Formal Justification of Underspecification for S5

Eric Aaron[1] and David Gries[2]
Computer Science, Cornell University
Ithaca, NY 14853

February 1997

Abstract. We formalize the notion of underspecification as a means of avoiding problems with partial functions in modal logic S5 and some semantically related logics. For these logics, underspecification respects validity, so incorporating it into their semantics leaves their classes of valid formulae unchanged.

1 Introduction

Gries and Schneider [4] suggest avoiding problems with partial operations and functions using *underspecification*. All operations and functions are assumed to be defined for all values of their operands, but the value assigned to an expression need not be uniquely specified in all cases. For example, consider division $/$, which has type $\mathbb{R} \times \mathbb{R} \to \mathbb{R}$. With underspecification, $5/0$ evaluates to a real number, but which one it is remains unspecified —and it is impossible to determine its value. Similarly, $5/0 = x/0$ is unspecified.

Thus, any type-correct operation or function is total, but it may be unspecified for some arguments.

Since $5/0$ is defined (but unspecified), $5/0 = 5/0$ evaluates to *true*. Therefore, some expressions can be evaluated even if their operands are unspecified. Underspecification can be handled in the logic by guarding each axiom and theorem (when necessary). For example, the law $y/y = 1$ holds iff $y \neq 0$, so the law is expressed as $y \neq 0 \ \Rightarrow\ y/y = 1$.

In [5], Gries and Schneider investigate propositional logic together with the everywhere operator $\Box$, where $\Box P$ means "P is true everywhere". They argue that modal logic S5 (see e.g. [6]) is a suitable such logic, where $\Box P$ is usually called "necessarily P". But they also show that S5 is incomplete with respect to the single model consisting of the set of all states and introduce an extension C of S5 that is complete. In this paper, we formalize the notion of underspecification in terms of the S5 and C models and show that these models respect validity.

We use the notation for quantification and rules for manipulating quantifications of [2].

[1] Supported by NSF grant GER-9454149.
[2] Supported by NSF grants CDA-9214957 and CCR-9503319.

Table 1: Table of abbreviations

$\alpha \wedge \beta$:	$\neg(\neg\alpha \vee \neg\beta)$	*true* :	$p \equiv p$
$\alpha \Rightarrow \beta$:	$\neg\alpha \vee \beta$	*false* :	$\neg$*true*
$\alpha \equiv \beta$:	$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$	$\diamond\alpha$:	$\neg\Box\neg\alpha$

2 Underspecification for S5

Let VP be a set of propositional variables. We use lower-case letters $p, q, r, \ldots$ for elements of VP. A *formula* of S5 has one of the following forms (p is any variable in VP, and metavariables α, β stand for formulas).

(1) $\quad p \qquad (\neg\alpha) \qquad (\alpha \vee \beta) \qquad (\Box\alpha)$

In addition, $(\alpha \wedge \beta)$, $(\alpha \Rightarrow \beta)$, $(\alpha \equiv \beta)$, $(\diamond\alpha)$, *true*, and *false* are abbreviations of certain formulas, as shown in Table 1. (Operator $\diamond$ is read as "possibly" or "somewhere".) Precedences eliminate the need for some parentheses; prefix operators $\neg$, $\Box$, and $\diamond$ bind tightest, then $\vee$ and $\wedge$, then $\Rightarrow$, and finally $\equiv$.

A formula of S5 that contains neither $\Box$ nor $\diamond$ is called a *propositional formula.*

(2) **Definition.** An *S5 model*[3] is a pair (V, W) where W is a non-empty set of *worlds* and *valuation function* V on W satisfies $V.w.p = \mathbf{t}$ or $V.w.p = \mathbf{f}$ for all worlds w and variables p.

Evaluation $ev.V.w.\alpha$ of a formula α in a world w in an S5 model (V, W) is defined as follows. In this definition and throughout the paper, when not explicitly given, the range of dummy w is the set W of worlds.

