Log. Univers. 7 (2013), 125–145

1661-8297/13/020125-21, *published online* September 16, 2012
DOI 10.1007/s11787-012-0069-2

Logica Universalis

A Shared Framework for Consequence Operations and Abstract Model Theory

Christian Wallmann

Abstract. In this paper we develop an abstract theory of adequacy. In the same way as the theory of consequence operations is a general theory of logic, this theory of adequacy is a general theory of the interactions and connections between consequence operations and its sound and complete semantics. Addition of axioms for the connectives of propositional logic to the basic axioms of consequence operations yields a unifying framework for different systems of classical propositional logic. We present an abstract model-theoretical semantics based on model mappings and theory mappings. Between the classes of models and theories, i.e., the set of sentences verified by a model, it obtains a connection that is well-known within algebra as Galois correspondence. Many basic semantical properties can be derived from this observation. A sentence A is a semantical consequence of T if every model of T is also a model of A. A model mapping is adequate for a consequence operation if its semantical inference operation is identical with the consequence operation. We study how properties of an adequate model mapping reflect the properties of the consequence operation and vice versa. In particular, we show how every concept of the theory of consequence operations can be formulated semantically.

Mathematics Subject Classification (2010). Primary 03C95; Secondary 03B22.

Keywords. Consequence operation, abstract model theory, Galois correspondence, completeness, adequacy, classical propositional logic.

1. Introduction

Currently there exists a variety of different logics and each of these logics can be defined through different axioms, rules or semantics. There has been a great effort to provide a general framework for all these logics [4]. The syntactical theory of consequence operations (see, e.g., [15–17]) is such a framework. A similar degree of generality has been achieved in semantics by the development of abstract model theory [1,2]. Surma [14] was the first to study the interaction between consequence operations and semantics using an axiomatic approach. However, apart from the work of Surma, there has been little effort to study what is common to all relations of adequacy between syntax and semantics. In the present contribution we develop such a theory of adequacy for consequence operations on the syntactical side and abstract model theory. We study how certain properties of consequence operations determine properties of its adequate semantics and vice versa. Another main aim of the paper is to show how syntactical concepts can be formulated semantically. Such a formulation clearly depends on the class of consequence operations and semantics considered. In particular, we treat adequacy for classical propositional logic.

After presenting the basic axioms for consequence operations and some basic notions of logic, we state axioms for the connectives of classical propositional logic. The resulting concept of propositional consequence operation covers all systems of classical logic. By employing semantics based on relatively maximal sets, we can prove the completeness of many logics in a much more simplified way [3,17]. We review some of these results and show that every finitary propositional consequence operation is complete with respect to classical logic [17].

In Sect. 3, we present an abstract semantics [7,9] which has the same degree of generality as consequence operations. The set of structures on which the semantics is based on is not specified. It could be any non-empty set. As a consequence, our semantical framework covers many different systems, such as valuation semantics, semantics based on maximally consistent sets, and probability semantics. Roughly speaking, a model mapping Mod assigns to every formula the set of structures that verify it. The theory $Th(N)$ of a model N is the set of all sentences verified by N. Mod and Th form a Galois correspondence [6]—a relation that is well-established within algebra [5,6]. This observation is of main importance because many semantical facts derive immediately from the theory of Galois correspondences. The semantical consequence operation is given by the mapping $Th \circ Mod$. It turns out that a sentence A is a semantical consequence of a set of sentences T, if and only if every model of T is a model of A. We study a special class of model mappings for propositional consequence operations. This class, 'propositional model mappings', has the Negation property and the Conjunction property. Finally, we review an alternative approach towards abstract semantics that is based on deductively closed sets instead of models [10].

In the last section, we develop a theory of adequacy that can be applied to many different kinds of logic. A semantics is adequate for Cn iff Cn is identical with the semantical inference operation $Th \circ Mod$. After studying adequacy in its most general form, we investigate how properties of Mod reflect properties of Cn and vice versa. We treat the cases where Cn is a propositional consequence operation and where Mod is a propositional model mapping. Furthermore, we determine for every basic notion of the theory of consequence operations a semantical equivalent.

2. Syntax

In this section, we outline the theory of consequence operations and the theory of propositional consequence operations.

2.1. Consequence Operations

2.1.1. Basic Framework. Let Av be a countable infinite set of propositional variables and $f_1, \ldots, f_n$ be a set of propositional connectives. A *formal language* is the smallest set closed under $f_1, \ldots, f_n$ and containing Av.

