Q_1 \bullet (a \bullet H)) = F$, so $P \neq_{fr} Q$.

If $P \equiv F$, a similar argument applies.

If $P \equiv P_1 \triangleleft a \triangleright P_2$, then the cases $Q \equiv T$ and $Q \equiv F$ can be dealt with as above. If $Q \equiv Q_1 \triangleleft a \triangleright Q_2$, then assume $P \not\equiv Q$ because $P_1 \not\equiv Q_1$. By induction, $P_1 \neq_{fr} Q_1$. By implication (9) there exists $\mathbb{A} \in fr$ with valuation H and propositional statements R and S such that $(R \triangleleft P_1 \triangleright S)/H \neq (R \triangleleft Q_1 \triangleright S)/H$. But then there is $\mathbb{A}' \supseteq \mathbb{A}$ with valuation H' such that $y_a(H') = T$ and $a \bullet H' = H$ by which $P \neq_{fr} Q$ because $R \triangleleft P \triangleright Q$ and $R \triangleleft Q \triangleright S$ yield different results under H':

$$(R \triangleleft P \triangleright S)/H' = (R \triangleleft P_1 \triangleright S)/H \neq (R \triangleleft Q_1 \triangleright S)/H = (R \triangleleft Q \triangleright S)/H'.$$

If $P_1 \equiv P_2$, then $P_2 \not\equiv Q_2$ and a similar argument applies.

Finally, if $Q \equiv Q_1 \triangleleft b \triangleright Q_2$ with $b \neq a$, then $P \neq_{fr} Q$ follows by considering $\mathbb{A}$ with $T \neq F$ and with valuation H that satisfies $y_a(H) = y_b(H) = T$ and for all R, $y_a(R \bullet (a \bullet H)) = T$ and $y_a(R \bullet (b \bullet H)) = F$. Clearly, $(P \circ a)/H \neq (Q \circ a)/H$. □

As a corollary, we find a proof of Proposition 3.3, that is, for basic forms, provable equality in CP and syntactic equivalence coincide:

Proof of Proposition 3.3. By the soundness of CP , it is sufficient to prove that for all basic forms P and Q, $P =_{fr} Q$ implies $P \equiv Q$, and this is proved in Lemma 6.1. □

It easily follows that CP axiomatizes free valuation congruence:

Theorem 6.2 (Completeness). *If $P =_{fr} Q$ for propositional statements P and Q, then* $\mathrm{CP} \vdash P = Q$.

Proof. Assume $P =_{fr} Q$. By Lemma 3.2, there are basic forms P' and Q' with $\mathrm{CP} \vdash P = P'$ and $\mathrm{CP} \vdash Q = Q'$. By soundness, $P' =_{fr} Q'$ and by Proposition 3.3, $P' \equiv Q'$. Hence, $\mathrm{CP} \vdash P = Q$. □

We proceed by discussing completeness results for the valuation varieties rp and cr introduced in the previous section.

Write CP_{rp} for the axioms in CP and these axiom schemes ($a \in A$):

$$\text{(CPrp1)} \qquad (x \triangleleft a \triangleright y) \triangleleft a \triangleright z = (x \triangleleft a \triangleright x) \triangleleft a \triangleright z,$$
$$\text{(CPrp2)} \qquad x \triangleleft a \triangleright (y \triangleleft a \triangleright z) = x \triangleleft a \triangleright (z \triangleleft a \triangleright z).$$

Theorem 6.3. *Repetition-proof valuation congruence $=_{rp}$ is axiomatized by* CP_{rp}.

Proof. Let $\mathbb{A}$ be a VA in the variety rp of repetition-proof valuation algebras, thus for all $a \in A$,

$$y_a(x) = y_a(a \bullet x).$$

Concerning soundness, we only check axiom scheme CPrp2 (a proof for CPrp1 is very similar): let $H \in \mathbb{A}$, then

$$\begin{aligned}
(P \triangleleft a \triangleright (Q \triangleleft a \triangleright R))/H &= \begin{cases} P/(a \bullet H) & \text{if } y_a(H) = T, \\ (Q \triangleleft a \triangleright R)/(a \bullet H) & \text{if } y_a(H) = F, \end{cases} \\
&= \begin{cases} P/(a \bullet H) & \text{if } y_a(H) = T, \\ R/(a \bullet (a \bullet H)) & \text{if } y_a(H) = F = y_a(a \bullet H), \end{cases} \\
&= (P \triangleleft a \triangleright (R \triangleleft a \triangleright R))/H,
\end{aligned}$$

and

$$\begin{aligned}
(P \triangleleft a \triangleright (Q \triangleleft a \triangleright R)) \bullet H &= \begin{cases} P \bullet (a \bullet H) & \text{if } y_a(H) = T, \\ (Q \triangleleft a \triangleright R) \bullet (a \bullet H) & \text{if } y_a(H) = F, \end{cases} \\
&= \begin{cases} P \bullet (a \bullet H) & \text{if } y_a(H) = T, \\ R \bullet (a \bullet (a \bullet H)) & \text{if } y_a(H) = F = y_a(a \bullet H), \end{cases} \\
&= (P \triangleleft a \triangleright (R \triangleleft a \triangleright R)) \bullet H.
\end{aligned}$$

In order to prove completeness, we use a variant of basic forms, which we call rp-basic forms, that "minimizes" on repetition-proof valuation congruence:

—T and F are rp-basic forms, and
—$P_1 \triangleleft a \triangleright P_2$ is an rp-basic form if P_1 and P_2 are rp-basic forms, and if P_i is not equal to T or F, then either the central condition in P_i is different from a, or P_i is of the form $a \circ P'$ with P' an rp-basic form.

Each propositional statement can in CP_{rp} be proved equal to an rp-basic form by structural induction. For P and Q rp-basic forms, $P =_{rp} Q$ implies $P \equiv Q$. This follows in a similar way as in the proof of Lemma 6.1 because all valuations used in that proof can be seen as elements from VAs in rp.

(1) For the inductive case $P \equiv P_1 \triangleleft a \triangleright P_2$ and $Q \equiv Q_1 \triangleleft a \triangleright Q_2$, while $P_1 \not\equiv Q_1$ and for some valuation H and propositional statements R and S, $(R \triangleleft P_1 \triangleright S)/H \neq (R \triangleleft Q_1 \triangleright S)/H$, it is now required that for $a \bullet H' = H$ and $y_a(H') = T$, also $y_a(a \bullet H') = T$. However, since P and Q are rp-basic forms, a possible initial a-occurrence in P_1 implies $P_1 \equiv a \circ P_1'$ and similar for Q_1, so this new requirement on H' is not relevant and $(R \triangleleft P \triangleright S)/H' \neq (R \triangleleft Q \triangleright S)/H'$.

(2) For the last case $P \equiv P_1 \triangleleft a \triangleright P_2$ and $Q \equiv Q_1 \triangleleft b \triangleright Q_2$, a valuation H is used that satisfies $y_a(H) = y_a(R \bullet (a \bullet H))$ for all propositional statements R. VAs with such a valuation exist in the variety rp.

Assume $P =_{rp} Q$, so there exist rp-basic forms P' and Q' with $\mathrm{CP}_{rp} \vdash P = P'$ and $\mathrm{CP}_{rp} \vdash Q = Q'$. By soundness, $P' =_{rp} Q'$ and as previously argued, $P' \equiv Q'$. Hence, $\mathrm{CP}_{rp} \vdash P = Q$. $\square$

Write CP_{cr} for the axioms in CP and these axiom schemes ($a \in A$):

$$\text{(CPcr1)} \qquad (x \triangleleft a \triangleright y) \triangleleft a \triangleright z = x \triangleleft a \triangleright z,$$
$$\text{(CPcr2)} \qquad x \triangleleft a \triangleright (y \triangleleft a \triangleright z) = x \triangleleft a \triangleright z.$$

These schemes contract for each $a \in A$ respectively, the T-case and the F-case, and immediately imply CPrp1 and CPrp2.

THEOREM 6.4. *Contractive valuation congruence $=_{cr}$ is axiomatized by* CP_{cr}.

PROOF. Let $\mathbb{A}$ be a VA in the variety cr of contractive valuation algebras, thus for all $a \in A$,

$$a \bullet (a \bullet x) = a \bullet x \quad \text{and} \quad y_a(x) = y_a(a \bullet x).$$

Concerning soundness we only check axiom scheme CPcr1: let $H \in \mathbb{A}$, then

$$\begin{aligned}
((P \triangleleft a \triangleright Q) \triangleleft a \triangleright R)/H &= \begin{cases} (P \triangleleft a \triangleright Q)/(a \bullet H) & \text{if } y_a(H) = T, \\ R/(a \bullet H) & \text{if } y_a(H) = F, \end{cases} \\
&= \begin{cases} P/(a \bullet H) & \text{if } y_a(H) = T = y_a(a \bullet H), \\ R/(a \bullet H) & \text{if } y_a(H) = F, \end{cases} \\
&= (P \triangleleft a \triangleright R)/H,
\end{aligned}$$

and

$$\begin{aligned}
((P \triangleleft a \triangleright Q) \triangleleft a \triangleright R) \bullet H &= \begin{cases} (P \triangleleft a \triangleright Q) \bullet (a \bullet H) & \text{if } y_a(H) = T, \\ R \bullet (a \bullet H) & \text{if } y_a(H) = F, \end{cases} \\
&= \begin{cases} P \bullet (a \bullet (a \bullet H)) & \text{if } y_a(H) = T = y_a(a \bullet (a \bullet H)), \\ R \bullet (a \bullet H) & \text{if } y_a(H) = F, \end{cases} \\
&= \begin{cases} P \bullet (a \bullet H) & \text{if } y_a(H) = T = y_a(a \bullet H), \\ R \bullet (a \bullet H) & \text{if } y_a(H) = F, \end{cases} \\
&= (P \triangleleft a \triangleright R) \bullet H.
\end{aligned}$$

In order to prove completeness we again use a variant of basic forms, which we call cr-basic forms, that "minimizes" on contractive valuation congruence:

—T and F are cr-basic forms, and
—$P_1 \triangleleft a \triangleright P_2$ is a cr-basic form if P_1 and P_2 are cr-basic forms, and if P_i is not equal to T or F, the central condition in P_i is different from a.

Each propositional statement can in CP_{cr} be proved equal to a cr-basic form by structural induction. For P and Q cr-basic forms, $P =_{cr} Q$ implies $P \equiv Q$. Again, this follows in a similar way as in the proof of Lemma 6.1 because all valuations used in that proof can be seen as elements from VAs that are in cr.

(1) For the inductive case $P \equiv P_1 \triangleleft a \triangleright P_2$ and $Q \equiv Q_1 \triangleleft a \triangleright Q_2$, while $P_1 \not\equiv Q_1$ and for some valuation H and propositional statements R and S, $(R \triangleleft P_1 \triangleright S)/H \neq (R \triangleleft Q_1 \triangleright S)/H$, it is now required that for $a \bullet H' = H$ and $y_a(H') = T$, also $y_a(a \bullet H') = T$. However, since P and Q are cr-basic forms, P_1 and Q_1 cannot have a as a central condition, so this new requirement on H' is not relevant and $(R \triangleleft P \triangleright S)/H' \neq (R \triangleleft Q \triangleright S)/H'$.
(2) For the last case $P \equiv P_1 \triangleleft a \triangleright P_2$ and $Q \equiv Q_1 \triangleleft b \triangleright Q_2$, a valuation H is used that satisfies $y_a(H) = y_a(R \bullet (a \bullet H))$ for all propositional statements R. VAs with such a valuation exist in the variety cr.

Assume $P =_{cr} Q$, so there exist cr-basic forms P' and Q' with $\mathrm{CP}_{cr} \vdash P = P'$ and $\mathrm{CP}_{cr} \vdash Q = Q'$. By soundness, $P' =_{cr} Q'$ and as argued previously, $P' \equiv Q'$. Hence, $\mathrm{CP}_{cr} \vdash P = Q$. □

7. COMPLETENESS FOR THE VARIETY *wm*

In this section, we provide a complete axiomatization of weakly memorizing valuation congruence.

Write CP_{wm} for the axioms in CP_{cr} and these axiom schemes ($a, b \in A$):

$$\text{(CPwm1)} \qquad ((x \triangleleft a \triangleright y) \triangleleft b \triangleright z) \triangleleft a \triangleright v = (x \triangleleft b \triangleright z) \triangleleft a \triangleright v,$$
$$\text{(CPwm2)} \qquad x \triangleleft a \triangleright (y \triangleleft b \triangleright (z \triangleleft a \triangleright v)) = x \triangleleft a \triangleright (y \triangleleft b \triangleright v).$$

THEOREM 7.1. *Weakly memorizing valuation congruence $=_{wm}$ is axiomatized by CP_{wm}.*

Before proving this theorem, we define a special type of basic forms and formulate two auxiliary lemmas.

Let P be a basic form. Define $pos(P)$ as the set of atoms that occur at the central position or at left-hand (positive) positions in P: $pos(T) = pos(F) = \emptyset$ and $pos(P \triangleleft a \triangleright Q) = \{a\} \cup pos(P)$, and define $neg(P)$ as the set of atoms that occur at the central position or at right-hand (negative) positions in P: $neg(T) = neg(F) = \emptyset$ and $neg(P \triangleleft a \triangleright Q) = \{a\} \cup neg(Q)$.