(3)
$$\begin{array}{llcl}
ev.V.w.p & = & V.w.p & \\
ev.V.w.\neg\alpha & = & \mathbf{t} & \text{iff } ev.V.w.\alpha = \mathbf{f} \\
 & = & \mathbf{f} & \text{iff } ev.V.w.\alpha = \mathbf{t} \\
ev.V.w.(\alpha \vee \beta) & = & \mathbf{t} & \text{iff } ev.V.w.\alpha = \mathbf{t} \text{ or } ev.V.w.\beta = \mathbf{t} \\
 & = & \mathbf{f} & \text{iff } ev.V.w.\alpha = \mathbf{f} \text{ and } ev.V.w.\beta = \mathbf{f} \\
ev.V.w.\Box\alpha & = & \mathbf{t} & \text{iff } (\forall w \mid : ev.V.w.\alpha = \mathbf{t}) \\
 & = & \mathbf{f} & \text{iff } (\exists w \mid : ev.V.w.\alpha = \mathbf{f})
\end{array}$$

In the last three cases of the definition, the two subcases are mutually exclusive.

A formula ϕ is *S5-valid*, denoted by $\models_{S5} \phi$, iff $ev.V.w.\phi = \mathbf{t}$ for all S5 models (V, W) and worlds w in W.

[3] Some treatments of S5 define a model to be triple (W, R, V), where W and V are as defined above and *accessibility relation* R on W is an equivalence relation, which is used in the semantics of operator $\Box$ and the description of S5-validity. Hughes and Cresswell [6] argue that the two definitions are equivalent.

In S5 models, evaluation of any expression results in a truth value. We now define S5U models, which permit underspecification. We use the three symbols $\mathbf{t}$, $\mathbf{f}$, and $\mathbf{u}$, where $\mathbf{u}$ denotes "not specified" and $\mathbf{t}$, $\mathbf{f}$ denote truth values in the expected way. Alternatively, think of $\mathbf{u}$ as the set $\{\mathbf{t},\mathbf{f}\}$. If an expression evaluates to $\{\mathbf{t},\mathbf{f}\}$, then its value is either $\mathbf{t}$ or $\mathbf{f}$, but it is not known which it is. Note that $\mathbf{u}$ is *not* a symbol of the language, and it may not appear in expressions.

(4) **Definition.** An *S5U model* is a pair (V,W) where W is a non-empty set of worlds and valuation function V on W satisfies $V.w.p=\mathbf{t}$, $V.w.p=\mathbf{f}$, or $V.w.p=\mathbf{u}$ for all worlds w and variables p .

We define some properties of models and valuations. Valuation V on W is *full* iff for all w in W and all propositional variables p , $V.w.p=\mathbf{t}$ or $V.w.p=\mathbf{f}$. For a given W , let FV denote the set of all full valuations on W . We say that a model (V,W) is full iff V is full on W . Hence, the set of S5 models equals the set of full S5U models.

We can extend an unfull model (V,W) and its unfull valuation V to ones that are full.

(5) **Definition.** A *full extension of a model* (V,W) is a model (V',W) where V' satisfies the following:

1. V' is full on W : $V \in FV$,
2. $V.w.p=\mathbf{t} \Rightarrow V'.w.p=\mathbf{t}$ and $V.w.p=\mathbf{f} \Rightarrow V'.w.p=\mathbf{f}$ for all w in W and variables p .

We call any such V' a *full extension of* V *on* W , and we denote the set of all such V' by $FX.V$. This enables us to concisely formalize properties such as the following three.

A full-extension valuation is full:

(6) $FX.V \subseteq FV$

A valuation is full exactly when it is a full extension of itself:

(7) $V \in FV \equiv V \in FX.V$

A full valuation has exactly one full extension, itself:

(8) $V \in FV \Rightarrow (V'=V \equiv V' \in FX.V)$

We now define evaluation $evu.V.w.\phi$ in an S5U model (V,W) for a world $w \in W$ and a formula ϕ . This definition is supposed to mean that $evu.V.w.\phi=\mathbf{t}$ iff $evu.V'.w.\phi=\mathbf{t}$ in all full extensions V' of V (and similarly for $\mathbf{f}$). However, it is easier to define evu

in terms of ev and then *prove* that the definition of evu has the desired property. In the definition of evu, note that ev is applied only to full valuations.