Suppose that L is a formal language, that $T, T' \subseteq L$, and that $A, B \in L$. The axioms given in [15] are equivalent to those given in

Definition 2.1. A mapping $Cn\colon 2^L \to 2^L$ is a **consequence operation** or **closure operator** iff for every T, T':

1. $T \subseteq Cn(T)$ (Reflexivity) and
2. if $T \subseteq Cn(T')$, then $Cn(T) \subseteq Cn(T')$ (Transitivity).

Corollary 2.2. *Cn is a consequence operation iff for every T, T':*

1. $T \subseteq Cn(T)$,
2. $Cn((Cn(T)) \subseteq Cn(T)$ *(Idempotency), and*
3. *if* $T \subseteq T'$, *then* $Cn(T) \subseteq Cn(T')$ *(Monotonicity).*

For the remainder of this article, suppose that Cn is a consequence operation.

A consequence operation is structural [11] if it is closed with respect to substitution. Uniform and simultaneous substitution of an arbitrary formula for a propositional variable within a valid inference, yields also a valid inference. The validity of an inference consequently does not depend on the content, but only on the form, i.e., on the kind and order of the occurring connectives. A mapping $e\colon Av \to L$ is called *substitution*. Every substitution can be extended to an endomorphism $h_e\colon L \to L$, i.e., to a uniform and simultaneous substitution.

Definition 2.3. Cn is **structural** iff $h_e(Cn(T)) \subseteq Cn(h_e(T))$ for every substitution e.

Definition 2.4.

Cn is **stronger** than Cn' ($Cn' \leq Cn$) iff for all T: $Cn'(T) \subseteq Cn(T)$.
Cn is **properly stronger** than Cn' ($Cn' < Cn$) iff Cn is stronger than Cn' and there is some T such that $Cn'(T) \subset Cn(T)$.

Basic concepts of logic can be defined within the theory of consequence operations.

Definition 2.5.
1. Cn is **consistent** iff $Cn(\emptyset) \neq L$.
2. Cn is **finitary** iff for each $T \subseteq L$: $Cn(T) = \bigcup\{Cn(T') : T' \subseteq T$ and T' is finite$\}$.
3. Cn is **compact** iff for each $T \subseteq L$: If $Cn(T) = L$, then there exists a finite $T' \subseteq T$ such that $Cn(T') = L$.
4. T is a Cn-**theory** iff $T = Cn(T)$.

5. T is Cn-**consistent** iff $Cn(T) \neq L$.
6. T is Cn-**complete** iff for all A: If $T \cup \{A\}$ is consistent, then $A \in Cn(T)$.
7. T is Cn-**maximally consistent** iff T is consistent, and there does not exist a consistent $T' \supset T$.
8. T is a Cn-**axiom system** for T' iff $Cn(T) = Cn(T')$.
9. A is in T Cn-**independent** iff $A \in T$ and $A \notin Cn(T \backslash \{A\})$.
10. A is a Cn-**tautology** iff $A \in Cn(\emptyset)$.

Although finitariness and compactness are in many cases equivalent, this is generally not the case.

Lemma 2.6. *Let Cn be finitary. Then: Cn is compact iff L has a finite Cn-axiom system.*

The next lemma facilitates the proof of Theorem 4.11.

Lemma 2.7. *T is Cn-maximally consistent iff T is a consistent and complete Cn-theory.*

2.1.2. Completeness. To show that every propositional consequence operation is complete with respect to classical logic, we employ semantics based on relatively maximal sets. The set of its relatively maximal sets is an adequate semantics for a finitary consequence operation (for the notion of adequacy, refer to Definition 4.1 and especially the subsequent example). This fact hinges essentially on the Lindenbaum Lemma. Consequently, by identifying a relatively maximal set with its characteristic function, every finitary consequence operation has a bivalent semantics. Furthermore, the set of relatively maximal sets forms a minimal semantics, i.e., every semantics that does not contain all relatively maximal sets, is not adequate. The framework of this section is very general and can be applied to prove the completeness of many logics. The results of this section can be found, e.g., in [3], [17, p. 25–28].

Definition 2.8. A set $\mathcal{E}$ is a **closure system** iff it is closed under intersection, i.e., if $\mathcal{D} \subseteq \mathcal{E}$, then $\bigcap \mathcal{D} \in \mathcal{E}$.

The following fact is well-known (see, e.g., [6]) and implies that the closure system of all Cn-theories forms an adequate semantics for Cn.

Lemma 2.9. *Every consequence operation Cn, i.e., closure operator, determines a closure system $\{T | Cn(T) = T\}$. Conversely, every closure system $\mathcal{E}$ determines a closure operator Cn through the condition:*
$A \in Cn(T)$ iff $A \in \bigcap \{T' \in \mathcal{E} | T \subseteq T'\}$.

