

A Calculus for Four-Valued Sequential Logic

Jan A. Bergstra^{a,*}, Jaco van de Pol^{b,*}

^a*University of Amsterdam, Programming Research Group
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands*

^b*University of Twente, Formal Methods and Tools
P.O.-box 217, 7500 AE Enschede, The Netherlands*

Abstract

We present a complete axiomatisation for four-valued sequential logic. It consists of nine axioms, from which all valid laws can be derived by equational reasoning. These nine axioms are independent of each other.

Keywords: four valued sequential logic; complete axiomatisation; finite basis.

1. Introduction

This paper presents a complete algebraic axiomatisation for four-valued sequential logic. ‘Sequential’ means that arguments are evaluated from left to right, until an answer can be obtained. This models the ‘short-cutting’ semantics of connectives in programming languages.

Related work. Three-valued sequential logic is due to McCarthy [10]. In [4] four truth-values are introduced: *true*, *false*, *mistake* and *divergent*. The purpose of introducing these four values is to distinguish between computation errors, both in specification and programming. Several four-valued logics arise by restricting the set of connectives. In the nomenclature of [4], four-valued sequential logic is characterised as $\Sigma_4(\neg, \wedge, \vee)$. An axiomatisation of this system has not been published before.

In [5] it is examined whether four-valued sequential logic can serve as a basis for data type specifications. That application motivates and justifies the meta-mathematical study of four-valued logics. Our complete axiomatisation can also itself be viewed as an ω -complete data type specification, see [8], which is quite different in nature than a Gentzen-style deduction system as in [2]. We refer to [4] for an introduction to three- and four-valued logic and also for further references.

*The main part of this research was carried out when both authors were affiliated at Utrecht University, Department of Philosophy

Email addresses: J.A.Bergstra@uva.nl (Jan A. Bergstra),
J.C.vandePol@ewi.utwente.nl (Jaco van de Pol)

In [7] a complete axiomatisation is given for McCarthy’s system. Completeness is obtained by characterizing all algebras satisfying the axioms. The completeness proof for the axiomatisation of the four-valued system that we give is quite different. Our proof yields a systematic method to prove each valid formula from the axioms.

In [12] a complete proof system for full four-valued logic is provided, consisting of 25 equations. It is based on the functionally complete set of connectives $\overset{\circ}{\vee}$, \wedge and \downarrow , with \neg as a complex defined operation. Our result concerns a simpler axiomatisation for a simpler logic, and consequently has a simpler completeness proof as well. It is unlikely that our completeness result follows from [12].

Belnap [3] introduced a four-valued logic with a different purpose; we shortly discuss it in Section 4. A functionally complete version has been studied, with eight sublogics [2, Theorem 14], all including a non-monotonic form of implication. Our sequential 4-valued logic has only monotonic operators and is strictly less expressive than all eight, according to the conditions of that theorem.

Contribution. In this paper, we provide a complete finite equational axiomatisation of a sequential logic with four values, solving a question posed in [4]. Moreover, we show that its nine axioms are independent, so in some sense our axiomatisation is minimal. This finally solves an open question posed in [6]. There an additional axiom of left-distributivity was provided, but its independence could not be established, nor its derivability from the other axioms. The latter fact has now been established by the use of an automated theorem prover. This finally finishes off the result from 1996, and justifies (at least according to the second author) publication in this Festschrift.

2. Four-valued Sequential Logic

Following [4] we extend the usual truth values **t** and **f** (for *true* and *false*) with two other constants **d** and **m**, modeling a *diverging* computation and an error-situation (i.e. a *mistake* has been made and there is no point in going on). The difference between **d** and **m** can be illustrated by the following equations for conjunction (see Table 1 for the complete definition).

$$\mathbf{f} \wedge \mathbf{d} = \mathbf{d} \wedge \mathbf{f} = \mathbf{f} \quad \text{but} \quad \mathbf{f} \wedge \mathbf{m} = \mathbf{m} \wedge \mathbf{f} = \mathbf{m} .$$

\wedge	m	t	f	d	\vee	m	t	f	d	\downarrow	
m	m	m	m	m	m	m	m	m	m	m	f
t	m	t	f	d	t	m	t	t	t	t	t
f	m	f	f	f	f	m	t	f	d	f	t
d	m	d	f	d	d	m	t	d	d	d	f

Table 1: Parallel conjunction, parallel disjunction and definedness.