(9)
$$\begin{array}{lcll} evu.V.w.\phi & = & \mathbf{t} & \text{iff } (\forall V' \mid V' \in FX.V : ev.V'.w.\phi = \mathbf{t}) \\ & = & \mathbf{f} & \text{iff } (\forall V' \mid V' \in FX.V : ev.V'.w.\phi = \mathbf{f}) \\ & = & \mathbf{u} & \text{otherwise} \end{array}$$

A formula ϕ is *S5U-valid*, denoted by $\models_{\mathrm{S5U}} \phi$, iff $evu.V.w.\phi = \mathbf{t}$ for all S5U models (V, W) and worlds w in W.

We now prove three simple lemmas about ev and evu. The first says that ev and evu agree on full models. The second proves the desired property of evu (stated before the definition of evu). And the third is a technical lemma, which will be used shortly.

(10) **Lemma of Agreement (of ev and evu).** For a set W of worlds, valuation $V \in FV$, world w in W, and formula ϕ, $ev.V.w.\phi = evu.V.w.\phi$.

Proof. Since V is full, function evu yields only $\mathbf{t}$ or $\mathbf{f}$, so the third case of definition (9), the "otherwise" condition, can be ignored. Secondly, since $V \in FV$, the quantifications in the definition of evu can be simplified as follows: Using L for $\mathbf{f}$ or $\mathbf{t}$, we have

$$\begin{array}{ll} & (\forall V' \mid V' \in FX.V : ev.V'.w.\phi = L) \\ = & \quad \langle (8) \text{ — } V \in FV \rangle \\ & (\forall V' \mid V' = V : ev.V'.w.\phi = L) \\ = & \quad \langle \text{One-point rule: Provided } x \text{ not free in E, } (\forall x \mid x = E : P) = P[x := E] \rangle \\ & ev.V.w.\phi = L \end{array}$$

Hence, the definition of $evu.V.w.\phi$ reduces to $ev.V.w.\phi$ □

(11) **Lemma.** A formula ϕ has a value in a given world of an S5U model exactly when ϕ has that value in that world in all full extensions of the model: for L in $\{\mathbf{t}, \mathbf{f}\}$, $evu.V.w.\phi = L \equiv (\forall V' \mid V' \in FX.V : evu.V'.w.\phi = L)$.

Proof.
$$\begin{array}{ll} & (\forall V' \mid V' \in FX.V : evu.V'.w.\phi = L) \\ = & \quad \langle V' \text{ is full; use Lemma of Agreement (10)} \rangle \\ & (\forall V' \mid V' \in FX.V : ev.V'.w.\phi = L) \\ = & \quad \langle \text{Definition (9) of } evu \rangle \\ & evu.V.w.\phi = L \end{array}$$
□

(12) **Lemma.** For arbitrary set W of worlds and valuation V on W, w in W, and unary predicate P in which V does not appear free, $(\forall V', V \mid V' \in FX.V : P.V') \equiv (\forall V' \mid V' \in FV : P.V')$.

Proof. The proof is by mutual implication.

$$
\begin{array}{cl}
 & (\forall V', V \mid V' \in FX.V : P.V') \\
\Rightarrow & \quad \langle \text{Instantiation, with } V := V' \rangle \\
 & (\forall V' \mid V' \in FX.V' : P.V') \\
= & \quad \langle (7),\ V' \in FX.V' \equiv V' \in FV \rangle \\
 & (\forall V' \mid V' \in FV : P.V')
\end{array}
\qquad
\begin{array}{cl}
 & (\forall V' \mid V' \in FV : P.V') \\
= & \quad \langle \text{Introduce } V \rangle \\
 & (\forall V', V \mid V' \in FV : P.V') \\
\Rightarrow & \quad \langle \text{Antimonotonicity: (6), } FX.V \subseteq FV \rangle \\
 & (\forall V', V \mid V' \in FX.V : P.V') \qquad \square
\end{array}
$$

We now have the machinery to prove our main result about S5-validity, that we can use S5 in the standard way even if we treat our possible models as the set of S5U models. We can reason about partial functions without sacrificing any of S5.