Definition 2.10. $\mathcal{G} \subseteq \mathcal{E}$ is a **generator set** of a closure system $\mathcal{E}$ iff for every $E \in \mathcal{E}$: $E = \bigcap G'$ for some $G' \subseteq \mathcal{G}$. A subset of $\mathcal{E}$ is a **minimal generator set** iff it is not generated by one of its proper subsets. T is called **completely meet irreducible** in $\mathcal{E}$ iff if $T = \bigcap G'$ for some $G' \subseteq \mathcal{E}$, then $T \in G'$.

The next lemma shows that for a consequence operation Cn, it is sufficient to consider a generator set of the closure system of Cn. Hence, every generator set also forms an adequate semantics for Cn.

Lemma 2.11. *Suppose that G is a generator set of $\mathcal{E}$.*
Then: $\bigcap\{T' \in \mathcal{E} | T \subseteq T'\} = \bigcap\{T' \in \mathcal{G} | T \subseteq T'\}$.

Definition 2.12. T is **relatively Cn-maximal in** A iff $A \notin T$ and if $B \notin T$, then $A \in Cn(T \cup \{B\})$. T is relatively Cn-maximal iff it is relatively maximal in some $A \in L$.

Observe that every relatively maximal set is a theory. We denote the set of relatively Cn-maximal theories by $RELMAX(Cn)$ and the set of relatively maximal extensions of T by $RELEXT(T)$.

No relatively maximal set T can be generated by a set of closures not containing T. This implies that if $RELMAX(Cn)$ is a generator set, it is a minimal generator set.

Lemma 2.13. *T is relatively Cn-maximal iff it is completely meet irreducible in the closure system given by Cn.*

From the Lindenbaum Lemma, it follows that $RELMAX(Cn)$ is a generator set.

Lemma 2.14 (Lindenbaum Lemma). *Let Cn be finitary. For every T where $A \notin T$, there exists a $T' \supseteq T$ that is relatively maximal in A.*

The following lemma states that the set of all relatively maximal sets forms a minimal adequate semantics for finitary consequence operations. As already pointed out, it can be obtained by Lemma 2.13.

Lemma 2.15. *If Cn satisfies the Lindenbaum Lemma, then $RELMAX(Cn)$ is a generator set of the closure system generated by Cn. Specifically, $T = \bigcap RELEXT(T)$. Moreover, $RELMAX(Cn)$ is a minimal generator set.*

If Cn is finitary, then the Lindenbaum Lemma is valid. As a consequence, $RELMAX(Cn)$ is a generator set, and we have the following tool to prove the completeness of finitary consequence operations.

Theorem 2.16. *If Cn is finitary and $RELMAX(Cn)$ is a subset of the closure system generated by Cn', then $Cn' \leq Cn$.*

2.2. Propositional Consequence Operations

The concept of propositional consequence operation (see, e.g., [15], [17, p. 110]) covers all systems of classical propositional logic. Since it describes the connectives of classical logic, the underlying language is the formal language generated by the connectives $\neg, \wedge, \vee, \rightarrow$. We denote this language by 'L_{AL}' and write '$Cn(T, A_1, \ldots, A_n)$' instead of '$Cn(T \cup \{A_1, \ldots, A_n\})$'. It is fundamental that every finitary, consistent, and structural consequence operation that satisfies the conditions ($\neg$), ($\wedge$), ($\vee$), and ($\rightarrow$), is identical with classical logic. We sketch the proof in this section (for a detailed proof see, e.g., [17, pp. 123–127]).

2.2.1. Propositional Consequence Operations.

Definition 2.17. A consequence operation Cn is a **propositional consequence operation** iff for $A, B \in L_{AL}$ and $T \subseteq L_{AL}$:

($\neg$) $A \in Cn(T)$ iff $Cn(T, \neg A) = L_{AL}$,
($\wedge$) $Cn(T, A \wedge B) = Cn(T, A, B)$,
($\vee$) $Cn(T, A \vee B) = Cn(T, A) \cap Cn(T, B)$, and
($\rightarrow$) $A \rightarrow B \in Cn(T)$ iff $B \in Cn(T, A)$.

Example. A is a **classical consequence** of T iff for every Boolean valuation α: If $\models_\alpha T$, then $\models_\alpha A$. We denote the classical consequence operation by $Cn_{\models}$. It can easily be verified that the classical consequence operation is a propositional consequence operation.

In order to prove Theorems 4.8 and 4.14, we have to employ several properties of propositional consequence operations.