The first equation shows that a divergent computation may be circumvented, because the final result will be \mathbf{f} in any case. In the second case, a mistake has been made, and this has to be reported. Conjunction behaves strict w.r.t. \mathbf{m} , but non-strict w.r.t. \mathbf{d} .

As the first equation reveals, conjunction needs a parallel computation. If one of the arguments can be evaluated to false, a diverging computation in the other argument must be avoided. The definedness operator (\downarrow) is not computable at all (in a plausible model of computation). By definition, $\downarrow X = \mathbf{f}$ when $X = \mathbf{d}$ or $X = \mathbf{m}$ and $\downarrow X = \mathbf{t}$ otherwise. Intuitively, we cannot know whether an ongoing computation of X will diverge, or result in an answer eventually.

It is also possible to study sequential connectives. Here the computation starts on the left and terminates as soon as an answer can be given. We write \wp for the *left sequential conjunction* (also known as ‘conditional and’). See Table 2 for the definition. Here a mistake is not always reported. Some typical equations are:

$$\mathbf{f} \wp \mathbf{d} = \mathbf{f} \qquad \mathbf{d} \wp \mathbf{f} = \mathbf{d} \qquad \mathbf{f} \wp \mathbf{m} = \mathbf{f} \qquad \mathbf{m} \wp \mathbf{f} = \mathbf{m} .$$

The system that extends the truth values $\{\mathbf{t}, \mathbf{f}, \mathbf{m}, \mathbf{d}\}$ with the connectives o_1, \dots, o_n is denoted by $\Sigma_4(o_1, \dots, o_n)$. In [4] it is proved that the system $\Sigma_4(\neg, \wedge, \downarrow, \wp)$ is truth-functionally complete. This means that every four-valued function can be expressed in terms of the truth values, combined with negation, conjunction, definedness and the left sequential conjunction.

In this paper we focus on $\Sigma_4(\neg, \wp, \wp)$, the system of strongly sequential truth functions. Although \wp can be defined from \wp and \neg in the usual way, we incorporate it in the language. Table 2 contains the truth tables for the sequential connectives.

In this system, several classical principles are lacking. We mention commutativity of \wp and right-distributivity. In Table 3 we list 9 laws that hold in $\Sigma_4(\neg, \wp, \wp)$. These laws are self explaining, except the last which expresses a valid variant of right-distributivity. We claim that all other valid laws can be derived from these 9. So Table 3 gives a complete axiomatisation of $\Sigma_4(\neg, \wp, \wp)$. This claim is proved in Section 3. In Section 4 we show that each of the laws (1)–(9) are independent of the other laws.

3. A Complete Axiomatisation of $\Sigma_4(\neg, \wp, \wp)$

We write P, Q for arbitrary open terms over $\Sigma_4(\neg, \wp, \wp)$. X and Y are arbitrary variables, while b, c range over the four constants. With $P[X := c]$, we will denote the term P , with all occurrences of X replaced by c . We write $\models P = Q$ if $P = Q$ holds in $\Sigma_4(\neg, \wp, \wp)$. With $\vdash P = Q$, we denote that $P = Q$ is derivable with equational logic, using laws (1)–(9) of Table 3.

\neg		δ	m	t	f	d	\wp	m	t	f	d
m	m	m	m	m	m	m	m	m	m	m	m
t	f	t	m	t	f	d	t	t	t	t	t
f	t	f	f	f	f	f	f	m	t	f	d
d	d	d	d	d	d	d	d	d	d	d	d

Table 2: The connectives of $\Sigma_4(\neg, \delta, \wp)$.