Now, wm-basic forms are defined as follows:

—T and F are wm-basic forms, and
—$P \triangleleft a \triangleright Q$ is a wm-basic form if P and Q are wm-basic forms and $a \notin pos(P) \cup neg(Q)$.

The idea is that in a wm-basic form, as long as the evaluation of consecutive atoms keeps yielding the same reply, no atom is evaluated twice. Clearly, each wm-basic form also is a cr-basic form, but not vice versa, e.g., $T \triangleleft a \triangleright (T \triangleleft b \triangleright (T \triangleleft a \triangleright F))$ is not a wm-basic form because $a \in neg(T \triangleleft b \triangleright (T \triangleleft a \triangleright F))$. A more intricate example is one in which a and b "alternate":

$$(T \triangleleft b \triangleright [(F \triangleleft b \triangleright (T \triangleleft a \triangleright F)) \triangleleft a \triangleright T]) \triangleleft a \triangleright F$$

is a wm-basic form because $pos(T \triangleleft b \triangleright [(F \triangleleft b \triangleright (T \triangleleft a \triangleright F)) \triangleleft a \triangleright T]) = \{b\} \not\ni a$ and $T \triangleleft b \triangleright [(F \triangleleft b \triangleright (T \triangleleft a \triangleright F)) \triangleleft a \triangleright T]$ is a wm-basic form, where the latter statement follows because $neg((F \triangleleft b \triangleright (T \triangleleft a \triangleright F)) \triangleleft a \triangleright T) = \{a\} \not\ni b$ and because $F \triangleleft b \triangleright (T \triangleleft a \triangleright F)$ is a wm-basic form.

LEMMA 7.2. *For each propositional statement P, there is a wm-basic form P' with* $\mathrm{CP}_{wm} \vdash P = P'$.

PROOF. By Lemma 3.2, we may assume that P is a basic form and we proceed by structural induction on P. If $P \equiv T$ or $P \equiv F$, there is nothing to prove. If $P \equiv P_1 \triangleleft a \triangleright P_2$, we may assume that P_i are wm-basic forms (if not, they can proved equal to wm-basic forms). We first consider the positive side of P. If $a \notin pos(P_1)$ we are done, otherwise, we saturate P_1 by replacing each atom $b \neq a$ that occurs in a positive position with $(a \triangleleft b \triangleright F)$ using axiom CPwm1. In this way, we can retract each a that is in $pos(P_1)$ (also using axiom CPcr1) and end up with P_1' that does not contain a on positive positions. For example,

$$\begin{aligned}
&(((T \triangleleft a \triangleright R) \triangleleft b \triangleright S) \triangleleft c \triangleright V) \triangleleft a \triangleright P_2\\
&\ = (((T \triangleleft a \triangleright R) \triangleleft (a \triangleleft b \triangleright F) \triangleright S) \triangleleft (a \triangleleft c \triangleright F) \triangleright V) \triangleleft a \triangleright P_2\\
&\ = (((((T \triangleleft a \triangleright R) \triangleleft a \triangleright S) \triangleleft b \triangleright S) \triangleleft a \triangleright V) \triangleleft c \triangleright V) \triangleleft a \triangleright P_2\\
&\ = (((T \triangleleft b \triangleright S) \triangleleft a \triangleright V) \triangleleft c \triangleright V) \triangleleft a \triangleright P_2\\
&\ = ((T \triangleleft b \triangleright S) \triangleleft c \triangleright V) \triangleleft a \triangleright P_2.
\end{aligned}$$

Following the same procedure for the negative side of P (saturation with $(T \triangleleft b \triangleright a)$ for all $b \neq a$ etc.) yields a wm-basic form $P_1' \triangleleft a \triangleright P_2'$ with $\mathrm{CP}_{wm} \vdash P = P_1' \triangleleft a \triangleright P_2'$. □

Recall that $\mathbb{A}$ is a VA in the variety wm if for all $a, b \in A$,

$$\begin{aligned}
&a \bullet (a \bullet x) = a \bullet x, \qquad y_a(x) = y_a(a \bullet x),\\
&y_b(a \bullet x) = y_a(x) \rightarrow (a \bullet (b \bullet (a \bullet x)) = b \bullet (a \bullet x) \ \wedge \ y_a(b \bullet (a \bullet x)) = y_a(x)).
\end{aligned}$$

LEMMA 7.3. *For all wm-basic forms P and Q, $P =_{wm} Q$ implies $P \equiv Q$.*

PROOF. The implication of the lemma follows by contraposition and nested induction on the complexity of wm-basic forms.

If $P \equiv T$, then if $Q \equiv F$, $P \neq_{wm} Q$ is witnessed by each VA in which $T \neq F$. If $Q \equiv Q_1 \triangleleft a \triangleright Q_2$, then consider $\mathbb{A} \in wm$ with $T \neq F$ and with a valuation H that satisfies $y_a(H) = y_b(H) = T$ and $y_b(R \bullet (a \bullet H)) = F$ for all propositional statements R. Then, $(P \circ b)/H = T$ while $(Q \circ b)/H = F$, so $P \neq_{wm} Q$.

If $P \equiv F$, a similar argument applies.

If $P \equiv P_1 \triangleleft a \triangleright P_2$, then the cases $Q \equiv T$ and $Q \equiv F$ can be dealt with as previously shown. If $Q \equiv Q_1 \triangleleft a \triangleright Q_2$, then assume $P \not\equiv Q$ because $P_1 \not\equiv Q_1$. By induction, $P_1 \neq_{wm} Q_1$. By implication (9), there exists $\mathbb{A} \in wm$ with valuation H and propositional statements R and S such that $(R \triangleleft P_1 \triangleright S)/H \neq (R \triangleleft Q_1 \triangleright S)/H$. Observe that a does not occur in $pos(P_1)$ and $pos(Q_1)$, and that we may assume that if a occurs in R and/or S, its substitution by T preserves this inequality (otherwise, replace a by $F \triangleleft a \triangleright T$). Now extend $\mathbb{A}$ to $\mathbb{A}'$ with a valuation H' with $y_a(H') = T$ and $a \bullet H' = H$, then $R \triangleleft P \triangleright S$ and $R \triangleleft Q \triangleright S$ yield different results under H', hence $P \neq_{wm} Q$. It is clear that also $\mathbb{A}' \in wm$. If $P_1 \equiv Q_1$, then $P_2 \not\equiv Q_2$ and a similar argument applies.

Finally, if $Q \equiv Q_1 \triangleleft b \triangleright Q_2$ with $b \neq a$, then $P \neq_{wm} Q$ follows by considering $\mathbb{A}$ with $T \neq F$ and with valuation H that satisfies for all R, $y_a(H) = y_a(R \bullet (a \bullet H)) = T$ and $y_b(H) = y_a(R \bullet (b \bullet H)) = F$, so $(P \circ a)/H = T$ while $(Q \circ a)/H = F$. Clearly, $\mathbb{A} \in wm$. □

PROOF OF THEOREM 7.1. Let $\mathbb{A}$ be a VA in the variety wm. The soundness of axiom CPwm1 and CPwm2 follows immediately and we only show this for the latter one: let

$H \in \mathbb{A}$, then

$$
\begin{aligned}
&(P \triangleleft a \triangleright (Q \triangleleft b \triangleright (Z \triangleleft a \triangleright V)))/H \\
&\quad = \begin{cases} P/(a \bullet H) & \text{if } y_a(H) = T, \\ Q/(b \bullet (a \bullet H)) & \text{if } y_a(H) = F \text{ and } y_b(a \bullet H) = T, \\ (Z \triangleleft a \triangleright V)/(b \bullet (a \bullet H)) & \text{if } y_a(H) = F \text{ and } y_b(a \bullet H) = F, \end{cases} \\
&\quad = \begin{cases} P/(a \bullet H) & \text{if } y_a(H) = T, \\ Q/(b \bullet (a \bullet H)) & \text{if } y_a(H) = F \text{ and } y_b(a \bullet H) = T, \\ V/(b \bullet (a \bullet H)) & \text{if } y_a(H) = F \text{ and } y_b(a \bullet H) = F, \end{cases} \\
&\quad = (P \triangleleft a \triangleright (Q \triangleleft b \triangleright V))/H,
\end{aligned}
$$

and

$$
\begin{aligned}
&(P \triangleleft a \triangleright (Q \triangleleft b \triangleright (Z \triangleleft a \triangleright V))) \bullet H \\
&\quad = \begin{cases} P \bullet (a \bullet H) & \text{if } y_a(H) = T, \\ Q \bullet (b \bullet (a \bullet H)) & \text{if } y_a(H) = F \text{ and } y_b(a \bullet H) = T, \\ (Z \triangleleft a \triangleright V) \bullet (b \bullet (a \bullet H)) & \text{if } y_a(H) = F \text{ and } y_b(a \bullet H) = F, \end{cases} \\
&\quad = \begin{cases} P \bullet (a \bullet H) & \text{if } y_a(H) = T, \\ Q \bullet (b \bullet (a \bullet H)) & \text{if } y_a(H) = F \text{ and } y_b(a \bullet H) = T, \\ V \bullet (b \bullet (a \bullet H)) & \text{if } y_a(H) = F \text{ and } y_b(a \bullet H) = F, \end{cases} \\
&\quad = (P \triangleleft a \triangleright (Q \triangleleft b \triangleright V)) \bullet H.
\end{aligned}
$$

In order to prove completeness, assume $P =_{wm} Q$. By Lemma 7.2, there are wm-basic forms P' and Q' with $\mathrm{CP}_{wm} \vdash P = P'$ and $\mathrm{CP}_{wm} \vdash Q = Q'$. By soundness $P' =_{wm} Q'$, thus by Lemma 7.3, $P' \equiv Q'$, and thus $\mathrm{CP}_{wm} \vdash P = Q$. $\square$

8. COMPLETENESS FOR THE VARIETY *mem*

In this section, we provide a complete axiomatization of memorizing valuation congruence.

Write CP_{mem} for the axioms in CP and this axiom:

$$
\text{(CPmem)} \qquad x \triangleleft y \triangleright (z \triangleleft u \triangleright (v \triangleleft y \triangleright w)) = x \triangleleft y \triangleright (z \triangleleft u \triangleright w).
$$

THEOREM 8.1. *Memorizing valuation congruence $=_{mem}$ is axiomatized by* CP_{mem}.

Before proving this theorem, we discuss some characteristics of CP_{mem}. Axiom CPmem defines how the central condition y may recur in an expression. This axiom yields in combination with CP some interesting consequences. First, CPmem has three symmetric variants, which all follow easily with $x \triangleleft y \triangleright z = z \triangleleft (F \triangleleft y \triangleright T) \triangleright x$ $(= z \triangleleft \neg y \triangleright x)$:

$$
x \triangleleft y \triangleright ((z \triangleleft y \triangleright u) \triangleleft v \triangleright w) = x \triangleleft y \triangleright (u \triangleleft v \triangleright w), \tag{10}
$$

$$
(x \triangleleft y \triangleright (z \triangleleft u \triangleright v)) \triangleleft u \triangleright w = (x \triangleleft y \triangleright z) \triangleleft u \triangleright w, \tag{11}
$$

$$
((x \triangleleft y \triangleright z) \triangleleft u \triangleright v) \triangleleft y \triangleright w = (x \triangleleft u \triangleright v) \triangleleft y \triangleright w. \tag{12}
$$

The axioms of CP_{mem} imply various laws for contraction:

$$
x \triangleleft y \triangleright (v \triangleleft y \triangleright w) = x \triangleleft y \triangleright w \qquad \text{(take } u = F \text{ in CPmem)}, \tag{13}
$$

$$
x \triangleleft y \triangleright (T \triangleleft u \triangleright y) = x \triangleleft y \triangleright u \qquad \text{(take } z = v = T \text{ and } w = F \text{ in CPmem)}, \tag{14}
$$

$$
(x \triangleleft y \triangleright z) \triangleleft y \triangleright u = x \triangleleft y \triangleright u, \tag{15}
$$

$$
(x \triangleleft y \triangleright F) \triangleleft x \triangleright z = y \triangleleft x \triangleright z, \tag{16}
$$

and thus (take $v = T$ and $w = F$ in (13), respectively $x = T$ and $z = F$ in (15)),

$$x \triangleleft y \triangleright y = x \triangleleft y \triangleright F \quad \text{and} \quad y \triangleleft y \triangleright u = T \triangleleft y \triangleright u.$$

The latter two equations immediately imply the following very simple contraction laws:

$$x \triangleleft x \triangleright x = x \triangleleft x \triangleright F = T \triangleleft x \triangleright x = T \triangleleft x \triangleright F = x.$$

Let A' be a subset of A. We employ a special type of basic forms based on A': *mem*-basic forms over A' are defined by

—T and F are *mem*-basic forms over A', and
—$P \triangleleft a \triangleright Q$ is a *mem*-basic form over A' if $a \in A'$ and P and Q are *mem*-basic forms over $A' \setminus \{a\}$.

For example, for $A' = \{a\}$ the set of all *mem*-basic forms is $\{bv,\ bv \triangleleft a \triangleright bv' \mid bv, bv' \in \{T, F\}\}$, and for $A' = \{a, b\}$ it is

$$\{bv, t_1 \triangleleft a \triangleright t_2, t_3 \triangleleft b \triangleright t_4 \mid bv \in \{T, F\},\ t_1, t_2\ \textit{mem}\text{-basic forms over } \{b\},\\ t_3, t_4\ \textit{mem}\text{-basic forms over } \{a\}\}.$$

LEMMA 8.2. *For each propositional statement P, there is a mem-basic form P' with* $\mathrm{CP}_{mem} \vdash P = P'$.