Lemma 2.18. *Let Cn be a propositional consequence operation. Then:*

1. *If $A, A \rightarrow B \in T$, then $B \in Cn(T)$.*
2. *$\{A, \neg A\}$ is not Cn-consistent.*
3. *$A, B \in Cn(A \wedge B)$ and $A \wedge B \in Cn(A, B)$.*
4. *$B \in Cn(A \vee B, \neg A)$ and $A \vee B \in Cn(A) \cap Cn(B)$.*
5. *$A \rightarrow B \in Cn(B)$ and $A \rightarrow B \in Cn(\neg A)$.*

The following closure properties of relatively maximal sets play a key role in proving that every propositional consequence operation is complete with respect to classical logic (for a proof see [17, p. 125–126]).

Lemma 2.19. *Let Cn be a propositional consequence operation and T be relatively Cn-maximal. Then:*

1. *$A \in T$ iff $\neg A \notin T$.*
2. *$A \wedge B \in T$ iff $A \in T$ and $B \in T$.*
3. *$A \vee B \in T$ iff $A \in T$ or $B \in T$.*
4. *$A \rightarrow B \in T$ iff $A \notin T$ or $B \in T$.*

Consistency and completeness can be formulated in a familiar way within the theory of propositional consequence operations.

Lemma 2.20. *Let Cn be a propositional consequence operation. Then:*

1. *T is Cn-consistent iff there is no sentence A such that $A, \neg A \in Cn(T)$.*
2. *T is Cn-complete iff for all sentences A, $A \in Cn(T)$ or $\neg A \in Cn(T)$.*

2.2.2. The Completeness of Finitary Propositional Consequence Operations. If Cn is a finitary propositional consequence operation, then the set of all its maximally consistent sets is a generator set of $\{T|Cn(T) = T\}$. Consequently, this set is an adequate semantics for Cn. By $MAX(Cn)$ we denote the set of all maximally Cn-consistent sets, and by $EXTMAX(T)$, we denote the set of all maximally consistent extensions of T.

Theorem 2.21. *Suppose that Cn is a propositional consequence operation. Then, $MAX(Cn) = RELMAX(Cn)$. In particular, $T = \bigcap EXTMAX(T)$. In addition, if Cn satisfies the Lindenbaum Lemma, then $MAX(Cn)$ is a minimal generator set of the closure system $\{T|Cn(T) = T\}$.*

Proof. $MAX(Cn) = RELMAX(Cn)$ follows from Lemmas 2.19 (1) and 2.20. Since Cn satisfies the Lindenbaum Lemma, it is according to Lemma 2.15 $MAX(Cn)$ a minimal generator set. □

According to Theorem 2.16, for completeness of Cn relative to classical logic, it is sufficient that $RELMAX(Cn)$ is a subset of $RELMAX(Cn_{\models}) = MAX(Cn_{\models})$. I.e., for every relatively maximally consistent set T, there is a Boolean valuation α such that $T = \{A \in L_{AL} | \models_{\alpha} A\}$. This fact hinges essentially on Lemma 2.19 and is not very difficult to prove by induction on the complexity of formulas.

Theorem 2.22. *Let Cn be a finitary propositional consequence operation. Then, $Cn_{\models} \leq Cn$, i.e., classical consequence is the weakest propositional consequence operation.*

Moreover, every finitary, structural, and consistent propositional consequence operation is identical with classical logic (for a proof, see [17, p. 127]).

Theorem 2.23. *There is no structural and consistent consequence operation Cn such that $Cn_{\models} < Cn$.*

Corollary 2.24. *If Cn is a finitary, structural, and consistent propositional consequence operation, then $Cn = Cn_{\models}$.*

3. Semantics

The basic framework of this section is that of abstract model theory (see, e.g., [7,9]). As observed, for instance, by Cohn [6, p. 205], abstract model theory can be based on a Galois correspondence between models and its theories. To be as general as possible, the set of structures Str is not specified. The following relation-based construction of Galois correspondences is attributed to Birkhoff (first edition of [5], 1940). The starting point of the construction is a satisfaction relation $\models_{Str}$ on $Str \times L$. The corresponding polarities [5] are the model mapping Mod and the theory mapping Th. Between Mod and Th, it obtains a Galois correspondence. All theorems of the present section derive from the theory of Galois correspondences. The mapping $Th \circ Mod$, for instance, is a consequence operation. It is the semantical consequence operation.

Propositional model mappings are obtained by adding axioms for the connectives of propositional logic. As we shall see in Sect. 4.3, they induce propositional consequence operations. At the end of this section, we review an alternative approach towards abstract semantics that is based on theories instead of models [10].

3.1. Model Mappings and Semantical Inference Operations

Let Str be an arbitrary non-empty set and L be a formal language.

The class of models $Mod(T)$ of T is the set of all models that satisfy T. The semantical theory $Th(M)$ of M is the set of all sentences that are satisfied by M.