(1)	$\neg \mathbf{d} = \mathbf{d}$
(2)	$\neg \mathbf{m} = \mathbf{m}$
(3)	$\neg \mathbf{t} = \mathbf{f}$
(4)	$\neg \neg X = X$
(5)	$\mathbf{t} \delta X = X$
(6)	$\mathbf{f} \delta X = \mathbf{f}$
(7)	$X \wp Y = \neg(\neg X \delta \neg Y)$
(8)	$(X \delta Y) \delta Z = X \delta (Y \delta Z)$
(9)	$(X \wp Y) \delta Z = (\neg X \delta (Y \delta Z)) \wp (X \delta Z)$

Table 3: Nine axioms of $\Sigma_4(\neg, \delta, \wp)$.

		<i>Derived from:</i>
(10)	$X \delta \mathbf{t} = X$	(3,5,5d,6d,9d)
(11)	$X \delta Y = (\neg X \wp Y) \delta X$	(9d,10d)
(12)	$X \delta \mathbf{f} = \neg X \delta X$	(10d,11)
(13)	$X \delta \neg X = \neg X \delta X$	(4,5d,6,9,9d,11,12)
(14)	$X \delta X = X$	(4,10,11,12d,13d)
(15)	$X \delta Y = X \delta (Y \delta X)$	(8,11,14)
(16)	$X \delta (Y \wp Z) = (X \delta Y) \wp (X \delta Z)$	(4,8,8d,9,11,11d,15)
(17)	$X \delta (Y \delta Z) = (X \delta Y) \delta (X \delta Z)$	(8,15)
(18)	$\mathbf{d} \delta X = \mathbf{d}$	(1,6,8,12,13,14)
(19)	$\mathbf{m} \delta X = \mathbf{m}$	(2,6,8,12,13,14)

Table 4: Ten more laws of $\Sigma_4(\neg, \delta, \wp)$.

Proposition 1. For all P and Q , if $\vdash P = Q$ then $\vDash P = Q$.

PROOF. Laws (1)–(9) can be checked straightforwardly. This gives the required result. \square

The *dual* of a term is obtained by interchanging all occurrences of \mathbf{t} with \mathbf{f} , and $\delta\wedge$ with $\delta\vee$.

Lemma 2. For all P and Q , if $\vdash P = Q$ then also $\vdash P^{dual} = Q^{dual}$.

PROOF. By (4) it suffices to prove $\neg\neg P^{dual} = \neg\neg Q^{dual}$. Using (1)–(4) and (7), the second \neg can be pushed inside step by step. Eventually, an instance of $\neg P = \neg Q$ is obtained (with all variables X replaced by $\neg X$), which can be derived since by assumption $\vdash P = Q$. \square

In the sequel, we will denote the use of the dual of a derived law by the postfix d . E.g. (5d) is the dual of (5): $\mathbf{f}\delta\vee X = X$.

Lemma 3. Laws (10)–(19) from Table 4 are derivable from (1)–(9):

PROOF. **10.** $X \stackrel{5d,5}{=} (\mathbf{t}\delta\wedge\mathbf{f})\delta\vee X \stackrel{9d}{=} (\neg\mathbf{t}\delta\vee\mathbf{f}\delta\vee X)\delta\wedge(\mathbf{t}\delta\vee X) \stackrel{6d,3,5d}{=} X\delta\wedge\mathbf{t}$.

11. $X\delta\wedge Y \stackrel{10d}{=} (X\delta\wedge Y)\delta\vee\mathbf{f} \stackrel{9d}{=} (\neg X\delta\vee Y\delta\vee\mathbf{f})\delta\wedge(X\delta\vee\mathbf{f}) \stackrel{10d}{=} (\neg X\delta\vee Y)\delta\wedge X$.

12. $X\delta\wedge\mathbf{f} \stackrel{11}{=} (\neg X\delta\vee\mathbf{f})\delta\wedge X \stackrel{10d}{=} \neg X\delta\wedge X$.