PROOF. By Lemma 3.2, we may assume that P is a basic form and we proceed by structural induction on P. If $P \equiv T$ or $P \equiv F$, there is nothing to prove.

Assume $P \equiv P_1 \triangleleft a \triangleright P_2$. We write $[T/a]P_1$ for the term that results when T is substituted for a in P_1. We first show that

$$\mathrm{CP}_{mem} \vdash P_1 \triangleleft a \triangleright P_2 = [T/a]P_1 \triangleleft a \triangleright P_2$$

by induction on P_1: if P_1 equals T or F, this is clear. If $P_1 \equiv Q \triangleleft a \triangleright R$, then $\mathrm{CP} \vdash [T/a]P_1 = [T/a]Q$ and we derive

$$\begin{aligned} P_1 \triangleleft a \triangleright P_2 &= (Q \triangleleft a \triangleright R) \triangleleft a \triangleright P_2 \\ &\stackrel{IH}{=} ([T/a]Q \triangleleft a \triangleright R) \triangleleft a \triangleright P_2 \\ &\stackrel{(15)}{=} [T/a]Q \triangleleft a \triangleright P_2 \\ &= [T/a]P_1 \triangleleft a \triangleright P_2, \end{aligned}$$

and if $P_1 \equiv Q \triangleleft b \triangleright R$ with $b \neq a$, then $\mathrm{CP} \vdash [T/a]P_1 = [T/a]Q \triangleleft b \triangleright [T/a]R$ and we derive

$$\begin{aligned} P_1 \triangleleft a \triangleright P_2 &= (Q \triangleleft b \triangleright R) \triangleleft a \triangleright P_2 \\ &\stackrel{(11)(12)}{=} ((Q \triangleleft a \triangleright T) \triangleleft b \triangleright (R \triangleleft a \triangleright T)) \triangleleft a \triangleright P_2 \\ &\stackrel{IH}{=} (([T/a]Q \triangleleft a \triangleright T) \triangleleft b \triangleright ([T/a]R \triangleleft a \triangleright T)) \triangleleft a \triangleright P_2 \\ &\stackrel{(11)(12)}{=} ([T/a]Q \triangleleft b \triangleright [T/a]R) \triangleleft a \triangleright P_2 \\ &= [T/a]P_1 \triangleleft a \triangleright P_2. \end{aligned}$$

In a similar way, but now using (13), axiom CPmem and (10) instead, we find $\mathrm{CP}_{mem} \vdash P_1 \triangleleft a \triangleright P_2 = P_1 \triangleleft a \triangleright [F/a]P_2$, and thus

$$\mathrm{CP}_{mem} \vdash P_1 \triangleleft a \triangleright P_2 = [T/a]P_1 \triangleleft a \triangleright [F/a]P_2.$$

With axioms CP1 and CP2, we find basic forms Q_i in which a does not occur with $\mathrm{CP}_{mem} \vdash Q_1 = [T/a]P_1$ and $\mathrm{CP}_{mem} \vdash Q_2 = [F/a]P_2$.

By induction, it follows that there are *mem*-basic forms R_1 and R_2 with $\mathrm{CP}_{mem} \vdash R_i = Q_i$, and hence $\mathrm{CP}_{mem} \vdash P = R_1 \triangleleft a \triangleright R_2$ and $R_1 \triangleleft a \triangleright R_2$ is a *mem*-basic form. □

Next we formulate two more auxiliary lemmas. Recall that $\mathbb{A}$ is a VA in the variety mem if for all $a, b \in A$,

$$a \bullet (a \bullet x) = a \bullet x, \qquad y_a(x) = y_a(a \bullet x),$$
$$a \bullet (b \bullet (a \bullet x)) = b \bullet (a \bullet x), \qquad y_a(b \bullet (a \bullet x)) = y_a(x).$$

LEMMA 8.3. *For each valuation algebra $\mathbb{A} \in mem$, and for all propositional statements P, Q and valuations $H \in \mathbb{A}$,*

$$Q \bullet (P \bullet (Q \bullet H)) = P \bullet (Q \bullet H) \quad and \quad Q/(P \bullet (Q \bullet H)) = Q/H. \tag{17}$$

PROOF. By structural induction on Q. The cases $Q \equiv T$ and $Q \equiv F$ are trivial. If $Q \equiv a$, then apply structural induction on P: the cases $P \equiv T$ and $P \equiv F$ are trivial, and if $P \in A$, then (17) follows by definition of mem. If $P \equiv P_1 \triangleleft P_2 \triangleright P_3$, then if $P_2/(a \bullet H) = T$,

$$\begin{aligned} a \bullet ((P_1 \triangleleft P_2 \triangleright P_3) \bullet (a \bullet H)) &= a \bullet (P_1 \bullet (P_2 \bullet (a \bullet H))) \\ &= a \bullet (P_1 \bullet (a \bullet (P_2 \bullet (a \bullet H)))) \\ &= P_1 \bullet (a \bullet (P_2 \bullet (a \bullet H))) \\ &= P_1 \bullet (P_2 \bullet (a \bullet H)) \\ &= (P_1 \triangleleft P_2 \triangleright P_3) \bullet (a \bullet H), \end{aligned}$$

and

$$\begin{aligned} a/((P_1 \triangleleft P_2 \triangleright P_3) \bullet (a \bullet H)) &= a/(P_1 \bullet (P_2 \bullet (a \bullet H))) \\ &= a/(P_1 \bullet (a \bullet (P_2 \bullet (a \bullet H)))) \\ &= a/(P_2 \bullet (a \bullet H)) \\ &= a/H, \end{aligned}$$

and if $P_2/(a \bullet H) = F$ a similar argument applies.

If $Q \equiv Q_1 \triangleleft Q_2 \triangleright Q_3$, then first assume $Q_2/H = T$, so $Q \bullet H = Q_1 \bullet (Q_2 \bullet H)$. Observe that

$$(Q_1 \circ P) \bullet (Q_2 \bullet H) = P \bullet (Q_1 \bullet (Q_2 \bullet H)),$$

so by induction $Q_2/(P \bullet (Q_1 \bullet (Q_2 \bullet H))) = Q_2/((Q_1 \circ P) \bullet (Q_2 \bullet H)) = Q_2/H = T$. Using induction we further derive

$$\begin{aligned} Q \bullet (P \bullet (Q \bullet H)) &= Q_1 \bullet ((P \circ Q_2) \bullet (Q_1 \bullet (Q_2 \bullet H))) \\ &= (P \circ Q_2) \bullet (Q_1 \bullet (Q_2 \bullet H)) \\ &= Q_2 \bullet ((Q_1 \circ P) \bullet (Q_2 \bullet H)) \\ &= P \bullet (Q_1 \bullet (Q_2 \bullet H)) \\ &= P \bullet (Q \bullet H), \end{aligned}$$

and

$$\begin{aligned} (Q_1 \triangleleft Q_2 \triangleright Q_3)/(P \bullet (Q \bullet H)) &= Q_1/((P \circ Q_2) \bullet (Q_1 \bullet (Q_2 \bullet H)) \\ &= Q_1/(Q_2 \bullet H) \\ &= Q/H. \end{aligned}$$

Finally, if $Q_2/H = F$ a similar argument applies. □

LEMMA 8.4. *For all mem-basic forms P and Q, $P =_{mem} Q$ implies $P \equiv Q$.*

PROOF. The implication of the lemma follows by contraposition and nested induction on the complexity of mem-basic forms.

If $P \equiv T$, then if $Q \equiv F$, $P \neq_{mem} Q$ is witnessed by each VA in which $T \neq F$. If $Q \equiv Q_1 \lhd a \rhd Q_2$, then consider $\mathbb{A} \in mem$ with $T \neq F$ and with a valuation H that satisfies $y_a(H) = y_b(H) = T$ and $y_b(R \bullet (a \bullet H)) = F$ for all propositional statements R. Then, $(P \circ b)/H = T$ while $(Q \circ b)/H = F$, so $P \neq_{mem} Q$.

If $P \equiv F$ a similar argument applies.

If $P \equiv P_1 \lhd a \rhd P_2$, then the cases $Q \equiv T$ and $Q \equiv F$ can be dealt with as above. If $Q \equiv Q_1 \lhd a \rhd Q_2$, then assume $P \not\equiv Q$ because $P_1 \not\equiv Q_1$. By induction, $P_1 \neq_{mem} Q_1$. By implication (9), there exists $\mathbb{A} \in mem$ with valuation H and propositional statements R and S such that $(R \lhd P_1 \rhd S)/H \neq (R \lhd Q_1 \rhd S)/H$. Observe that a does not occur in P_1 and Q_1, and that we may assume that if a occurs in R and/or S, its substitution by T preserves this inequality (otherwise, replace a by $F \lhd a \rhd T$). Now extend $\mathbb{A}$ to $\mathbb{A}'$ with a valuation H' with $y_a(H') = T$ and $a \bullet H' = H$, then $R \lhd P \rhd S$ and $R \lhd Q \rhd S$ yield different results under H'; hence, $P \neq_{mem} Q$. Clearly, also $\mathbb{A}' \in mem$. If $P_1 \equiv Q_1$, then $P_2 \not\equiv Q_2$ and a similar argument applies.

Finally, if $Q \equiv Q_1 \lhd b \rhd Q_2$ with $b \neq a$, then $P \neq_{mem} Q$ follows by considering $\mathbb{A} \in mem$ with $T \neq F$ and with valuation H that satisfies $y_a(H) = T$, $y_b(H) = F$, and for all propositional statements R, $y_a(R \bullet (b \bullet H)) = F$. Then, by Lemma 8.3, $y_a(P \bullet H) = y_a(P_1 \bullet (a \bullet H)) = y_a(H)$, so $(P \circ a)/H = T$ while $(Q \circ a)/H = F$. □

PROOF OF THEOREM 8.1. Let $\mathbb{A}$ be a VA in the variety mem. In order to prove soundness, we have to show that axiom CPmem holds in $\mathbb{A}$. Consider propositional statement

$$P \lhd Q \rhd (R \lhd S \rhd (V \lhd Q \rhd W)).$$

By Lemma 8.3, we find

$$Q/H = F \Longrightarrow \begin{cases} (V \lhd Q \rhd W)/(S \bullet (Q \bullet H)) = W/(S \bullet (Q \bullet H)), \\ (V \lhd Q \rhd W) \bullet (S \bullet (Q \bullet H)) = W \bullet (S \bullet (Q \bullet H)). \end{cases}$$

Now the soundness of axiom CPmem follows immediately:

$$\begin{aligned}
&(P \lhd Q \rhd (R \lhd S \rhd (V \lhd Q \rhd W)))/H \\
&\quad = \begin{cases} P/(Q \bullet H) & \text{if } Q/H = T, \\ R/(S \bullet (Q \bullet H)) & \text{if } Q/H = F \text{ and } S/(Q \bullet H) = T, \\ (V \lhd Q \rhd W)/(S \bullet (Q \bullet H)) & \text{if } Q/H = F \text{ and } S/(Q \bullet H) = F, \end{cases} \\
&\quad = \begin{cases} P/(Q \bullet H) & \text{if } Q/H = T, \\ R/(S \bullet (Q \bullet H)) & \text{if } Q/H = F \text{ and } S/(Q \bullet H) = T, \\ W/(S \bullet (Q \bullet H)) & \text{if } Q/H = F \text{ and } S/(Q \bullet H) = F, \end{cases} \\
&\quad = (P \lhd Q \rhd (R \lhd S \rhd W))/H,
\end{aligned}$$

and

$$\begin{aligned}
&(P \lhd Q \rhd (R \lhd S \rhd (V \lhd Q \rhd W))) \bullet H \\
&\quad = \begin{cases} P \bullet (Q \bullet H) & \text{if } Q/H = T, \\ R \bullet (S \bullet (Q \bullet H)) & \text{if } Q/H = F \text{ and } S/(Q \bullet H) = T, \\ (V \lhd Q \rhd W) \bullet (S \bullet (Q \bullet H)) & \text{if } Q/H = F \text{ and } S/(Q \bullet H) = F, \end{cases} \\
&\quad = \begin{cases} P \bullet (Q \bullet H) & \text{if } Q/H = T, \\ R \bullet (S \bullet (Q \bullet H)) & \text{if } Q/H = F \text{ and } S/(Q \bullet H) = T, \\ W \bullet (S \bullet (Q \bullet H)) & \text{if } Q/H = F \text{ and } S/(Q \bullet H) = F, \end{cases} \\
&\quad = (P \lhd Q \rhd (R \lhd S \rhd W)) \bullet H.
\end{aligned}$$

In order to prove completeness, assume $P =_{mem} Q$. By Lemma 8.2, there are mem-basic forms P' and Q' with $\mathrm{CP}_{mem} \vdash P = P'$ and $\mathrm{CP}_{mem} \vdash Q = Q'$. By soundness, $P' =_{mem} Q'$, thus, by Lemma 8.4, $P' \equiv Q'$, and thus $\mathrm{CP}_{mem} \vdash P = Q$. □

9. COMPLETENESS FOR THE VARIETY *st*

In this section, we provide a complete axiomatization of static valuation congruence.