Definition 3.1. Let $\models_{Str}$ be a binary relation on $Str \times L$.

1. $Mod\colon 2^L \to 2^{Str}$ is the $\models_{Str}$-**model mapping** iff
   $Mod(T) = \{N \in Str | N \models_{Str} A$ for all $A \in T\}$.
2. $Th\colon 2^{Str} \to 2^L$ is the $\models_{Str}$-**theory mapping** iff
   $Th(M) = \{A \in L | N \models_{Str} A$ for all $N \in M\}$.

Example. The following model mappings serve as examples throughout the article.

$Str_1 = \{\alpha : \alpha$ is a Boolean valuation$\}$
$Str_2 = \{M : M$ is a maximally consistent set of classical propositional logic$\}$
$Str_3 = \{p : p$ is a one-place probability function on $L_{AL}\}$[1]

$Mod_1(A) = \{\alpha : \models_\alpha A\}$
$Mod_2(A) = \{M : A \in M\}$
$Mod_3(A) = \{p : p(A) = 1\}$

As already pointed out, if two mappings are defined by a binary relation in the way described above, then they form a Galois correspondence [5].

Theorem 3.2. *The mappings Mod and Th form a Galois correspondence between* 2^L *and* 2^{Str}*, i.e.,*

1. *For every two elements of* 2^L *with* $T \subseteq T'$ *and for every two elements of* 2^{Str} *with* $M \subseteq M'$
$$Mod(T') \subseteq Mod(T) \quad and \quad Th(M') \subseteq Th(M)\ . \tag{3.1}$$
2. *For any* $T \in 2^L$ *and* $M \in 2^{Str}$
$$T \subseteq Th(Mod(T)) \quad and \quad M \subseteq Mod(Th(M))\ . \tag{3.2}$$

Mappings that satisfy (3.1) are called antitone. The following consequences are well-known within the theory of Galois correspondences.

Lemma 3.3. 1. $M \subseteq Mod(T)$ *iff* $T \subseteq Th(M)$.
2. $Th(Mod(Th(M))) = Th(M)$.
3. $Mod(Th(Mod(T))) = Mod(T)$.

Lemma 3.4. 1. $Mod(\bigcup T_i) = \bigcap Mod(T_i)$.
2. $Th(\bigcup T_i) = \bigcap Th(T_i)$.
3. $Mod(\emptyset) = Str$.

The mapping $Th \circ Mod\colon 2^L \to 2^L$ is the semantical inference operation. $Th \circ Mod$ is a consequence operation. $Mod \circ Th$ is a closure operator. This is an immediate consequence of the fact that Mod and Th form a Galois correspondence.

---

[1] A mapping $p\colon L \to [0, 1]$ is called a **one-place probability function** iff p satisfies the following postulates:
$P(A) = 1$ for some $A \in L$,
$P(A) \leq P(B)$ whenever $A \models B$,
$P(A \vee B) = P(A) + P(B)$, if $\models \neg(A \wedge B)$.

Theorem 3.5. *$Th \circ Mod$ is a consequence operation. Moreover, $Mod \circ Th$ is a closure operator.*

Proof. Reflexivity is given by the left side of (3.2). For *monotonicity*, observe that if $T \subseteq T'$, then by (3.1) $Mod(T') \subseteq Mod(T)$. Hence, by (3.1) $Th(Mod(T)) \subseteq Th(Mod(T'))$. *Idempotency* is obtained by setting $M = Mod(T)$ in Lemma 3.3 (2). The proof for $Mod \circ Th$ is analogous. □

The set of all semantical theories is the closure system of the closure operator $Th \circ Mod$ and the set of all axiomatic classes is the closure system of the closure operator $Mod \circ Th$.

Definition 3.6. T is a **semantical theory** iff $T = Th(M)$ for some $M \subseteq Str$. By THE, we denote the set of all semantical theories.
M is called an **axiomatic class** iff $M = Mod(T)$ for some $T \subseteq L$. By AXC, we denote the set of all axiomatic classes.

Lemma 3.7. *THE (resp. AXC) is the closure system of $Th \circ Mod$ (resp. $Mod \circ Th$).*

Proof. Every $Th(M)$ is closed under $Th \circ Mod$, since according to Lemma 3.3 $Th(Mod(Th(M))) = Th(M)$. Since $T = Th(Mod(T))$ and $Mod(T) \subseteq Str$, the claim follows. The proof for AXC is analogous. □

A sentence A is a semantical consequence of a set T, if and only if every model of T is a model of A. Since THE is the closure system of $Th \circ Mod$, it is according to Lemma 2.9, $A \in Th(Mod(T))$ iff $A \in \bigcap\{T' \in THE | T \subseteq T'\}$.