(13) **Theorem.** For any formula ϕ, $\models_{\mathrm{S5}} \phi$ iff $\models_{\mathrm{S5U}} \phi$.

Proof. We begin with the definition of validity over S5 models and conclude with the definition of validity over S5U models. We take arbitrary set W of worlds and world w in W and assume V ranges over possible valuations on W. In the definition of validity over S5 models, we make explicit the restriction that only full valuations are considered.

$$
\begin{array}{cl}
 & (\forall V' \mid V' \in FV : ev.V'.w.\phi = \mathbf{t}) \\
= & \quad \langle \text{Lemma of agreement (10)} \rangle \\
 & (\forall V' \mid V' \in FV : evu.V'.w.\phi = \mathbf{t}) \\
= & \quad \langle (12),\ (\forall V' \mid V' \in FV : P.V') \ \equiv\ (\forall V', V \mid V' \in FX.V : P.V') \rangle \\
 & (\forall V', V \mid V' \in FX.V : evu.V'.w.\phi = \mathbf{t}) \\
= & \quad \langle \text{Nesting} \rangle \\
 & (\forall V \mid : (\forall V' : V' \in FX.V : evu.V'.w.\phi = \mathbf{t})) \\
= & \quad \langle\ (11),\ evu.V.w.\phi = L \ \equiv\ (\forall V' \mid V' \in FX.V : evu.V'.w.\phi = L\,) \\
 & \quad\ \ \text{with } L := \mathbf{t} \,\rangle \\
 & (\forall V \mid : evu.V.w.\phi = \mathbf{t}) \qquad \square
\end{array}
$$

(14) **Corollary.** All S5 axioms are valid in S5U. More precisely, since all axioms of S5 are valid over S5 models, they are valid over S5U models.

(15) **Corollary.** All inference rules of S5 are sound over S5U.

3 Underspecification for C

In [5], Gries and Schneider present a sound and complete logic C (after Carnap) for propositional formulas together with $\Box$ for the model consisting of the conventional set of states. In this section, we do for C what we did in the last section for S5, prove that underspecification respects C-validity.

(16) **Definition.** Let $\widehat{W}$ be the set of all total functions $w : VP \rightarrow \{\mathbf{t}, \mathbf{f}\}$. Define a valuation $\widehat{V}$ on $\widehat{W}$ by $\widehat{V}.w.p = w.p$. Then, $(\widehat{V}, \widehat{W})$ is the (only) *C model.*

(17) **Definition.** Let $\widehat{WU}$ be the set of all functions $w : VP \to \{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$, and define valuation $\widehat{VU}$ on $\widehat{WU}$ by $\widehat{VU}.w.p = w.p$. Then, $(\widehat{VU}, \widehat{WU})$ is the (only) *CU model.*

A formula ϕ is *C-valid*, denoted by $\models_{\mathrm{C}} \phi$, iff $ev.\widehat{V}.w.\phi = \mathbf{t}$ for all w in $\widehat{W}$. A formula ϕ is *CU-valid*, denoted by $\models_{\mathrm{CU}} \phi$, iff $evu.\widehat{VU}.w.\phi = \mathbf{t}$ for all w in $\widehat{WU}$.