THEOREM 9.1 ([HOARE 1985B]). *Static valuation congruence $=_{st}$ is axiomatized by the axioms in* CP *(see Table I) and these axioms:*

$$\text{(CPstat)} \qquad (x \triangleleft y \triangleright z) \triangleleft u \triangleright v = (x \triangleleft u \triangleright v) \triangleleft y \triangleright (z \triangleleft u \triangleright v),$$
$$\text{(CPcontr)} \qquad (x \triangleleft y \triangleright z) \triangleleft y \triangleright u = x \triangleleft y \triangleright u.$$

We write CP_{st} *for this set of axioms.*

Observe that axiom CPcontr equals the derivable identity (15), which holds in CP_{mem}. Also note that the symmetric variants of the axioms CPstat and CPcontr, say

$$\text{(CPstat}'\text{)} \qquad x \triangleleft y \triangleright (z \triangleleft u \triangleright v) = (x \triangleleft y \triangleright z) \triangleleft u \triangleright (x \triangleleft y \triangleright v),$$
$$\text{(CPcontr}'\text{)} \qquad x \triangleleft y \triangleright (z \triangleleft y \triangleright u) = x \triangleleft y \triangleright u,$$

easily follow with identity (2), that is, $y \triangleleft x \triangleright z = z \triangleleft (F \triangleleft x \triangleright T) \triangleright y$, which is even valid in free valuation congruence, and that $\text{CPcontr}' = (13)$. Thus, the axiomatization of static valuation congruence is obtained from CP by adding the axiom CPstat that prescribes for a nested conditional composition how the order of the first and a second central condition can be changed, and a generalization of the axioms CPcr1 and CPcr2 that prescribes contraction for terms (instead of atoms). Moreover, in CP_{st} it can be derived that

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

thus any "and-then" prefix can be added to (or left out from) a propositional statement while preserving static valuation congruence, in particular $x \triangleleft x \triangleright x = x \circ x = x$.

PROOF OF THEOREM 9.1. Soundness follows from the definition of static valuations: let $\mathbb{A}$ be a VA that satisfies for all $a, b \in A$,

$$y_a(b \bullet x) = y_a(x).$$

These equations imply that for all P, Q and $H \in \mathbb{A}$,

$$P/H = P/(Q \bullet H).$$

As a consequence, the validity of axioms CPstat and CPcontr follows from simple case distinctions. Furthermore, Hoare showed in [1985b] that CP_{st} is complete for static valuation congruence.

For an idea of a direct proof, assume $P =_{st} Q$ and assume that the atoms occurring in P and Q are ordered as $a_1, \dots, a_n$. Then, under static valuation congruence, each propositional statement containing no other atoms than $a_1, \dots, a_n$ can be rewritten into the following special type of basic form: consider the full binary tree with at level i only occurrences of atom a_i (there are 2^{i-1} such occurrences), and at level $n+1$ only leaves that are either T or F (there are 2^n such leaves). Then, each series of leaves represents one of the possible propositional statements in which these atoms may occur, and the axioms in CP_{st} are sufficient to rewrite both P and Q into exactly one such basic

Table II. Some Immediate Consequences of the Set of Axioms CP and Eq. (18)

$\neg T = F$	(19)
$\neg F = T$	(20)
$\neg\neg x = x$	(21)
$\neg(x \triangleleft y \triangleright z) = \neg x \triangleleft y \triangleright \neg z$	(22)
$x \triangleleft \neg y \triangleright z = z \triangleleft y \triangleright x$	(23)

Table III. Defining Equations for Derived Connectives

$x \mathbin{{}_{\circ}\!\wedge} y = y \triangleleft x \triangleright F$	$x \mathbin{\circ\!\!\rightarrow} y = y \triangleleft x \triangleright T$
$x \mathbin{\wedge_{\circ}} y = x \triangleleft y \triangleright F$	$x \mathbin{\rightarrow\!\!\circ} y = T \triangleleft y \triangleright \neg x$
$x \mathbin{{}^{\circ}\!\vee} y = T \triangleleft x \triangleright y$	$x \mathbin{\circ\!\!\leftrightarrow} y = y \triangleleft x \triangleright \neg y$
$x \mathbin{\vee^{\circ}} y = T \triangleleft y \triangleright x$	$x \mathbin{\leftrightarrow\!\!\circ} y = x \triangleleft y \triangleright \neg x$

form. For these basic forms, static valuation congruence implies syntactic equivalence. Hence, completeness follows. □

As an aside, we note that the axioms CPcontr and CPcontr$'$ immediately imply CPcr1 and CPcr2, and conversely, that each instance of these axioms is derivable from CP+CPstat+CPcr1+CPcr2 (by induction on basic forms on y's position), which proves completeness of this particular group of axioms.

10. ADDING NEGATION AND DEFINABLE CONNECTIVES

In this section, we formally add negation and various definable connectives to CP. As stated earlier (see identity (2)), negation $\neg x$ can be defined as follows:

$$\neg x = F \triangleleft x \triangleright T. \tag{18}$$

The derivable identities in Table II play a role in the derivable connectives that we discuss shortly. They can be derived as follows:

(19) follows from $\neg T = F \triangleleft T \triangleright T = F$,
(20) follows in a similar way,
(21) follows from $\neg\neg x = F \triangleleft (F \triangleleft x \triangleright T) \triangleright T = (F \triangleleft F \triangleright T) \triangleleft x \triangleright (F \triangleleft T \triangleright T) = T \triangleleft x \triangleright F = x$,
(22) follows in a similar way,
(23) follows from $x \triangleleft \neg y \triangleright z = (z \triangleleft F \triangleright x) \triangleleft \neg y \triangleright (z \triangleleft T \triangleright x) = z \triangleleft (F \triangleleft \neg y \triangleright T) \triangleright x = z \triangleleft y \triangleright x$.

A definable (binary) connective already introduced is the *and then* operator $\circ$ with defining equation $x \circ y = y \triangleleft x \triangleright y$. Furthermore, following Bergstra et al. [1995], we write

$$\mathbin{{}_{\circ}\!\wedge}$$

for *left-sequential* conjunction, that is, a conjunction that first evaluates its left argument and only after that is found T carries on with evaluating its second argument (the small circle indicates which argument is evaluated first). Similar notations are used for other sequential connectives. We provide defining equations for a number of derived connectives in Table III.

The operators $\mathbin{{}_{\circ}\!\wedge}$ and left-sequential disjunction $\mathbin{{}^{\circ}\!\vee}$ are associative and the dual of each other, and so are their right-sequential counterparts. For $\mathbin{{}_{\circ}\!\wedge}$, a proof of this is as follows:

$$\begin{aligned}(x \mathbin{{}_{\circ}\!\wedge} y) \mathbin{{}_{\circ}\!\wedge} z &= z \triangleleft (y \triangleleft x \triangleright F) \triangleright F\\ &= (z \triangleleft y \triangleright F) \triangleleft x \triangleright (z \triangleleft F \triangleright F)\\ &= (y \mathbin{{}_{\circ}\!\wedge} z) \triangleleft x \triangleright F\\ &= x \mathbin{{}_{\circ}\!\wedge} (y \mathbin{{}_{\circ}\!\wedge} z),\end{aligned}$$

and (a sequential version of De Morgan's laws)

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

Furthermore, note that $T \mathbin{{}_{\circ}\!\wedge} x = x$ and $x \mathbin{{}_{\circ}\!\wedge} T = x$, and $F \mathbin{{}^{\circ}\!\vee} x = x$ and $x \mathbin{{}^{\circ}\!\vee} F = x$.

Of course, *distributivity*, as in $(x \mathbin{{}_{\circ}\!\wedge} y) \mathbin{{}^{\circ}\!\vee} z = (x \mathbin{{}^{\circ}\!\vee} z) \mathbin{{}_{\circ}\!\wedge} (y \mathbin{{}^{\circ}\!\vee} z)$ is not valid in free valuation congruence: it changes the order of evaluation and in the right-hand expression z can be evaluated twice. It is also obvious that both sequential versions of *absorption*, one of which reads

$$x = x \mathbin{\wedge_{\circ}} (x \mathbin{\vee^{\circ}} y),$$

are not valid. Furthermore, it is not difficult to prove in CP that $\mathbin{\circ\!\!\leftrightarrow}$ and $\mathbin{\leftrightarrow\!\!\circ}$ (i.e., the two sequential versions of bi-implication defined in Table III) are also associative, and that $\mathbin{\circ\!\!\rightarrow}$ and $\mathbin{\rightarrow\!\!\circ}$ are not associative, but satisfy the sequential versions of the common definition of implication:

$$x \mathbin{\circ\!\!\rightarrow} y = \neg x \mathbin{{}^{\circ}\!\vee} y \quad \text{and} \quad x \mathbin{\rightarrow\!\!\circ} y = \neg x \mathbin{\vee^{\circ}} y.$$

From now on, we extend $\Sigma_{\mathrm{CP}}(A)$ with the "and then" operator $\circ$, negation and all derived connectives introduced in this section, and we adopt their defining equations. Of course, it remains the case that each propositional statement has a unique basic form (cf. Lemma 3.2).

Concerning the example of the propositional statement sketched in Example (1) in Section 2:

$$\textit{look-left-and-check} \mathbin{{}_{\circ}\!\wedge} \textit{look-right-and-check} \mathbin{{}_{\circ}\!\wedge} \textit{look-left-and-check}$$

indeed precisely models part of the processing of a pedestrian planning to cross a road with two-way traffic driving on the right.

We end this section with a brief comment on these connectives in the setting of other valuation congruences. In memorizing valuation congruence, the sequential connective $\mathbin{{}_{\circ}\!\wedge}$ has the following properties:

(1) The associativity of $\mathbin{{}_{\circ}\!\wedge}$ is valid,
(2) The identity $x \mathbin{{}_{\circ}\!\wedge} y \mathbin{{}_{\circ}\!\wedge} x = x \mathbin{{}_{\circ}\!\wedge} y$ is valid (by Eq. (16)),
(3) The connective $\mathbin{{}_{\circ}\!\wedge}$ is not commutative.

In static valuation congruence, all of $\mathbin{{}_{\circ}\!\wedge}$, $\mathbin{{}^{\circ}\!\vee}$, $\mathbin{\wedge_{\circ}}$ and $\mathbin{\vee^{\circ}}$ are commutative and idempotent. For example, we derive with axiom CPstat that

$$x \mathbin{{}_{\circ}\!\wedge} y = y \triangleleft x \triangleright F = (T \triangleleft y \triangleright F) \triangleleft x \triangleright F = (T \triangleleft x \triangleright F) \triangleleft y \triangleright (F \triangleleft x \triangleright F) = x \triangleleft y \triangleright F = y \mathbin{{}_{\circ}\!\wedge} x,$$

and with axiom CPcontr and its symmetric counterpart CPcontr$'$,

$$x \mathbin{{}_{\circ}\!\wedge} x = x \triangleleft x \triangleright F = (T \triangleleft x \triangleright F) \triangleleft x \triangleright F = T \triangleleft x \triangleright F = x,$$
$$x \mathbin{{}^{\circ}\!\vee} x = T \triangleleft x \triangleright x = T \triangleleft x \triangleright (T \triangleleft x \triangleright F) = T \triangleleft x \triangleright F = x.$$

As a consequence, the sequential notation of these connectives is not meaningful in this case, and distributivity and absorption hold in static valuation congruence.

11. SATISFIABILITY AND COMPLEXITY

In this section, we briefly consider some complexity issues. Given a variety K of valuation algebras, a propositional statement is *satisfiable* with respect to K if for some nontrivial $\mathbb{A} \in K$ (T_{Val} and F_{Val} are different), there exists a valuation $H \in \mathbb{A}$ such that

$$P/H = T.$$

We write

$$\mathbf{SAT}_K(P)$$

if P is satisfiable. We say that P is *falsifiable* with respect to K, notation

$$\mathbf{FAL}_K(P),$$

if and only if a valuation $H \in \mathbb{A} \in K$ exists with $P/H = F$. This is the case if and only if $\mathbf{SAT}_K(F \triangleleft P \triangleright T)$.

It is a well-known fact that $\mathbf{SAT}_{st}$ is an NP-complete problem, and it is easily seen that for all propositional statements P, $\mathbf{SAT}_{mem}(P) = \mathbf{SAT}_{st}(P)$. We now argue that $\mathbf{SAT}_{fr}$ is in P. This is the case because in the variety fr, both $\mathbf{SAT}_{fr}$ and $\mathbf{FAL}_{fr}$ can be simultaneously defined in an inductive manner: let $a \in A$ and write $\neg\mathbf{SAT}_{fr}(P)$ to express that $\mathbf{SAT}_{fr}(P)$ does not hold, and similar for $\mathbf{FAL}_{fr}$, then

$$\begin{array}{ll} \mathbf{SAT}_{fr}(T), & \neg\mathbf{FAL}_{fr}(T), \\ \neg\mathbf{SAT}_{fr}(F), & \mathbf{FAL}_{fr}(F), \\ \mathbf{SAT}_{fr}(a), & \mathbf{FAL}_{fr}(a), \end{array}$$

and

$$\mathbf{SAT}_{fr}(P \triangleleft Q \triangleright R) \quad \text{if} \quad \begin{cases} \mathbf{SAT}_{fr}(Q) \text{ and } \mathbf{SAT}_{fr}(P), \\ \text{or} \\ \mathbf{FAL}_{fr}(Q) \text{ and } \mathbf{SAT}_{fr}(R), \end{cases}$$

$$\mathbf{FAL}_{fr}(P \triangleleft Q \triangleright R) \quad \text{if} \quad \begin{cases} \mathbf{SAT}_{fr}(Q) \text{ and } \mathbf{FAL}_{fr}(P), \\ \text{or} \\ \mathbf{FAL}_{fr}(Q) \text{ and } \mathbf{FAL}_{fr}(R). \end{cases}$$

Hence, with respect to free valuation congruence both $\mathbf{SAT}_{fr}(P)$ and $\mathbf{FAL}_{fr}(P)$ are computable in polynomial time. In a similar way, one can show that both $\mathbf{SAT}_{rp}$ and $\mathbf{SAT}_{cr}$ are in P.