Lemma 3.8. *The following three conditions are equivalent.*

1. $A \in Th(Mod(T))$.
2. $Mod(T) \subseteq Mod(A)$.
3. $A \in \bigcap\{T' \in THE | T \subseteq T'\}$.

Example. The semantical inference operations of our exemplary model mappings are given by:

$A \in F_{Mod1}(T)$ iff for every Boolean valuation α: If $\models_\alpha T$, then $\models_\alpha A$.
$A \in F_{Mod2}(T)$ iff for every classical maximally consistent set M: If $T \subseteq M$, then $A \in M$.
$A \in F_{Mod3}(T)$ iff for every probability function p: If $p(B) = 1$ for every $B \in T$, then $p(A) = 1$.

Although they are generated by rather different model mappings, each of the above inference operations is identical with classical logic. The (non trivial) fact that F_{Mod3} is classical logic is well-known in the area of probability semantics [12, Theorem 5.3]. Here, we give a proof that makes use of our general results concerning completeness (Sect. 2.1.2).

Theorem 3.9. *Each of the consequence operations F_{Mod1}, F_{Mod2}, and F_{Mod3} is identical with classical logic.*

Proof. For F_{Mod2}, this follows from Lemma 2.21. For F_{Mod3} observe that the semantical theories of probability functions are closed under classical consequence. The set of all theories of Boolean valuations generates the set of all classical theories. Since every Boolean valuation is a probability function, the set of all theories of Boolean valuations generates the closure space generated by the set of all probability functions. Therefore, by Lemma 2.11, F_{Mod_3} is classical logic. □

The operation Mod is a bijection from THE to AXC. The inverse of Mod is Th. Moreover, Mod is a dual isomorphism[2] between these two sets (see, e.g., [5]). Mod inverts the order of THE, i.e., larger theories correspond to smaller classes of models and vice versa.

Theorem 3.10. *The mappings Mod and Th determine a dual isomorphism between THE and AXC.*

We conclude this section with a general remark concerning an alternative approach towards semantics that is based on theories of models, rather than models [10].

Since $Th(M) = \bigcap\{Th(\{N\})|N \in M\}$ (Lemma 3.4 (2)), $\{Th(\{N\})|N \in Str\}$ is a generator set of the closure system THE. A generator set uniquely determines a consequence operation (Lemma 2.11). Consequently, from the point of view of the consequence operation $Th \circ Mod$, instead of working with a set of models, one can also work with the set of their theories $\{Th(\{N\})|N \in Str\}$. Moreover, since the set of all relatively maximal theories (or completely meet irreducible theories, or totally prime theories) is a minimal generator set, if a logic is minimally generated, it is sufficient to consider relatively maximal theories. Abstract connectives are then interpreted as closure conditions on relatively maximal sets (see also Remark 1 below). This approach leads to interesting results concerning intuitionistic and classical propositional logic. One can give, for instance, elegant semantical characterizations of intuitionistic and propositional consequence operations [10, Theorem 3.2].

However, this approach is not feasible if we move to first-order logic. If we identify models with their theories, we cannot distinguish between elementary equivalent models, i.e., models that have the same first-order theory. As a consequence, many important concepts and results of first-order logic become inaccessible. Isomorphy of models and categoricity of theories[3] are excluded from this framework. Moreover, the theorem of Löwenheim–Skolem which permits a model-theoretical characterization of first-order logic [7] cannot be obtained.

3.2. Propositional Model Mappings

We have seen that $Th \circ Mod$ is a consequence operation. It is natural to ask which properties of Mod are sufficient for $Th \circ Mod$ being a propositional

[2] A mapping $I\colon R \to S$ is dual isomorphism between two partially ordered sets $(R, \leq)$ and $(S, \leq)$ iff I is a bijection and for all $r, r' \in R$: $r \leq r'$ iff $I(r) \geq I(r')$.

[3] A theory T is categorical iff every two models of T are isomorphic.

consequence operation. This question is dealt with in Theorem 4.9. We call these model mappings 'propositional model mappings'. However, the fact that $Th \circ Mod$ is a propositional consequence operation does not imply that Mod is a propositional model mapping.

Definition 3.11. Mod is a **propositional model mapping** iff

1. $Mod(\neg A) = Mod(A)^c$ [4] (Negation property),
1. $Mod(A \wedge B) = Mod(A) \cap Mod(B)$ (Conjunction property),
2. $Mod(A \vee B) = Mod(A) \cup Mod(B)$, and
3. $Mod(A \rightarrow B) = Mod(\neg A) \cup Mod(B)$.