To prove the correspondence between the C and CU models, we need a way to relate evaluation of formulas over $\widehat{WU}$ to their evaluations over $\widehat{W}$, to effectively collapse the larger model into the smaller one. Observe that, by definition, $\widehat{W}$ under $\widehat{V}$ consists of states, i.e. assignments to variables; the same is true for $\widehat{WU}$ under any full extension of $\widehat{VU}$. For any world w in $\widehat{W}$ and any V' in $FX.\widehat{VU}$, let $D.w$ be the set of all worlds in $\widehat{WU}$ that are the same under V' as w is under $\widehat{V}$:

(18) **Definition.** For V' in $FX.\widehat{VU}$ and w in $\widehat{W}$, define $D.w$ by $w' \in D.w \equiv w' \in \widehat{WU} \wedge \widehat{V}.w = V'.w'$.

Since V' is full, every w' in $\widehat{WU}$ is in $D.w$ for some w in $\widehat{W}$. In fact, the collection of sets $D.w$ is a partitioning of $\widehat{WU}$, since no two worlds in $\widehat{W}$ are identical. We have:

(19) **Collapsing Lemma.** For all $V' \in FX.\widehat{VU}$, formulas ϕ, w in $\widehat{W}$, and w' in $D.w$, $ev.\widehat{V}.w.\phi = ev.V'.w'.\phi$.

Proof. For an arbitrary V' in $FX.\widehat{VU}$, we prove the lemma by induction on the structure of ϕ. In the cases in which ϕ is a variable, a negation, or a disjunction, the proofs follow straightforwardly from choosing an arbitrary w and w' and are left to the reader. The remaining case $\phi = \Box\alpha$ itself has two similar subcases; we present only the subcase $ev.\widehat{V}.w.\Box\alpha = \mathbf{t}$. For arbitrary w in $\widehat{W}$ and w' in $D.w$, we have:

$$\begin{array}{ll}
 & ev.V'.w'.\Box\alpha = \mathbf{t} \\
= & \quad \langle \text{Definition (3) of } ev \rangle \\
 & (\forall w' \mid w' \in \widehat{WU} : ev.V'.w'.\alpha = \mathbf{t}) \\
= & \quad \langle \text{The sets } D.w \text{ partition } \widehat{WU} \rangle \\
 & (\forall w' \mid (\exists w \mid w \in \widehat{W} : w' \in D.w) : ev.V'.w'.\alpha = \mathbf{t}) \\
= & \quad \langle \text{Trading; } (\exists x \mid R : P) \Rightarrow Q \equiv (\forall x \mid R : P \Rightarrow Q), \text{ if } x \text{ not free in } Q \rangle \\
 & (\forall w' \mid : (\forall w \mid w \in \widehat{W} : w' \in D.w \Rightarrow ev.V'.w'.\alpha = \mathbf{t})) \\
= & \quad \langle \text{Trading; Inductive Hypothesis} \rangle \\
 & (\forall w' \mid : (\forall w \mid w \in \widehat{W} \wedge w' \in D.w : ev.\widehat{V}.w.\alpha = \mathbf{t})) \\
= & \quad \langle \text{Predicate calculus, } w' \text{ does not occur free in } ev.\widehat{V}.w.\alpha = \mathbf{t} \rangle \\
 & (\forall w \mid w \in \widehat{W} : ev.\widehat{V}.w.\alpha = \mathbf{t}) \\
= & \quad \langle \text{Definition (3) of } ev \rangle \\
 & ev.\widehat{V}.w.\Box\alpha = \mathbf{t}
\end{array}$$

□

We can now prove:

(20) **Theorem.** For any formula ϕ, $\models_{\mathrm{C}} \phi$ iff $\models_{\mathrm{CU}} \phi$.

We begin with the definition of CU-validity and show it equivalent to the definition of C-validity. For arbitrary ϕ, we have:

$$\begin{array}{ll}
 & (\forall w' \mid w' \in \widehat{WU} : evu.\widehat{VU}.w'.\phi = \mathbf{t}) \\
= & \quad \langle \text{Definition (9) of } evu \, \rangle \\
 & (\forall w' \mid w' \in \widehat{WU} : (\forall V' \mid V' \in FX.\widehat{VU} : ev.V'.w'.\phi = \mathbf{t})) \\
= & \quad \langle \text{Nesting} \rangle \\
 & (\forall V' \mid V' \in FX.\widehat{VU} : (\forall w' \mid w' \in \widehat{WU} : ev.V'.w'.\phi = \mathbf{t})) \\
= & \quad \langle \text{The sets } D.w \text{ partition } \widehat{WU} \, \rangle \\
 & (\forall V' \mid V' \in FX.\widehat{VU} : (\forall w' \mid (\exists w \mid w \in \widehat{W} : w' \in D.w) : ev.V'.w'.\phi = \mathbf{t})) \\
= & \quad \langle \text{Trading} \rangle \\
 & (\forall V' \mid V' \in FX.\widehat{VU} : (\forall w' \mid: (\exists w \mid w \in \widehat{W} : w' \in D.w) \Rightarrow ev.V'.w'.\phi = \mathbf{t})) \\
= & \quad \langle\, (\exists x \mid R : P) \Rightarrow Q \equiv (\forall x \mid R : P \Rightarrow Q\,), \text{ provided } x \text{ not free in } Q \,\rangle \\
 & (\forall V' \mid V' \in FX.\widehat{VU} : (\forall w' \mid: (\forall w \mid w \in \widehat{W} : w' \in D.w \Rightarrow ev.V'.w'.\phi = \mathbf{t}))) \\
= & \quad \langle \text{Trading} \rangle \\
 & (\forall V' \mid V' \in FX.\widehat{VU} : (\forall w' \mid: (\forall w \mid w \in \widehat{W} \wedge w' \in D.w : ev.V'.w'.\phi = \mathbf{t}))) \\
= & \quad \langle \text{Collapsing Lemma (19)} \rangle \\
 & (\forall V' \mid V' \in FX.\widehat{VU} : (\forall w' \mid: (\forall w \mid w \in \widehat{W} \wedge w' \in D.w : ev.\widehat{V}.w.\phi = \mathbf{t}))) \\
= & \quad \langle \text{Predicate calculus, } V' \text{ and } w' \text{ do not appear free in } ev.\widehat{V}.w.\phi = \mathbf{t} \,\rangle \\
 & (\forall w \mid w \in \widehat{W} : ev.\widehat{V}.w.\phi = \mathbf{t}) \qquad \square
\end{array}$$

We close this section with a result about underspecification for PC: propositional logic with its conventional notion of *PC-validity*. The language of PC is that of C with □ removed, and the model and semantics for PC are the same as those for C. We incorporate underspecification by extending this basis to *PCU-validity* and its associated semantics in the same way we extended C to CU, so as it pertains to underspecification, PCU is to PC as CU is to C. It is then clear that the theorem concerning C-validity extends to PC-validity:

(21) **Corollary: Underspecification over PC.** For any propositional formula ϕ, ϕ is PC-valid iff ϕ is PCU-valid.

For propositional logic, Bijlsma [1] derives a similar result, but couched in terms of the undefined instead of underspecification. We prefer underspecification because then all functions and operations can be total and we can continue to use simple two-valued logic as the logic underlying all domains.

References

[1] Bijlsma, A. Semantics of quasi-boolean expressions. In Feijen, W.H.J., et al (eds.) *Beauty is Our Business*. Springer Verlag, New York, 1990, 27–35.

[2] Gries, D., and F.B. Schneider. *A Logical Approach to Discrete Math*. Springer-Verlag, New York, 1993.

[3] Gries, D., and F.B. Schneider. Equational propositional logic. *IPL 53* (1995), 145-152.

[4] Gries, D., and F.B. Schneider. Avoiding the undefined by underspecification. In J. van Leeuwen (Ed.). *Computer Science Today*. Springer Verlag Lecture Notes in Computer Science 1000, October 1995, 366–373.

[5] Gries, D., and F.B. Schneider. Adding the everywhere operator to propositional logic. Tech. Rpt. 96-1583, Computer Science Department, Cornell University, May 1996.

[6] Hughes, G.E., and M.J. Cresswell. *An Introduction to Modal Logic*. Mehuen and Co., New York, 1968.