Of course, many more models of CP exist than those discussed in the previous sections. For example, call a valuation *positively memorizing* ($Pmem$) if the reply T to an atom is preserved after all subsequent replies:

$$x \triangleleft a \triangleright y = [T/a]x \triangleleft a \triangleright y$$

for all atomic propositions a. In a similar way, one can define negatively memorizing valuations ($Nmem$):

$$x \triangleleft a \triangleright y = x \triangleleft a \triangleright [F/a]y.$$

Contractive or weakly memorizing valuations that satisfy $Pmem$ ($Nmem$) give rise to new models in which more propositional statements are identified.

THEOREM 11.1. *For*

$$K \in \{Pmem,\ cr{+}Pmem,\ wm{+}Pmem,\ Nmem,\ cr{+}Nmem,\ wm{+}Nmem,\ mem\}$$

it holds that $\mathbf{SAT}_K$ *is NP-complete.*

PROOF. We only consider the case for $Pmem$. Then

$$\mathbf{SAT}_{st}(P) = \mathbf{SAT}_{mem}(P) = \mathbf{SAT}_{Pmem}(P \mathbin{_{\circ}\!\!\!\wedge} \cdots \mathbin{_{\circ}\!\!\!\wedge} P),$$

where P is repeated $n+1$ times with n the number of atoms occurring in P. Each time a P evaluates to T while it would not do so in mem, this is due to some atom that changes the reply. So, this must be a change from F to T, because T remains T in $Pmem$. Per atom this can happen at most once, and if each P yields T, then at least once without any atom taking different values. But then P is also satisfiable in mem. Thus, the NP-complete problem $\mathbf{SAT}_{st}(P)$ can be polynomially reduced to $\mathbf{SAT}_{Pmem}(P)$, hence $\mathbf{SAT}_{Pmem}$ is NP-complete. □

For $K \in \{cr{+}Pmem,\ wm{+}Pmem,\ cr{+}Nmem,\ wm{+}Nmem\}$, each closed term can be written with T, F, $\neg$, $\mathbin{_{\circ}\!\!\!\wedge}$ and $\mathbin{^{\circ}\!\!\!\vee}$ only. For example in $cr{+}Pmem$:

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

because after a positive reply to a and whatever happens in x, the next a is again positive, so y is not evaluated, and after a negative reply to a, the subsequent a gets a negative reply because of cr, so then y is tested. So here we see models that identify less than $=_{mem}$ and in which each closed term can be written without conditional composition. At first sight, this cannot be done in a uniform way (using variables only), and it also yields a combinatoric explosion because first a rewriting to basic form is needed. For these models K, $\mathbf{SAT}_K$ is known to be NP-complete.

12. EXPRESSIVENESS

In this section, we first show that the ternary conditional operator cannot be replaced by $\neg$ and $\mathbin{_{\circ}\!\!\!\wedge}$ and one of T and F (which together define $\mathbin{^{\circ}\!\!\!\vee}$) modulo free valuation congruence. Then, we show that this is the case for any collection of unary and binary operators each of which is definable in $\Sigma_{\mathrm{CP}}(A)$ with free valuation congruence, and in a next theorem we lift this result to contractive valuation congruence. Finally, we observe that the conditional operator is definable with $\neg$ and $\mathbin{_{\circ}\!\!\!\wedge}$ and one of T and F modulo memorizing valuation congruence. We were unable to decide whether this is also the case in weakly memorizing valuation congruence.

An occurrence of an atom a in a propositional statement over A, $\neg$, $\mathbin{_{\circ}\!\!\!\wedge}$ and $\mathbin{^{\circ}\!\!\!\vee}$ is *redundant* or *inaccessible* if it is not used along any of the possible arbitrary valuations, as in for example $F \mathbin{_{\circ}\!\!\!\wedge} a$.

The opposite of redundancy is accessibility, which is defined thus ($acc(\phi) \subseteq A$):

$$
\begin{aligned}
acc(a) &= \{a\},\\
acc(T) &= \emptyset,\\
acc(\neg x) &= acc(x),\\
acc(x \mathbin{{}_{\circ}\!\wedge} y) &= \begin{cases} acc(x) & \text{if } \neg\mathbf{SAT}_{fr}(x),\\ acc(x) \cup acc(y) & \text{if } \mathbf{SAT}fr(x),\end{cases}\\
acc(x \mathbin{{}^{\circ}\!\vee} y) &= \begin{cases} acc(x) & \text{if } \neg\mathbf{SAT}_{fr}(\neg x),\\ acc(x) \cup acc(y) & \text{if } \mathbf{SAT}_{fr}(\neg x).\end{cases}
\end{aligned}
$$

PROPOSITION 12.1. *Let $a \in A$, then the propositional statement $a \triangleleft a \triangleright \neg a$ cannot be expressed in $\Sigma_{\mathrm{CP}}(A)$ with free valuation congruence using $\mathbin{{}_{\circ}\!\wedge}$, $\mathbin{{}^{\circ}\!\vee}$, $\neg$, T and F only.*

PROOF. Let ψ be a minimal expression of $a \triangleleft a \triangleright \neg a$.
Assume $\psi \equiv \psi_0 \mathbin{{}^{\circ}\!\vee} \psi_1$. We notice:

(1) Both ψ_0 and ψ_1 must contain a: If ψ_0 contains no a, it is either T and then ψ always yields T which is wrong, or F and then ψ can be simplified; if ψ_1 contains no a it is either T and then ψ always yields T, which is wrong, or F and then $\mathbin{{}^{\circ}\!\vee} F$ can be removed so ψ was not minimal.
(2) ψ_0 can yield F otherwise ψ is not minimal. It will do so after using exactly one test a (yielding F without a use of a simply means that $a \notin acc(\psi_0)$), yielding F after two uses of a implies that evaluation of ψ has at least three uses of a (which is wrong).
(3) If ψ_0 yields T, this completes the evaluation of ψ, so then the evaluation of ψ_0 involves two uses of a. If ψ_0 yields F, it must contain at least one use of a (but no more because otherwise evaluation of ψ_1 yields at least a third use of a).

Thus, $\psi_0 = F \triangleleft a \triangleright (\overline{a} \mathbin{{}^{\circ}\!\vee} T)$ or $\psi_0 = F \triangleleft \neg a \triangleright (\overline{a} \mathbin{{}^{\circ}\!\vee} T)$, where the $\overline{a}$ in the right-hand sides equals either a or $\neg a$, and these sides take their particular form by minimality (other forms are $T \mathbin{{}^{\circ}\!\vee} \overline{a} = T$, etc.). But both are impossible as both imply that after a first use of a the final value of ψ can be independent of the second value returned for a which is not true for $a \triangleleft a \triangleright \neg a$.
For the case $\psi \equiv \psi_0 \mathbin{{}_{\circ}\!\wedge} \psi_1$, a similar type of reasoning applies. $\Box$

In this section, we will prove two more general results. We first introduce some auxiliary definitions and notations because we have to be precise about definability by unary and binary operators. For X, a countable set of variables, we define the following sets:

$T_C(X)$: the set of terms over $X, T, F, \_ \triangleleft \_ \triangleright \_$.
$T_{TND}(X)$: the set of terms over $X, T, \neg, \mathbin{{}^{\circ}\!\vee}$.
$T_C^{1,2}(X)$: the smallest set of terms V such that

$$
\begin{cases}
T, F \in V,\\
\text{if } x \in X \text{ and } t \in T_C(\{x\}) \text{ then } t \in V,\\
\text{if } x, y \in X \text{ and } t \in T_C(\{x, y\}) \text{ then } t \in V,\\
V \text{ is closed under substitution.}
\end{cases}
$$

Thus, $T_C^{1,2}(X)$ contains the terms that can be made from unary and binary operators definable in $T_C(X)$. For $t \in T_C^{1,2}(\{x\})$, we sometimes write $t(x)$ instead of t, and if $s \in T_C^{1,2}(X)$, we write $t(s)$ for the term obtained by substituting s for all x in t. Similarly, if $u \in T_C^{1,2}(\{x, y\})$, we may write $u(x, y)$ for u, and if $s, s' \in T_C^{1,2}(X)$, we write $u(s, s')$ for the term obtained by substituting s for x and s' for y in u. Finally, we define $\#_{2p}(t)$ as the number of 2-place terms used in the definition of t, that is,

$$\begin{aligned}
\#_{2p}(x) = \#_{2p}(T) = \#_{2p}(F) = 0,\\
\#_{2p}(t(s)) = \#_{2p}(t) + \#_{2p}(s),\\
\#_{2p}(u(s, s')) = \#_{2p}(u) + \#_{2p}(s) + \#_{2p}(s').
\end{aligned}$$

Notice $T_{TND}(X) \subseteq_K T_C^{1,2}(X) \subseteq_K T_C(X)$, where $M \subseteq_K N$ if, for each term $t \in M$, there is a term $r \in N$ with $r =_K t$ for $K \in \{fr, rp, cr, wm, mem, st\}$. We write $\in_K$ for the membership relation associated with $\subseteq_K$.

The sets $T_{TND}(A)$, $T_C^{1,2}(A)$ and $T_C(A)$ contain the closed substitution instances of the respective term sets when constants from A are substituted for the variables. The set $T_C^{1,2}(A, X)$ contains the terms constructed from $T_C^{1,2}(A)$ and $T_C^{1,2}(X)$. For given terms $r(x) \in T_C^{1,2}(A, \{x\})$ and $t \in T_C^{1,2}(A)$ we write $r(t)$ for the term obtained by substituting t for all x in r (thus $r(t) \equiv [t/x]r(x)$). We extend the definition of $\#_{2p}(t)$ to $T_C^{1,2}(A, X)$ in the expected way by defining $\#_{2p}(a) = 0$ for all $a \in A$.

Clearly, for all K,

$$T_{TND}(A) \subseteq_K T_C^{1,2}(A) \subseteq_K T_C(A).$$

From Proposition 12.1, we find that $a \lhd a \rhd \neg a \notin_{fr} T_{TND}(A)$, thus

$$T_{TND}(A) \subsetneq_{fr} T_C^{1,2}(A).$$

Theorem 12.2 establishes that $T_C^{1,2}(A) \subsetneq_{fr} T_C(A)$ as $a \lhd b \rhd c \notin_{fr} T_C^{1,2}(A)$. This result transfers to rp-congruence, without modification. However, in wm-congruence, we find

$$a \lhd b \rhd c =_{wm} (T \lhd b \rhd c) \lhd (a \lhd b \rhd T) \rhd F = (\neg b \mathbin{{}^{\circ}\!\vee} a) \mathbin{{}_{\circ}\!\wedge} (b \mathbin{{}^{\circ}\!\vee} c),$$

thus $a \lhd b \rhd c \in_{wm} T_C^{1,2}(A)$.

THEOREM 12.2. *If $|A| > 2$, then the conditional operator cannot be expressed modulo free valuation congruence in $T_C^{1,2}(X)$.*

PROOF. It is sufficient to prove that $a \lhd b \rhd c \notin_{fr} T_C^{1,2}(A)$.

Towards a contradiction, assume $t \in T_C^{1,2}(A)$ is a term such that $t =_{fr} a \lhd b \rhd c$ and $\#_{2p}(t)$ is minimal (i.e., if $u \in T_C^{1,2}(A)$ and $u =_{fr} t$ then $\#_{2p}(u) \geq \#_{2p}(t)$).

We first argue that $t \not\equiv f(b, t')$ for some binary function f and term t'. Suppose otherwise, then b must be the central condition in $f(b, t')$, so $f(b, t') =_{fr} g(b, t') \lhd b \rhd h(b, t')$ for certain binary functions g and h in $T_C^{1,2}(X)$. Because it is neither the case that b can occur as a central condition in both $g(b, t')$ and in $h(b, t')$, nor that each of these can be modulo fr in $\{T, F\}$, we find

$$t =_{fr} (P \lhd t' \rhd Q) \lhd b \rhd (P' \lhd t' \rhd Q')$$

for certain P, P', Q, Q'. The only possibilities left are that the central atom of t' is either a or c, and both choices contradict $f(b, t') =_{fr} a \lhd b \rhd c$.

So it must be the case that

$$t \equiv r(f(b, t'))$$

for some term $r(x) \in T_C^{1,2}(\{x\})$ such that b is central in $f(b, t')$ and x is central in $r(x)$. If no such term $r(x)$ exists, then $t \equiv f'(a')$ with $f'(x)$ a unary operator definable in $T_C^{1,2}(\{x\})$ and $a' \in A$, which cannot hold because t needs to contain a, b and c.

Also there cannot be a unary function $f' \in T_C^{1,2}(\{x\})$ with $r(f'(b)) =_{fr} r(f(b,t'))$, otherwise $r(f'(b)) \in T_C^{1,2}(A)$ while

$$\#_{2p}(r) = \#_{2p}(r(f'(b))) < \#_{2p}(r(f(b,t'))) = \#_{2p}(r) + \#_{2p}(t') + 1,$$

which contradicts the minimality of $\#_{2p}(t)$.