13. First, we prove the auxiliary identity (*): $(X\delta\wedge\mathbf{f})\delta\vee(X\delta\wedge\mathbf{f}) = X\delta\wedge\mathbf{f}$.

$$\begin{aligned} & (X\delta\wedge\mathbf{f})\delta\vee(X\delta\wedge\mathbf{f}) \\ \stackrel{4,6,12}{=} & (\neg\neg X\delta\wedge(\mathbf{f}\delta\wedge X))\delta\vee(\neg X\delta\wedge X) \\ \stackrel{9}{=} & (\neg X\delta\vee\mathbf{f})\delta\wedge X \\ \stackrel{11}{=} & X\delta\wedge\mathbf{f} \end{aligned}$$

Next we prove 13:

$$\begin{aligned} & X\delta\wedge\neg X \\ \stackrel{11}{=} & (\neg X\delta\vee\neg X)\delta\wedge X \\ \stackrel{9}{=} & (\neg\neg X\delta\wedge(\neg X\delta\wedge X))\delta\vee(\neg X\delta\wedge X) \\ \stackrel{4,12}{=} & (X\delta\wedge(X\delta\wedge\mathbf{f}))\delta\vee(X\delta\wedge\mathbf{f}) \\ \stackrel{9}{=} & (\neg X\delta\vee((X\delta\wedge\mathbf{f})\delta\vee(X\delta\wedge\mathbf{f})))\delta\wedge(X\delta\vee(X\delta\wedge\mathbf{f})) \\ \stackrel{(*)}{=} & (\neg X\delta\vee(X\delta\wedge\mathbf{f}))\delta\wedge(X\delta\vee(X\delta\wedge\mathbf{f})) \\ \stackrel{5d}{=} & (\neg X\delta\vee(\mathbf{f}\delta\vee(X\delta\wedge\mathbf{f})))\delta\wedge(X\delta\vee(X\delta\wedge\mathbf{f})) \\ \stackrel{9d}{=} & (X\delta\wedge\mathbf{f})\delta\vee(X\delta\wedge\mathbf{f}) \\ \stackrel{(*)}{=} & X\delta\wedge\mathbf{f} \\ \stackrel{12}{=} & \neg X\delta\wedge X \end{aligned}$$

14. $X \triangleleft X$
 $\stackrel{11}{\equiv} (\neg X \circlearrowleft X) \triangleleft X$
 $\stackrel{13d}{\equiv} (X \circlearrowleft \neg X) \triangleleft X$
 $\stackrel{4,12d}{\equiv} (\neg X \circlearrowleft \mathbf{t}) \triangleleft X$
 $\stackrel{11}{\equiv} X \triangleleft \mathbf{t}$
 $\stackrel{10}{\equiv} X$
15. $X \triangleleft Y$
 $\stackrel{11}{\equiv} (\neg X \circlearrowleft Y) \triangleleft X$
 $\stackrel{14}{\equiv} (\neg X \circlearrowleft Y) \triangleleft (X \triangleleft X)$
 $\stackrel{8}{\equiv} ((\neg X \circlearrowleft Y) \triangleleft X) \triangleleft X$
 $\stackrel{11}{\equiv} (X \triangleleft Y) \triangleleft X$
 $\stackrel{8}{\equiv} X \triangleleft (Y \triangleleft X)$
16. $X \triangleleft (Y \circlearrowleft Z)$
 $\stackrel{11}{\equiv} (\neg X \circlearrowleft (Y \circlearrowleft Z)) \triangleleft X$
 $\stackrel{8d}{\equiv} ((\neg X \circlearrowleft Y) \circlearrowleft Z) \triangleleft X$
 $\stackrel{4,11d}{\equiv} (((X \triangleleft Y) \circlearrowleft \neg X) \circlearrowleft Z) \triangleleft X$
 $\stackrel{8d}{\equiv} ((X \triangleleft Y) \circlearrowleft (\neg X \circlearrowleft Z)) \triangleleft X$
 $\stackrel{9}{\equiv} (\neg(X \triangleleft Y) \triangleleft ((\neg X \circlearrowleft Z) \triangleleft X)) \circlearrowleft ((X \triangleleft Y) \triangleleft X)$
 $\stackrel{11}{\equiv} (\neg(X \triangleleft Y) \triangleleft (X \triangleleft Z)) \circlearrowleft ((X \triangleleft Y) \triangleleft X)$
 $\stackrel{8,15}{\equiv} (\neg(X \triangleleft Y) \triangleleft (X \triangleleft Z)) \circlearrowleft (X \triangleleft Y)$
 $\stackrel{11d}{\equiv} (X \triangleleft Y) \circlearrowleft (X \triangleleft Z)$
17. $X \triangleleft (Y \triangleleft Z) \stackrel{8}{\equiv} (X \triangleleft Y) \triangleleft Z \stackrel{15}{\equiv} (X \triangleleft (Y \triangleleft X)) \triangleleft Z \stackrel{8}{\equiv} (X \triangleleft Y) \triangleleft (X \triangleleft Z)$.
18. If $Z = \neg Z$ then $Z = Z \triangleleft \mathbf{f}$ (*), for $Z \stackrel{14}{\equiv} Z \triangleleft Z = Z \triangleleft \neg Z \stackrel{13,12}{\equiv} Z \triangleleft \mathbf{f}$; hence $Z \triangleleft X \stackrel{(*)}{\equiv} (Z \triangleleft \mathbf{f}) \triangleleft X \stackrel{8}{\equiv} Z \triangleleft (\mathbf{f} \triangleleft X) \stackrel{6}{\equiv} Z \triangleleft \mathbf{f} \stackrel{(*)}{\equiv} Z$. Now using (1), the required result follows.