Example. Mod_1 and Mod_2 are propositional model mappings. Mod_3 is not a propositional model mapping.

Remark 1 (Semantics based on maximally consistent sets). The closure operator $Th \circ Mod$ is determined by the generator set $\{Th(N)|N \in Str\}$ (Lemma 2.11). Each set $Th(N)$ is maximally consistent (Theorem 4.11). Hence, the closure conditions stated for relatively maximal sets in [10, Definition 3.1] are equivalent with those below. Because $N \in Mod(A)$ iff $A \in Th(N)$, Definition 3.11 is also equivalent with those conditions.

1. $A \in Th(N)$ iff $\neg A \notin Th(N)$.
2. $A \wedge B \in Th(N)$ iff $A \in Th(N)$ and $B \in Th(N)$.
3. $A \vee B \in Th(N)$ iff $A \in Th(N)$ or $B \in Th(N)$.
4. $A \rightarrow B \in Th(N)$ iff $A \notin Th(N)$ or $B \in Th(N)$.

Consequently, for propositional model mappings the approach taken here and in [10] are equivalent.

4. Adequacy

A semantical system is adequate with respect to a consequence operation, if and only if both yield the same theorems[5]. This leads to a concept of adequacy that does not dependent upon the kind of logic considered. We study the interaction between a consequence operation and its adequate semantics in its most general form. We show how all concepts defined in the framework of consequence operations (Definition 2.5) can be expressed semantically. We then investigate adequacy for propositional consequence operations. However, only if we require Mod to be a propositional model mapping, we obtain familiar connections between semantics and syntax of classical logic. As a main result, we shall see that every consequence operation that has an adequate propositional model mapping is a propositional consequence operation but not vice versa.

[4] $Mod(A)^c = Str \backslash Mod(A)$.

[5] The relation of adequacy between a consequence operation and our semantics has been introduced by Kleinknecht [8].

4.1. Adequacy for Consequence Operations

Definition 4.1. Let Mod be a model mapping and Cn be a consequence operation. Mod is Cn-**adequate**[6] iff $Cn = Th \circ Mod$.

Example. The closure system of all Cn-theories forms an adequate semantics for Cn (compare Lemma 2.9). Let $Str = \{X \subseteq L | Cn(X) = X\}$, $X \models_{Str} B$ iff $B \in X$. Then: $X \in Mod(T)$ iff $T \subseteq X$, and $Th(X) = X$. Furthermore, $A \in Cn(T)$ iff $A \in \bigcap\{X \in Str | T \subseteq X\}$.

Satisfiability and logical truth are semantical equivalents for consistency and tautology.

Definition 4.2.
1. T is Mod-**satisfiable** iff $Mod(T) \neq Mod(L)$.
2. T is **maximally** Mod-**satisfiable** iff there does not exist a Mod-satisfiable $T' \supset T$.
3. A is a Mod-**logical truth** iff $Str \subseteq Mod(A)$.

Each notion in Definition 2.5 can be expressed semantically if the concept of adequate model mapping is used as a link between syntax and semantics. Since THE and AXC are dually isomorphic (Theorem 3.10), we have the following picture (see also [13]). $Cn(L)$ is the largest set in THE and corresponds therefore to the smallest set of AXC. Because Mod is antitone, this set is $Mod(L)$. Since the correspondence is one to one, all consistent sets correspond to model classes different from $Mod(L)$. Maximally consistent sets are next in the ordering within THE and correspond consequently to sets of the form $M \cup Mod(L)$, such that all members of M are elementary equivalent. The converse need not be true. Observe further that since $Mod(Cn(T)) = Mod(T)$, maximally consistent sets and consistent and complete sets have the same models. Each theory that is not maximal, corresponds to a subset of the remaining axiomatic classes inversely ordered. On the lower end of THE, $Cn(\emptyset)$ corresponds to Str. We obtain

Theorem 4.3. *Let Mod be Cn-adequate. Then:*

1. *T is Cn-consistent iff T is Mod-satisfiable.*
2. *A is a Cn-tautology iff A is a Mod-logical truth.*
3. *T is a Cn-theory iff there exists an $M \subseteq Str$ such that $T = Th(M)$.*
4. *The consequence operation Cn is consistent iff $Mod(L) \neq Str$.*
5. *T is a Cn-axiom system for T' iff $Mod(T) = Mod(T')$.*
6. *T is maximally Cn-consistent iff T is maximally Mod-satisfiable.*