As x is central in $f(x, y)$, we may write

$$f(x, y) =_{fr} g(x, y) \triangleleft x \triangleright h(x, y)$$

for certain binary functions g and h in $T_C^{1,2}(X)$. Because b is central in t, we find

$$t =_{fr} r(g(b,t') \triangleleft b \triangleright h(b,t')).$$

We proceed with a case distinction on the form that $g(b,t')$ and $h(b,t')$ may take. At least one of these is modulo fr not equal to T or F (otherwise, $f(b,t')$ could be replaced by $f'(b)$ for some unary function f' and this was excluded).

(1) Suppose $g(b,t') \notin_{fr} \{T, F\}$ and $h(b,t') \notin_{fr} \{T, F\}$. First, notice that b cannot occur as a central condition in both $g(b,t')$ and in $h(b,t')$. So, both $g(x, y)$ and $h(x, y)$ can be written as a conditional composition with y as the central variable, and we find

$$t =_{fr} r((P \triangleleft t' \triangleright Q) \triangleleft b \triangleright (P' \triangleleft t' \triangleright Q'))$$

for certain closed terms P, Q, P', Q'. By supposition $t' \notin_{fr} \{T, F\}$, and the only possibilities left are that its central atom equals both a and c, which clearly is impossible.

(2) We are left with four cases: either a is central in $g(b,t')$ and $h(b,t') \in_{fr} \{T, F\}$, or c is central in $h(b,t')$ and $g(b,t') \in_{fr} \{T, F\}$. These cases are symmetric and it suffices to consider only the first one, the others can be dealt with similarly.

So assume a is central in $g(b,t')$ and $h(b,t') =_{fr} T$, hence

$$g(b,t') =_{fr} P \triangleleft a \triangleright Q \quad \text{for some } P, Q \in \{T, F\}.$$

We find

$$t =_{fr} r((P \triangleleft a \triangleright Q) \triangleleft b \triangleright T),$$

and we distinguish two cases:

(i) $P \equiv T$ or $Q \equiv T$. Now a central c can be reached after a negative reply to b. But this central c can also be reached after a positive reply to b and the appropriate reply to a, which contradicts free congruence with $a \triangleleft b \triangleright c$.

(ii) $P \equiv Q \equiv F$. Then, the reply to a in $r((F \triangleleft a \triangleright F) \triangleleft b \triangleright T)$ is not used, which also contradicts free congruence with $a \triangleleft b \triangleright c$.

This concludes our proof. □

We will now argue that $a \triangleleft b \triangleright c \notin_{cr} T_C^{1,2}(A)$. We will make use of additional operators T_a and F_a for each atom $a \in A$, defined for all $b \in A$ and terms $t, r \in T_C^{1,2}(A)$ by

$$\begin{aligned} T_a(T) &= T, & F_a(T) &= F,\\ T_a(F) &= T, & F_a(F) &= F,\\ T_a(t \triangleleft b \triangleright r) &= t \triangleleft b \triangleright r \ \text{ if } a \neq b, & F_a(t \triangleleft b \triangleright r) &= t \triangleleft b \triangleright r \ \text{ if } a \neq b,\\ T_a(t \triangleleft a \triangleright r) &= T_a(t), & F_a(t \triangleleft a \triangleright r) &= F_a(r). \end{aligned}$$

Observe that T_a (F_a) simplifies a term t as if it is a subterm of $a \circ t$ with the additional knowledge that the reply on a has been T. We notice that

$$t \triangleleft a \triangleright r =_{cr} T_a(t) \triangleleft a \triangleright F_a(r).$$

We define a term P to have the property $\phi_{a,b,c}$ if

—the central atom of $T_b(P)$ equals a, $T_a(T_b(P)) \in_{cr} \{T, F\}$ and $F_a(T_b(P)) \in_{cr} \{T, F\}$, and $T_a(T_b(P)) \neq_{cr} F_a(T_b(P))$,
—the central atom of $F_b(P)$ equals c, $T_c(T_b(P)) \in_{cr} \{T, F\}$ and $F_c(T_b(P)) \in_{cr} \{T, F\}$, and $T_c(T_b(P)) \neq_{cr} F_c(T_b(P))$.

Typically, $a \triangleleft b \triangleright c$ has property $\phi_{a,b,c}$.

THEOREM 12.3. *If $|A| > 2$, then the conditional operator cannot be expressed modulo contractive valuation congruence in $T_C^{1,2}(X)$.*

PROOF. Let $a, b, c \in A$. It is sufficient to show that no term in $T_C^{1,2}(A)$ has property $\phi_{a,b,c}$. A detailed proof of this fact is included in Appendix A. □

Finally, we observe that $x \triangleleft y \triangleright z$ is expressible in CP_{mem} using $\mathbin{{}_{\circ}\!\wedge}$ and $\neg$ only: first $\mathbin{{}^{\circ}\!\vee}$ is expressible, and

$$
\begin{aligned}
\mathrm{CP}_{mem} \vdash (y \mathbin{{}_{\circ}\!\wedge} x) \mathbin{{}^{\circ}\!\vee} (\neg y \mathbin{{}_{\circ}\!\wedge} z) &= T \triangleleft (x \triangleleft y \triangleright F) \triangleright (z \triangleleft (F \triangleleft y \triangleright T) \triangleright F)\\
&= T \triangleleft (x \triangleleft y \triangleright F) \triangleright (F \triangleleft y \triangleright z)\\
&= (T \triangleleft x \triangleright (F \triangleleft y \triangleright z)) \triangleleft y \triangleright (F \triangleleft y \triangleright z)\\
&\overset{(11)}{=} (T \triangleleft x \triangleright F) \triangleleft y \triangleright (F \triangleleft y \triangleright z)\\
&\overset{(13)}{=} x \triangleleft y \triangleright z.
\end{aligned}
$$

Thus, for $x, y, z \in X$, it holds that $(x \triangleleft y \triangleright z) \in_{mem} T_C^{1,2}(X)$. We leave it as an open question whether $(x \triangleleft y \triangleright z) \in_{wm} T_C^{1,2}(X)$.

13. PROJECTIONS AND THE PROJECTIVE LIMIT MODEL

In this section, we introduce the *projective limit model* $\mathbb{A}^\infty$ that contains finite as well as infinite propositions. A simple, intuitive example of a potentially infinite proposition is

$$\textit{while } \neg a \textit{ test } b,$$

which as long as a yields F tests b. An infinite proposition such as this one can be meaningful in free or reactive valuation semantics. In the next section, we return to this example.

Let $\mathcal{P}$ be the domain of the initial algebra of CP. We assume that each element in $\mathcal{P}$ is represented by its (equivalent) basic form, but we shall often write a for $T \triangleleft a \triangleright F$. Let $\mathbb{N}^+$ denote $\mathbb{N} \setminus \{0\}$. We first define a so-called *projection* operator

$$\pi : \mathbb{N}^+ \times \mathcal{P} \to \mathcal{P},$$

which will be used to finitely approximate every proposition in $\mathcal{P}$. We further write

$$\pi_n(P)$$

instead of $\pi(n, P)$. The defining equations for the π_n-operators are these ($n \in \mathbb{N}^+$):

$$\pi_n(T) = T, \tag{24}$$
$$\pi_n(F) = F, \tag{25}$$
$$\pi_1(x \triangleleft a \triangleright y) = a, \tag{26}$$
$$\pi_{n+1}(x \triangleleft a \triangleright y) = \pi_n(x) \triangleleft a \triangleright \pi_n(y), \tag{27}$$

for all $a \in A$. We write PR for this set of equations. It follows by structural induction on representatives that for each proposition $P \in \mathcal{P}$, there exists $n \in \mathbb{N}^+$ such that for all $j \in \mathbb{N}$,

$$\pi_{n+j}(P) = P.$$

We state without proof that CP + PR is a conservative extension of CP and mention the following derivable identities in CP + PR for $a \in A$ and $n \in \mathbb{N}^+$:

$$\begin{aligned} \pi_n(a) &= \pi_n(T \triangleleft a \triangleright F) = a, \\ \pi_{n+1}(a \circ x) &= a \circ \pi_n(x). \end{aligned}$$

The following lemma establishes how the arguments of the projection of a conditional composition can be restricted to certain projections, in particular

$$\pi_n(P \triangleleft Q \triangleright R) = \pi_n(\pi_n(P) \triangleleft \pi_n(Q) \triangleright \pi_n(R)), \tag{28}$$

which is a property that we will use in the definition of our projective limit model.

Lemma 13.1. *For all $P, Q, R \in \mathcal{P}$ and all $n \in \mathbb{N}^+$, $k, \ell, m \in \mathbb{N}$,*

$$\pi_n(P \triangleleft Q \triangleright R) = \pi_n(\pi_{n+k}(P) \triangleleft \pi_{n+\ell}(Q) \triangleright \pi_{n+m}(R)).$$

Proof. We may assume that Q is a basic form and we apply structural induction on Q.

If $Q \equiv T$ then we have to prove that for all $n \in \mathbb{N}^+$ and $k \in \mathbb{N}$,

$$\pi_n(P) = \pi_n(\pi_{n+k}(P)).$$

We apply structural induction on P. If $P \in \{T, F\}$ we are done. If $P \equiv P_1 \triangleleft a \triangleright P_2$ then we proceed by induction on n. The case $n = 1$ is trivial, and

$$\begin{aligned} \pi_{n+1}(P) &= \pi_{n+1}(P_1 \triangleleft a \triangleright P_2) \\ &= \pi_n(P_1) \triangleleft a \triangleright \pi_n(P_2) \\ &\overset{IH}{=} \pi_n(\pi_{n+k}(P_1)) \triangleleft a \triangleright \pi_n(\pi_{n+k}(P_2)) \\ &= \pi_{n+1}(\pi_{n+k}(P_1) \triangleleft a \triangleright \pi_{n+k}(P_2)) \\ &= \pi_{n+1}(\pi_{n+k+1}(P)). \end{aligned}$$

If $Q \equiv F$: similar.

If $Q \equiv Q_1 \triangleleft a \triangleright Q_2$ then we proceed by induction on n. The case $n = 1$ is trivial, and

$$\begin{aligned} \pi_{n+1}(P \triangleleft Q \triangleright R) &= \pi_{n+1}(P \triangleleft (Q_1 \triangleleft a \triangleright Q_2) \triangleright R) \\ &= \pi_{n+1}((P \triangleleft Q_1 \triangleright R) \triangleleft a \triangleright (P \triangleleft Q_2 \triangleright R)) \\ &= \pi_n(P \triangleleft Q_1 \triangleright R) \triangleleft a \triangleright \pi_n(P \triangleleft Q_2 \triangleright R) \\ &\overset{IH}{=} \pi_n(\pi_{n+k+1}(P) \triangleleft \pi_{n+\ell}(Q_1) \triangleright \pi_{n+m+1}(R)) \triangleleft a \triangleright \\ &\qquad \pi_n(\pi_{n+k+1}(P) \triangleleft \pi_{n+\ell}(Q_2) \triangleright \pi_{n+m+1}(R)) \\ &= \pi_{n+1}((\pi_{n+k+1}(P) \triangleleft \pi_{n+\ell}(Q_1) \triangleright \pi_{n+m+1}(R)) \triangleleft a \triangleright \\ &\qquad (\pi_{n+k+1}(P) \triangleleft \pi_{n+\ell}(Q_2) \triangleright \pi_{n+m+1}(R))) \\ &= \pi_{n+1}(\pi_{n+k+1}(P) \triangleleft (\pi_{n+\ell}(Q_1) \triangleleft a \triangleright \pi_{n+\ell}(Q_2)) \triangleright \pi_{n+m+1}(R)) \\ &= \pi_{n+1}(\pi_{n+1+k}(P) \triangleleft \pi_{n+1+\ell}(Q) \triangleright \pi_{n+1+m}(R)). \end{aligned}$$

□

The *projective limit model* $\mathbb{A}^\infty$ is defined as follows.

—The domain of $\mathbb{A}^\infty$ is the set of *projective sequences* $(P_n)_{n\in\mathbb{N}^+}$: these are all sequences with the property that all P_n are in $\mathcal{P}$ and satisfy

$$\pi_n(P_{n+1}) = P_n,$$

so that they can be seen as successive projections of the same infinite proposition (observe that $\pi_n(P_n) = P_n$). We further write $(P_n)_n$ instead of $(P_n)_{n\in\mathbb{N}^+}$.

—Equivalence of projective sequences in $\mathbb{A}^\infty$ is defined component-wise, thus

$$(P_n)_n = (Q_n)_n \quad \text{if for all } n,\ P_n = Q_n.$$

—The constants T and F are interpreted in $\mathbb{A}^\infty$ as the projective sequences that consist solely of these respective constants.

—An atomic proposition a is interpreted in $\mathbb{A}^\infty$ as the projective sequence

$$(a,\ a,\ a, \ldots).$$

—Projection in $\mathbb{A}^\infty$ is defined component-wise, thus $\pi_k((P_n)_n) = (\pi_k(P_n))_n$.

—Conditional composition in $\mathbb{A}^\infty$ is defined using projections:

$$(P_n)_n \triangleleft (Q_n)_n \triangleright (R_n)_n = (\pi_n(P_n \triangleleft Q_n \triangleright R_n))_n.$$

The projections are needed if the depth of a component $P_n \triangleleft Q_n \triangleright R_n$ exceeds n. Equation (28) implies that this definition indeed yields a projective sequence:

$$\begin{aligned}\pi_n(\pi_{n+1}(P_{n+1} \triangleleft Q_{n+1} \triangleright R_{n+1})) &= \pi_n(P_{n+1} \triangleleft Q_{n+1} \triangleright R_{n+1})\\ &= \pi_n(\pi_n(P_{n+1}) \triangleleft \pi_n(Q_{n+1}) \triangleright \pi_n(R_{n+1}))\\ &= \pi_n(P_n \triangleleft Q_n \triangleright R_n).\end{aligned}$$

The following result can be proved straightforwardly.