19. Similar to (18), but now using (2). □

Lemma 4. *Every closed term is provably equal to \mathbf{t} , \mathbf{f} , \mathbf{d} or \mathbf{m} .*

PROOF. This is proved by term induction. In case of a negation, (1), (2), (3) and (3d) is used. In case of conjunction we use (5), (6), (18) and (19). Disjunction is the dual of conjunction. □

Lemma 5. *Every term P is either provably equal to a closed term, or it is provably equal to $(X \circlearrowleft P_1) \triangleleft P_2$, for some variable X and terms P_1 and P_2 . Moreover, X and the variables occurring in P_1 and P_2 also occur in P .*

PROOF. The lemma is proved with induction on P :

Case c : Constants are clearly closed.

Case X : By (10d) and (10), $\vdash X = (X \overset{\circ}{\vee} \mathbf{f}) \triangleleft \mathbf{t}$.

Case $\neg P$: If P is provably closed, then $\neg P$ is provably closed too. Otherwise, we obtain:

$$\begin{aligned}
& \neg P \\
\stackrel{\text{IH}}{=} & \neg((X \overset{\circ}{\vee} P_1) \triangleleft P_2) \\
\stackrel{9}{=} & \neg((\neg X \triangleleft (P_1 \triangleleft P_2)) \overset{\circ}{\vee} (X \triangleleft P_2)) \\
\stackrel{7}{=} & \neg(\neg(\neg X \triangleleft (P_1 \triangleleft P_2)) \triangleleft \neg(X \triangleleft P_2)) \\
\stackrel{4}{=} & \neg(\neg X \triangleleft \neg(P_1 \triangleleft P_2)) \triangleleft \neg(X \triangleleft P_2) \\
\stackrel{7}{=} & (X \overset{\circ}{\vee} \neg(P_1 \triangleleft P_2)) \triangleleft \neg(X \triangleleft P_2) ,
\end{aligned}$$

which is of the required format.

Case $P \triangleleft Q$: If P is provably closed, then by Lemma 4, it is provably equal to $\mathbf{t}, \mathbf{f}, \mathbf{d}$ or \mathbf{m} . Then $P \triangleleft Q$ is either provably closed, or provably equal to Q . In the latter case the induction hypothesis for Q yields the required format.

If P is not provably closed, then by the induction hypothesis for P , we obtain P_1 and P_2 such that $\vdash P \triangleleft Q = ((X \overset{\circ}{\vee} P_1) \triangleleft P_2) \triangleleft Q$. Using (8) this can be brought in the required form.

Case $P \overset{\circ}{\vee} Q$: The case that P is provably closed is similar to \triangleleft . Otherwise, we find P_1 and P_2 by induction hypothesis for P , such that

$$\begin{aligned}
& P \overset{\circ}{\vee} Q \\
\stackrel{\text{IH}}{=} & ((X \overset{\circ}{\vee} P_1) \triangleleft P_2) \overset{\circ}{\vee} Q \\
\stackrel{9}{=} & ((\neg X \triangleleft (P_1 \triangleleft P_2)) \overset{\circ}{\vee} (X \triangleleft P_2)) \overset{\circ}{\vee} Q \\
\stackrel{\text{8d}}{=} & (\neg X \triangleleft (P_1 \triangleleft P_2)) \overset{\circ}{\vee} ((X \triangleleft P_2) \overset{\circ}{\vee} Q) \\
\stackrel{\text{9d}}{=} & (\neg \neg X \overset{\circ}{\vee} ((P_1 \triangleleft P_2) \overset{\circ}{\vee} ((X \triangleleft P_2) \overset{\circ}{\vee} Q))) \triangleleft (\neg X \overset{\circ}{\vee} ((X \triangleleft P_2) \overset{\circ}{\vee} Q)) \\
\stackrel{4}{=} & (X \overset{\circ}{\vee} ((P_1 \triangleleft P_2) \overset{\circ}{\vee} ((X \triangleleft P_2) \overset{\circ}{\vee} Q))) \triangleleft (\neg X \overset{\circ}{\vee} ((X \triangleleft P_2) \overset{\circ}{\vee} Q)) ,
\end{aligned}$$

which is of the required form. \square

Lemma 6. For any term P and variable X , we have

$$(a) \vdash X \triangleleft P = X \triangleleft P[X := \mathbf{t}]$$

$$(b) \vdash \neg X \triangleleft P = \neg X \triangleleft P[X := \mathbf{f}]$$

PROOF. Without loss of generality, we assume that P is built from constants, variables and negated variables, using the connectives \triangleleft and $\overset{\circ}{\vee}$. Using (1)–(4) and (7) (including their duals), we can write each term in such a form. The lemma is then proved with induction on P :

Cases c , Y or $\neg Y$ with c a constant and $Y \neq X$, are trivial.

Case X : use (10,14) for (a); use (13,4,12) for (b).

Case $\neg X$: use (13,12,3) for (a); use (14,10,3d) to obtain (b).

Case $P \delta Q$: Use (17) and the induction hypothesis for P and Q .

Case $P \wp Q$: Use (16) and the induction hypothesis for P and Q . \square

Theorem 7. *Axioms (1)–(9) form a complete axiomatisation for $\Sigma_4(\neg, \delta, \wp)$.*

PROOF. Assume $\vDash P = Q$. We prove $\vdash P = Q$ by induction on the number of different variables occurring in this equation. By Lemma 5, we have that either

(a) P is provably closed; or

(b) $\vdash P = (X \wp P_1) \delta P_2$.

Similarly, we obtain that either

(c) Q is provably closed; or

(d) $\vdash Q = (Y \wp Q_1) \delta Q_2$.

By Lemma 4, each provably closed term is provably equal to \mathbf{t} , \mathbf{f} , \mathbf{d} or \mathbf{m} . We first prove that case (a) and (d) cannot occur both, for assume (a) and (d). Then $\vdash P = c$, where c is one of the constants. By soundness, $\vDash P = c$ and $\vDash Q = (Y \wp Q_1) \delta Q_2$, hence also $\vDash c = (Y \wp Q_1) \delta Q_2$. Now taking $Y = \mathbf{d}$ and $Y = \mathbf{m}$, respectively, we get $\vDash \mathbf{d} = \mathbf{m}$, quod non. Similarly, (b) and (c) cannot occur both. Two cases remain:

- (a) and (c) hold (this includes the base of the induction). In this case $\vdash P = b$ and $\vdash Q = c$, for some constants b and c . Using soundness and the assumption that $\vDash P = Q$, we obtain $b \equiv c$, so $\vdash P = Q$.

- (b) and (d) hold. In this case $X \equiv Y$, for otherwise we could substitute \mathbf{d} for X and \mathbf{m} for Y , implying (via 18,18d,19,19d and soundness) $\vDash \mathbf{m} = \mathbf{d}$.