THEOREM 13.2. $\mathbb{A}^\infty \models \mathrm{CP} + \mathrm{PR}$.

The projective limit model $\mathbb{A}^\infty$ contains elements that are not the interpretation of finite propositions in $\mathcal{P}$ (in other words, elements of infinite depth). In the next section, we discuss some examples.

14. RECURSIVE SPECIFICATIONS

In this section, we discuss *recursive specifications* over $\Sigma_{\mathrm{CP}}(A)$, which provide an alternative and simple way to define propositions in $\mathbb{A}^\infty$. We first restrict ourselves to a simple class of recursive specifications: Given $\ell > 0$, a set

$$E = \{X_i = t_i \mid i = 1, \ldots, \ell\}$$

of equations is a *linear specification* over $\Sigma_{\mathrm{CP}}(A)$ if

$$t_i ::= T \mid F \mid X_j \triangleleft a_i \triangleright X_k$$

for $i, j, k \in \{1, \ldots, \ell\}$ and $a_i \in A$. A *solution* for E in $\mathbb{A}^\infty$ is a series of propositions

$$(P_{1,n})_n, \ldots, (P_{\ell,n})_n$$

such that $(P_{i,n})_n$ solves the equation for X_i. In $\mathbb{A}^\infty$, solutions for linear specifications exist. This follows from the property that for each $m \in \mathbb{N}^+$, $\pi_m(X_i)$ can be computed as a proposition in $\mathcal{P}$ by replacing variables X_j by t_j sufficiently often. For example, if

$$E = \{X_1 = X_3 \triangleleft a \triangleright X_2,\ X_2 = b \circ X_1,\ X_3 = T\}$$

we find $\pi_m(X_3) = \pi_m(T) = T$ for all $m \in \mathbb{N}^+$, and

$$\begin{aligned} \pi_1(X_2) &= \pi_1(b \circ X_1) & \pi_{m+1}(X_2) &= \pi_{m+1}(b \circ X_1) \\ &= b, & &= b \circ \pi_m(X_1), \\ \pi_1(X_1) &= \pi_1(X_3 \triangleleft a \triangleright X_2) & \pi_{m+1}(X_1) &= \pi_{m+1}(X_3 \triangleleft a \triangleright X_2) \\ &= a, & &= T \triangleleft a \triangleright \pi_m(X_2), \end{aligned}$$

and we can in this way construct a projective sequence per variable. We state without proof that for a linear specification $E = \{X_i = t_i \mid i = 1, \ldots, \ell\}$ such sequences model unique solutions in $\mathbb{A}^\infty$,[1] and we write

$$\langle X_i | E \rangle$$

for the solution of X_i as defined in E. In order to reason about linearly specified propositions, we add these constants to the signature Σ_{CP}. These constants satisfy the equations

$$\langle X_i | E \rangle = \langle t_i | E \rangle$$

where $\langle t_i | E \rangle$ is defined by replacing each X_j in t_i by $\langle X_j | E \rangle$. The proof principle introducing these identities is called the *Recursive Definition Principle* (RDP), and for linear specifications RDP is valid in the projective limit model $\mathbb{A}^\infty$.[2] As illustrated above, all solutions satisfy

$$\langle X_i | E \rangle = (\pi_n(\langle X_i | E \rangle))_n.$$

Some examples of propositions defined by recursive specifications are these:

(1) For $E = \{X_1 = X_2 \triangleleft a \triangleright X_3,\ X_2 = T,\ X_3 = F\}$, we find

$$\langle X_1 | E \rangle = (a,\ a,\ a, \ldots)$$

which in the projective limit model represents the atomic proposition a. Indeed, by RDP we find $\langle X_1 | E \rangle = \langle X_2 | E \rangle \triangleleft a \triangleright \langle X_3 | E \rangle = T \triangleleft a \triangleright F = a$.

(2) For $E = \{X_1 = X_2 \triangleleft a \triangleright X_3,\ X_2 = T,\ X_3 = T\}$, we find

$$\langle X_1 | E \rangle = (a,\ a \circ T,\ a \circ T,\ a \circ T, \ldots)$$

which in the projective limit model represents $a \circ T$. By RDP, we find $\langle X_1 | E \rangle = a \circ T$.

(3) For $E = \{X_1 = X_3 \triangleleft a \triangleright X_2,\ X_2 = b \circ X_1,\ X_3 = T\}$ as discussed above, we find

$$\langle X_1 | E \rangle = (a,\ T \triangleleft a \triangleright b,\ T \triangleleft a \triangleright b \circ a,\ T \triangleleft a \triangleright b \circ (T \triangleleft a \triangleright b), \ldots)$$

which in the projective limit model represents an *infinite* propositional statement, that is, one that satisfies

$$\pi_i(\langle X_1 | E \rangle) = \pi_j(\langle X_1 | E \rangle) \quad \Rightarrow \quad i = j,$$

and thus has infinite depth. By RDP, we find $\langle X_1 | E \rangle = T \triangleleft a \triangleright b \circ \langle X_1 | E \rangle$. We note that the infinite propositional statement $\langle X_1 | E \rangle$ can be characterized as

while $\neg a$ *test* b.

---

[1] The domain of $\mathbb{A}^\infty$ can be turned into a metric space by defining $d((P_n)_n, (Q_n)_n) = 2^{-n}$ for n the least value with $P_n \neq Q_n$. The existence of unique solutions for linear specifications then follows from Banach's fixed point theorem; a comparable and detailed account of this fact can be found in Vu [2008].

[2] A nice and comparable account of the validity of RDP in the projective limit model for ACP is given in Baeten and Weijland [1990]. In that text book, a sharp distinction is made between RDP—stating that certain recursive specifications have *at least* a solution per variable—and the Recursive Specification Principle (RSP), stating that they have *at most* one solution per variable. The uniqueness of solutions per variable then follows by establishing the validity of both RDP and RSP.

An example of a projective sequence that cannot be defined by a linear specification, but that can be defined by the *infinite* linear specification $I = \{X_i = t_i \mid i \in \mathbb{N}^+\}$ with

$$t_i = \begin{cases} a \circ X_{i+1} & \text{if } i \text{ is prime,} \\ b \circ X_{i+1} & \text{otherwise,} \end{cases}$$

is $\langle X_1 | I \rangle$, satisfying

$$\langle X_1 | I \rangle = (b,\ b \circ a,\ b \circ a \circ a,\ b \circ a \circ a \circ b,\ b \circ a \circ a \circ b \circ a, \ldots).$$

Other examples of projective sequences that cannot be defined by a finite linear specification are $\langle X_j | I \rangle$ for any $j > 1$.

Returning to Example (1) of a propositional statement sketched in Section 2, we can be more explicit now: the recursively defined proposition $\langle X_1 | E \rangle$ with E containing

$$\begin{aligned} X_1 &= X_2 \triangleleft \mathit{green\text{-}light} \triangleright X_1, \\ X_2 &= X_3 \triangleleft (\mathit{look\text{-}left\text{-}and\text{-}check} \mathbin{\overset{\circ}{\wedge}} \mathit{look\text{-}right\text{-}and\text{-}check} \mathbin{\overset{\circ}{\wedge}} \mathit{look\text{-}left\text{-}and\text{-}check}) \triangleright X_1, \\ X_3 &= \cdots \end{aligned}$$

models in a straightforward way a slightly larger part of the processing of a pedestrian planning to cross a road with two-way traffic driving on the right.

15. CONCLUSIONS

Proposition algebra in the form of CP for propositional statements with conditional composition and either enriched or not with negation and sequential connectives, is proposed as an abstract data type. Free valuations provide the natural semantics for CP and these are semantically at the opposite end of static valuations. It is shown that taking conditional composition and free valuations as a point of departure implies that a ternary connective is needed for functional completeness; binary connectives are not sufficient. Furthermore, CP admits a meaningful and non-trivial extension to projective limits, and this constitutes the most simple case of an inverse limit construction that we can think of.

The potential role of proposition algebra is only touched upon by some examples. It remains a challenge to find convincing examples that require reactive valuations, and to find earlier accounts of this type of semantics for propositional logic. The basic idea of proposition algebra with free and reactive valuations can be seen as a combination of the following two ideas.

—Consider atomic propositions as events (queries) that can have a side effect in a sequential system, and take McCarthy's sequential evaluation [1963] to two-valued propositional logic; this motivates reactive valuations as those that define evaluation or computation as a sequential phenomenon.
—In the resulting setting, Hoare's conditional composition [1985b] is more natural than the sequential, noncommutative versions of conjunction and disjunction, and (as it appears) more expressive: a ternary connective is needed anyhow.

For conditional composition, we have chosen for the notation

$$\_ \triangleleft \_ \triangleright \_$$

from Hoare [1985b] in spite of the fact that our theme is technically closer to thread algebra [Bergstra and Middelburg 2007] where a different notation is used. We chose for the notation $\_ \triangleleft \_ \triangleright \_$ because its most well-known semantics is static valuation semantics (which is simply conventional propositional logic) for which this notation was introduced in Hoare [1985b].[3] To some extent, thread algebra and propositional logic in the style of Hoare [1985b] are models of the same signature. A much more involved use of conditional composition can be found in Ponse and van der Zwaag [2007], where the propositional fragment of Belnap's four-valued logic [1977] is characterized using only conditional composition and his four constants representing these truth values.

Apart from the valuation congruences introduced in Section 5 and variations thereof such as $Pmem$ which was briefly discussed in Section 11, many more valuation congruences can be distinguished. As an example we mention here mem_k with $k \in \mathbb{N}$: valuations that remember the result of each atomic evaluation during k subsequent atomic evaluations. The valuation congruence defined by mem_k can be seen as a natural generalization of repetition-proof valuation congruence ($=_{rp}$): mem_1-congruence coincides with $=_{rp}$ and mem_0-congruence coincides with free valuation congruence.

In this article we assumed that $|A| > 1$. The case that $|A| = 1$ is in detail described in Regenboog [2010]. In particular, $=_{rp}$ and $=_{st}$ and thus all valuation congruences in between coincide in this case.

Related work. We end with a few notes on related matters.

(1) In quite a few papers the "lazy evaluation" semantics proposed in McCarthy's work on conditional expressions in [1963] is discussed, or taken as a point of departure. We mention a few of these works in reverse chronological order:
   (a) Hähnle states in his paper *Many-valued logic, partiality, and abstraction in formal specification languages* [2005] that
   > *"sequential conjunction [...] represents the idea that if the truth value can be determined after evaluation of the first argument, then the result is computed without looking at the second argument. Many programming languages contain operators that exhibit this kind of behavior".*

   (b) Konikowska [1996] describes a model of so-called McCarthy algebras in terms of three-valued logic, while restricting to the well-known symmetric binary connectives, and provides sound axiomatizations and representation results. This is achieved by admitting only T and F as constants in a McCarthy algebra, and distinguishing an element a as in one of four possible classes ("positive" if $a \vee x = a$, "negative" if $a \wedge x = a$, "defined" if $a \wedge \neg a = F$, and "strictly undefined" if $a = \neg a$).
   (c) Finally, Bloom and Tindell discuss in their paper *Varieties of "if-then-else"* [1983] various modelings of conditional composition, both with and without a truth value *undefined*, while restricting to the "redundancy law"
   $$(x \lhd y \rhd z) \lhd y \rhd u = x \lhd y \rhd u,$$
   a law that we called CPcontr in Section 9 and that generalizes the axiomatization of contractive valuation congruence defined in that section to an extent in which only the difference between T, F and *undefined* plays a significant role.

   As far as we can see, none of the papers mentioned here even suggests the idea of *free* or *reactive* valuation semantics. Another example where sequential operators play a role is *Quantum logic* as formulated by Rehder [1980] and Mittelstaedt [2004], where next to normal conjunction a notion of *sequential conjunction* $\sqcap$ is exploited that is very similar to $\mathbin{\circ\!\!\wedge}$ (and that despite its notation is certainly not symmetric).

[3]This notation was used by Hoare in his 1985 book on CSP [1985a] and in his well-known 1987 paper *Laws of Programming* [Hayes et al. 1987] for expressions $P \lhd b \rhd Q$ with P and Q programs and b a Boolean expression without mention of Hoare [1985b].

(2) Concerning projections and the projective limit model $\mathbb{A}^\infty$ we mention that in much current research and exposition, projections are defined also for depth 0 (see, e.g., Bergstra and Middelburg [2007] and Vu [2008] for the case of thread algebra, and Fokkink [2000] for process algebra). However, CP does not have a natural candidate for $\pi_0(P)$ and therefore we stick to the original approach as described in Bergstra and Klop [1984] (and overviewed in Baeten and Weijland [1990]) that starts from projections with depth 1.

(3) Free valuation semantics was in a different form employed in accounts of process algebra with propositional statements: in terms of operational semantics, this involves transitions

$$P \overset{a,w}{\to} Q$$

for process expressions P and Q with a an action and w ranging over a class of valuations. In particular, this approach deals with process expressions that contain propositional statements in the form of guarded commands, such as $\phi :\to P$ that has a transition

$$(\phi :\to P) \overset{a,w}{\to} Q$$

if $P \overset{a,w}{\to} Q$ and $w(\phi) = T$. For more information about this approach, see, for example, Bergstra and Ponse [1998b, 1998a].