Define $P'_1 \equiv (P_1 \delta P_2)[X := \mathbf{f}]$, $P'_2 \equiv (P_2[X := \mathbf{t}])$. Then using (9), we have $\vdash P = (\neg X \delta P_1 \delta P_2) \wp (X \delta P_2)$. By Lemma 6 we have $\vdash P = (\neg X \delta P'_1) \wp (X \delta P'_2)$. In a similar way we can find Q'_1 and Q'_2 that do not contain X such that $\vdash Q = (\neg X \delta Q'_1) \wp (X \delta Q'_2)$.

Using $\vDash P = Q$ and soundness and taking $X = \mathbf{t}$, we find $\vDash P'_2 = Q'_2$. Taking $X = \mathbf{f}$, we find $\vDash P'_1 = Q'_1$. Now by induction hypothesis, $\vdash P'_1 = Q'_1$ and $\vdash P'_2 = Q'_2$. By equational logic, we find $\vdash P = Q$. \square

4. Concluding Remarks

Extension. The existence of at least two error values is needed in the proof of Theorem 7 to make sure that $X \equiv Y$. If there is only one error value (i.e. McCarthy's logic [10]) then the following law becomes valid; note that the leftmost variable changes:

$$((X \wp Y) \wp (Y \wp X)) = ((Y \wp X) \wp (X \wp Y)) .$$

Our proof easily generalises to more than two error values. For a new error value, add an axiom $e = \neg e$. As in Lemma 3.18, we can then prove $e \wp X = e$; via (7) we obtain $e \wp X = e$. With these equations, Lemma 4 can be extended to the new situation. Then the proof of Theorem 7 remains valid.

Independence of axioms. Below we list the arguments that each of the laws (1)–(9) is independent of the other laws. In order to prove some law independent, we use semantic (1,2,7,8,9) and syntactic (3,4,5,6) arguments. The semantic argument provides a model for the remaining laws that refutes this one. The syntactic argument shows that the remaining laws obey some invariant, which prevents the derivation of this law.

1. Take as model the restriction of \neg, \wp and \wp to the carrier set $\{\mathbf{t}, \mathbf{f}, \mathbf{m}\}$ and interpret \mathbf{d} by \mathbf{t} . Then law (1) is false but laws (2)–(9) hold.
2. Similar to 1.
3. If a term contains \mathbf{f} then after application of a rule from (1,2,4–9), it still contains \mathbf{f} . So law (3) is not derivable from those laws.
4. Without law (4), a term in which no symbols from $\{\neg, \wp, \mathbf{f}, \mathbf{m}, \mathbf{d}\}$ occur cannot be proved equal to a term that contains some of these symbols.
5. With the laws other than (5), terms without constants cannot be proved equal to a term containing a constant.
6. Without law (6), a closed term cannot be proved equal to an open term.
7. In the following model laws (1)–(6) and (8)–(9) hold, but (7) fails: The carrier set is $\{\mathbf{t}, \mathbf{f}, \mathbf{m}, \mathbf{d}\}$. Interpret \neg as usual negation, \wp as the constant function \mathbf{f} and interpret $x \wp y$ as y whenever $x = \mathbf{t}$ and \mathbf{f} otherwise.
8. Consider the model $\Sigma_4(\neg, \wp, \wp)$, modified only by $\mathbf{d} \wp \mathbf{m} = \mathbf{d} \wp \mathbf{m} = \mathbf{m}$. In this model, (1)–(7) and (9) hold, but (8) doesn't hold:
 $(\mathbf{d} \wp \mathbf{f}) \wp \mathbf{m} = \mathbf{m} \neq \mathbf{d} = \mathbf{d} \wp (\mathbf{f} \wp \mathbf{m})$.
9. In the following model (1)–(8) hold but (9) fails: Take $\{\mathbf{t}, \mathbf{f}, \mathbf{d}\}$ as carrier set, with $\mathbf{m} = \mathbf{d}$. Interpret \wp and \wp as the restriction of \wedge and \vee , respectively (Table 1). This model is known as Kleene's three-valued logic [9].

$$(9b) \quad (X \wp Y) \wp Z = (X \wp Z) \wp (\neg X \wp (Y \wp Z))$$

Table 5: Alternative axiom for right-distributivity.