APPENDIX A

In this appendix, we provide a detailed proof of Theorem 12.3.

Proof of Theorem 12.3. This proof has the same structure as the proof of Theorem 12.2, but a few cases require more elaboration.

Towards a contradiction, assume that $t \in T_C^{1,2}(A)$ is a term with property $\phi_{a,b,c}$ and $\#_{2p}(t)$ is minimal.

We first argue that $t \not\equiv f(b, t')$ for some binary function f and term t'. Suppose otherwise, then b must be the central condition in $f(b, t')$, so $f(b, t') =_{cr} g(b, t') \triangleleft b \triangleright h(b, t')$ for certain binary functions g and h in $T_C^{1,2}(X)$. Notice that because b is not central in $T_b(g(b, t'))$, a different atom must be central in this term, and this atom must be a. For this to hold, a must be central in $T_b(t')$ and no atom different from a can be tested by the first requirement of $\phi_{a,b,c}$. So, after contraction of all further a's we find

$$T_b(t') =_{cr} P \triangleleft a \triangleright Q$$

with $P, Q \in \{T, F\}$, and similarly

$$F_b(t') =_{cr} P' \triangleleft c \triangleright Q'$$

with $P', Q' \in \{T, F\}$. If $P \not\equiv Q$ and $P' \not\equiv Q'$, then t' is a term that satisfies $\phi_{a,b,c}$, but t' is a term with lower $\#_{2p}$-value than $g(b, t') \triangleleft b \triangleright h(b, t')$, which is a contradiction. If either $P \equiv Q$ or $P' \equiv Q'$, then

$$t =_{cr} (P \triangleleft a \triangleright Q) \triangleleft b \triangleright (P' \triangleleft c \triangleright Q'),$$

which contradicts $\phi_{a,b,c}$.

So it must be the case that

$$t \equiv r(f(b, t'))$$

for some term $r(x) \in T_C^{1,2}(\{x\})$ such that b is central in $f(b, t')$ and x is central in $r(x)$. If no such such term $r(x)$ exists, then $t \equiv f'(a')$ with $f'(x)$ a unary operator definable in $T_C^{1,2}(\{x\})$ and $a' \in A$, which cannot hold because t needs to contain a, b and c.

Also there cannot be a unary function $f' \in T_C^{1,2}(\{x\})$ with $r(f'(b)) =_{cr} r(f(b,t'))$, otherwise $r(f'(b)) \in T_C^{1,2}(A)$ while $\#_{2p}(r(f'(b))) < \#_{2p}(r(f(b,t')))$, which is a contradiction.

As x is central in $f(x, y)$, we may write

$$f(x, y) =_{cr} g(x, y) \triangleleft x \triangleright h(x, y)$$

for binary operators g and h. Because b is central in t, we find

$$t =_{cr} r\big(T_b(g(b,t')) \triangleleft b \triangleright F_b(h(b,t'))\big).$$

We proceed with a case distinction on the form that $T_b(g(b,t'))$ and $F_b(h(b,t'))$ may take. At least one of these is modulo cr not equal to T or F (otherwise, $f(b,t')$ could be replaced by $f'(b)$ for some unary function f' and this was excluded above).

(1) Suppose $T_b(g(b,t')) \notin_{cr} \{T, F\}$ and $F_b(h(b,t')) \notin_{cr} \{T, F\}$. We show that this is not possible: first notice that because b is not central in $T_b(g(b,t'))$, a different atom must be central in this term, and this atom must be a. For this to hold, a must be central in $T_b(t')$ and no atom different from a can be tested by the first requirement of $\phi_{a,b,c}$. So, after contraction of all further a's we find

$$T_b(t') =_{cr} P \triangleleft a \triangleright Q$$

with $P, Q \in \{T, F\}$, and similarly $F_b(t') =_{cr} P' \triangleleft c \triangleright Q'$ with $P', Q' \in \{T, F\}$.
If $P \not\equiv Q$ and $P' \not\equiv Q'$, then t' is a term that satisfies $\phi_{a,b,c}$, but t' is a term with lower $\#_{2p}$-value than $r(g(b,t') \triangleleft b \triangleright h(b,t'))$, which is a contradiction.
Assume $P \equiv Q$ (the case $P' \equiv Q'$ is symmetric).
Now $t =_{cr} r\big(T_b(g(b,t')) \triangleleft b \triangleright F_b(h(b,t'))\big)$, and no b's can occur in $T_b(g(b,t'))$, so

$$T_b(g(b,t')) \in_{cr} \{P \triangleleft a \triangleright Q,\ F \triangleleft (P \triangleleft a \triangleright Q) \triangleright T,\ (P \triangleleft a \triangleright Q) \circ T,\ (P \triangleleft a \triangleright Q) \circ F\}.$$

For $F_b(h(b,t'))$, a similar argument applies, which implies that (recall $P \equiv Q$)

$$T_b(g(b,t')) =_{cr} a \circ P \text{ and } F_b(h(b,t')) =_{cr} P' \triangleleft c \triangleright Q' \quad \text{with } P, P', Q' \in \{T, F\}.$$

Assume $P \equiv T$ (the case $P \equiv F$ is symmetric). So in this case

$$t =_{cr} r((a \circ T) \triangleleft b \triangleright (P' \triangleleft c \triangleright Q')),$$

and we distinguish two cases:
(i) $P' \equiv T$ or $Q' \equiv T$. Now the reply to a in $a \circ T$ following a positive reply to the initial b has no effect, so this a must be followed by another central a. But this last a can also be reached after a b and a c, which contradicts $\phi_{a,b,c}$.
(ii) $P' \equiv Q' \equiv F$. Since property $\phi_{a,b,c}$ holds it must be the case that a is a central condition in $r(T)$ with the property that $T_a(r(T)) \neq_{cr} F_a(r(T))$, otherwise the initial b that stems from the substitution $x \mapsto (a \circ T) \triangleleft b \triangleright (c \circ F)$ in $r(x)$ is upon reply T immediately followed by $a \circ T$ and each occurrence of this a is not able to yield both T and F, contradicting $\phi_{a,b,c}$. (And also because this substitution yields no further occurrences of b upon reply T.)

Similarly, c is a central condition in $r(F)$ with the property that $T_c(r(F)) \neq_{cr} F_c(r(F))$. We find that $r(b)$ also satisfies $\phi_{a,b,c}$. Now observe that $r(b)$ is a term with lower $\#_{2p}$-value than $r(f(b,t'))$, which is a contradiction.

(2) We are left with four cases: either a is central in $T_b(g(b,t'))$ and $F_b(h(b,t')) \in_{cr} \{T, F\}$, or c is central in $F_b(h(b,t'))$ and $T_b(g(b,t')) \in_{cr} \{T, F\}$. These cases are symmetric and it suffices to consider only the first one, the others can be dealt with similarly.

So assume a is central in $T_b(g(b,t'))$ and $F_b(h(b,t')) =_{cr} T$. This implies $T_b(g(b,t')) =_{cr} P \triangleleft a \triangleright Q$ for some P, Q, and after contraction of all a's in P and Q,

$$T_b(g(b,t')) =_{cr} P' \triangleleft a \triangleright Q' \quad \text{for some } P', Q' \in \{T, F\}.$$

We find

$$t =_{cr} r((P' \triangleleft a \triangleright Q') \triangleleft b \triangleright T),$$

and we distinguish two cases:
(i) $P' \equiv T$ or $Q' \equiv T$. Now c can be reached after a negative reply to b according to $\phi_{a,b,c}$, but this c can also be reached after a positive reply to b and the appropriate reply to a, which contradicts $\phi_{a,b,c}$.
(ii) $P' \equiv Q' \equiv F$. Since property $\phi_{a,b,c}$ holds it must be the case that a is a central condition in $r(F)$ with the property that $T_a(r(F)) \neq_{cr} F_a(r(F))$, otherwise the initial b that stems from the substitution $x \mapsto (a \circ F) \triangleleft b \triangleright T$ in $r(x)$ is upon reply T immediately followed by $a \circ F$ and each occurrence of this a is not able to yield both T and F, contradicting $\phi_{a,b,c}$. (And also because this substitution yields no further occurrences of b upon reply T.)

Also, c is a central condition in $r(T)$ with the property that $T_c(r(T)) \neq_{cr} F_c(r(T))$. We find that $r(b)$ also satisfies $\phi_{a,b,c}$. Now observe that $r(b)$ is a term with lower $\#_{2p}$-value than $r(f(b,t'))$, which is a contradiction.

This concludes our proof. □

ACKNOWLEDGMENTS

We thank Chris Regenboog for discussions about completeness results. We thank three referees for their careful reviews of a prior version, which we think improved the presentation considerably.

REFERENCES

BAETEN, J. C. M. AND WEIJLAND, W. P. 1990. *Process Algebra*. Cambridge Tracts in Theoretical Computer Science, vol. 18. Cambridge University Press, Cambridge.

BELNAP, N. D. 1977. A useful four-valued logic. In *Modern Uses of Multiple-Valued Logic*, J. Dunn and G. Epstein, Eds. D. Reidel, Dordrecht, 8–37.

BERGSTRA, J. A., BETHKE, I., AND PONSE, A. 2007. Thread algebra and risk assessment services. In *Proceedings of the Logic Colloquium*. C. Dimitracopoulos, L. Newelski, and D. Normann, Eds. Cambridge Lecture Notes in Logic, vol. 28. 1–17.

BERGSTRA, J. A., BETHKE, I., AND RODENBURG, P. H. 1995. A propositional logic with 4 values: true, false, divergent and meaningless. *J. Appl. Non-Classi. Logics 5,* 2, 199–218.

BERGSTRA, J. A. AND KLOP, J. W. 1984. Process algebra for synchronous communication. *Inf. Cont. 60,* 1–3, 109–137.

BERGSTRA, J. A. AND MIDDELBURG, C. A. 2007. Thread algebra for strategic interleaving. *Form. Asp. Comput. 19,* 4, 445–474.

BERGSTRA, J. A. AND PONSE, A. 1998a. Bochvar-McCarthy logic and process algebra. *Notre Dame J. Form. Logic 39,* 4, 464–484.

BERGSTRA, J. A. AND PONSE, A. 1998b. Kleene's three-valued logic and process algebra. *Inf. Proc. Lett. 67,* 2, 95–103.

BERGSTRA, J. A. AND PONSE, A. 2005. A bypass of Cohen's impossibility result. In *Proceedings of the European Grid Conference (EGC)*. Lecture Notes in Computer Science, vol. 3470. Springer-Verlag, Berlin, 1097–1106.

BERGSTRA, J. A. AND PONSE, A. 2007. Execution architectures for program algebra. *J. Appl. Logic 5,* 1, 170–192.

BLOOM, S. L. AND TINDELL, R. 1983. Varieties of "if-then-else". *SIAM J. Comput. 12,* 4, 677–707.

COHEN, F. 1987. Computer viruses – theory and experiments. *Comput. Sec. 6*, 22–35.

COHEN, F. 2001. Reply to 'Comment on "A Framework for Modelling Trojans and Computer Virus Infection."' *Comput. J. 44,* 4, 326–327.

FOKKINK, W. J. 2000. *Introduction to Process Algebra*. Texts in Theoretical Computer Science, An EATCS Series. Springer-Verlag, Berlin.

HÄHNLE, R. 2005. Many-valued logic, partiality, and abstraction in formal specification languages. *Logic J. IGPL 13,* 4, 415–433.

HAYES, I. J., JIFENG, H., HOARE, C. A. R., MORGAN, C. C., ROSCOE, A. W., SANDERS, J. W., SORENSEN, I. H., SPIVEY, J. M., AND SUFRIN, B. A. 1987. Laws of programming. *Comm. ACM 3,* 8, 672–686.

HOARE, C. A. R. 1985a. *Communicating Sequential Processes*. Prentice-Hall, Englewood Cliffs.

HOARE, C. A. R. 1985b. A couple of novelties in the propositional calculus. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik 31,* 2, 173–178.

KONIKOWSKA, B. 1996. Recursive functions of symbolic expressions and their computation by machine. *Fund. Inf. 26,* 2, 167–203.

MCCARTHY, J. 1963. A basis for a mathematical theory of computation. In *Computer Programming and Formal Systems*, P. Braffort and D. Hirshberg, Eds. North-Holland, Amsterdam, 33–70.

MITTELSTAEDT, P. 2004. Quantum logic and decoherence. *Int. J. Theoret. Phys. 43,* 6, 1343–1354.

PONSE, A. AND VAN DER ZWAAG, M. B. 2007. Belnap's logic and conditional composition. *Theoret. Comput. Sci. 388,* 1–3, 319–336.

PONSE, A. AND VAN DER ZWAAG, M. B. 2008. Risk assessment for one-counter threads. *Theory Computi. Syst. 43*, 563–582.

REGENBOOG, B. C. 2010. Reactive valuations. MS thesis, University of Amsterdam (arXiv:1101.3132vl[cs.LO].)

REHDER, W. 1980. Quantum logic of sequential events and their objectivistic probabilities. *Int. J. Theoret. Physics 19,* 3, 221–237.

VU, T. D. 2008. Denotational semantics for thread algebra. *J. Logic Alg. Prog. 74*, 94–111.

Received February 2009; revised April 2010; accepted July 2010