Use of tools. In [6], the proofs of law (13) and (16) were much simpler, based on the axiom of left-distributivity (here law 16). Actually, the current proofs were found with help of the automated theorem prover **Prover9**,¹ a successor of Otter [11], and its companion **Mace4** for finding finite counterexample models. Although this prover finds proofs by paramodulation within seconds, it still took several days to manually transform them to straight proofs in equational logic, and to introduce suitable intermediate lemmas.

Alternatives. The reader might wonder why in law (9) the disjuncts seem to be reordered. Indeed, the alternative rule (9b) from Table 5 is valid as well. However, replacing rule (9) by (9b) doesn't yield a complete axiomatisation. Apparently, some form of commutativity is needed as well. We claim that replacing (9) by (9b+11) or even by (9b+13) both yield a complete axiomatisation. This is readily checked by deriving law (9) from them. Eventually, the question which minimal set of axioms to choose as a basis is a matter of taste.

Belnap's four-valued logic. Belnap [3] introduced four-valued logic, with the purpose to reason about incomplete (*none*) and inconsistent (*both*) information from different sources. Linking **d** to *none* and **m** to *both*, our values are based on the same information lattice as his: $\mathbf{d} < \mathbf{f}, \mathbf{t} < \mathbf{m}$. Belnap's conjunction and disjunction are the meet and join in a single logical lattice $\mathbf{f} < \mathbf{d}, \mathbf{m} < \mathbf{t}$.

Our purpose is to study errors in specification and programming. We distinguish errors which may be avoided (e.g. divergence **d**) and errors that should be reported (e.g. mistakes **m**). Our four-valued logic is not based on a bilattice: As we want to be strict in **m**, our parallel conjunction can be viewed as the meet in the logical lattice $\mathbf{m} < \mathbf{f} < \mathbf{d} < \mathbf{t}$, while our parallel disjunction corresponds to the join in the different lattice $\mathbf{f} < \mathbf{d} < \mathbf{t} < \mathbf{m}$.

Concluding, we study *how we might think about computing* rather than *how computers should think*.

¹See <http://www.cs.unm.edu/~mccune/prover9/>

References

- [1] A.R. Anderson, N.D. Belnap, and J.M. Dunn, editors. *Entailment II, the logic of relevance and necessity*, volume II, chapter XII, §81. A useful four-valued logic: How computers should think, pages 506–541. Princeton University Press, 1992.
- [2] Ofer Arieli and Arnon Avron. The value of the four values. *Journal of Artificial Intelligence*, 102(1):97–141, 1998.
- [3] Nuel D. Belnap. How a computer should think. In G.Ryle, editor, *Contemporary aspects of Philosophy*, pages 30–55. Oriel Press, 1977. (cf. [1]).
- [4] Jan A. Bergstra, Inge Bethke, and Piet H. Rodenburg. A propositional logic with 4 values: true, false, divergent and meaningless. *Journal of Applied Non-Classical Logics*, 5(2):199–217, 1995.
- [5] Jan A. Bergstra and Alex Sellink. Sequential data algebra primitives. Utrecht University, Dept. of Philosophy, 1996.
- [6] Jan A. Bergstra and Jaco van de Pol. A calculus for sequential logic with 4 values. Technical Report Logic Group Preprint Series Nr. 160, Utrecht University, Dept. of Philosophy, 1996.
- [7] F. Guzman and C.S. Squier. The algebra of conditional logic. *Algebra Universalis*, 27:88–110, 1990.
- [8] Jan Heering. Partial evaluation and ω -completeness of algebraic specifications. *Theoretical Computer Science*, 43:149–167, 1986.
- [9] S.C. Kleene. On a notation for ordinal numbers. *Journal of Symbolic Logic*, 3:150–155, 1938.
- [10] John McCarthy. A basis for a mathematical theory of computation. In P. Braffort and D. Hirshberg, editors, *Computer Programming and Formal Systems*, pages 33–70. North-Holland, 1963.
- [11] William McCune and Larry Wos. Otter - the CADE-13 competition incarnations. *Journal of Automated Reasoning*, 18(2):211–220, 1997.
- [12] Piet H. Rodenburg. A complete system of four-valued logic. *Journal of Applied Non-Classical Logics*, 11(3-4):367–389, 2001.