Proof. (1) Since $L = Cn(L)$, we obtain by Lemma 3.3 $Cn(T) = L$ iff $Mod(Cn(T)) = Mod(Cn(L))$ iff $Mod(T) = Mod(L)$. (2) follows from $Mod(\emptyset) = Str$ (Lemma 3.4). (3) is given by Lemma 3.7. For (4) observe

[6] Adequacy of Mod with respect to Cn corresponds to the completeness and soundness of the logic frame $(Str, L, \models_{str}, Cn)$ in the sense of [7]. Instead of considering a set of axioms and inference rules A as in the definition of logic frame of [7], we consider here a consequence operation Cn.

that $Cn(\emptyset) = L$ iff $Mod(\emptyset) = Mod(L)$. Since $Mod(\emptyset) = Str$, the claim follows. (5) is an immediate corollary of the fact that $Mod(T) = Mod(Cn(T))$ (Lemma 3.3). □

Let $M \subseteq Str$ and $N, N' \in Str$. With the help of the following notion, a semantical formulation for completeness can be given.

Definition 4.4. N, N' are **elementary equivalent** ($N \equiv N'$) iff $Th(\{N\}) = Th(\{N'\})$.

The following result states that a set is complete, if and only if any two "consistent" models of this set satisfy exactly the same formulas.

Theorem 4.5. *T is Cn-complete iff for every N, N' such that $N, N' \in Mod(T) \backslash Mod(L)$: $N \equiv N'$.*

Proof. $\Longrightarrow$: Let $N, N' \in Mod(T) \backslash Mod(L)$, $A \in L$.

Case 1: $\{A\} \cup T$ is consistent. By completeness $A \in Cn(T)$, i.e., $Mod(T) \subseteq Mod(A)$, and hence $N, N' \in Mod(A)$.

Case 2: $\{A\} \cup T$ is inconsistent. By Lemma 4.3 $Mod(L) = Mod(\{A\} \cup T) = Mod(A) \cap Mod(T)$. Hence, since $N, N' \notin Mod(L)$, $N, N' \notin Mod(A)$.

$\Longleftarrow$: Suppose that $\{A\} \cup T$ is consistent. Let $N \in Mod(T \cup \{A\}) \backslash Mod(L)$. Then by supposition for all $N' \in Mod(T) \backslash Mod(L)$, $N \equiv N'$. Hence, $N \in Mod(A)$, $N' \in Mod(A)$. Since for every $N'' \in Mod(L)$, $N'' \in Mod(A)$, $Mod(T) \subseteq Mod(A)$. □

As a corollary, we obtain

Corollary 4.6. *If T is maximally Cn-consistent, then $T = Th(\{N\})$ for some $N \in Str$.*

Proof. Let T be maximally consistent. By Lemma 2.7, T is a consistent and complete theory.
Since T is a theory, by Lemma 4.3 Part 3, $T = Th(M)$ for some $M \subseteq Str$. Since $Th(M) = T$ is consistent, it is according to Lemma 4.10 (3) $M \supset Mod(L)$.
Let $N, N' \in M \backslash Mod(L)$. Since according to (3.2), $M \backslash Mod(L) \subseteq Mod(Th(M)) = Mod(T)$, it follows from Theorem 4.5 that $Th(\{N\}) = Th(\{N'\})$. Hence, $Th((M \backslash Mod(L)) \cup Mod(L)) = \bigcap \{Th(\{N'\}) : N' \in M \backslash Mod(L)\} \cap Th(Mod(L)) = Th(\{N\}) \cap L = Th(\{N\})$. □

The converse is not true because there may be structures N such that $Th(N)$ is not maximal.

4.2. Adequacy for Propositional Consequence Operations

The definition of propositional consequence operations concerns the connectives of L_{AL}. Hence, if Cn is a propositional consequence operation, we can establish relations between semantics and syntax that involve these connectives.

Theorem 4.7. *Let Cn be a propositional consequence operation and let Mod be Cn-adequate. Then:*

1. *$A \in Cn(T)$ iff $Mod(T \cup \{\neg A\})$ is not Mod-satisfiable.*
2. *Let $A \in T$. A is in T Cn-independent iff $T \backslash \{A\} \cup \{\neg A\}$ is Mod-satisfiable.*
3. *$Mod(A) \cap Mod(\neg A) = Mod(L_{AL})$.*

Proof. (1) follows immediately from the fact that for every propositional consequence operation, it holds that $A \in Cn(T)$ iff $\{\neg A\} \cup T$ is not consistent. (2) is obtained by (1). (3) is obtained by the fact that $\{A, \neg A\}$ is an axiom system for L_{AL} (Lemma 2.18) and Theorem 4.3 part 5. □

Theorem 4.8. *Let Cn be a propositional consequence operation and let Mod be Cn-adequate. Then:*

1. *$Mod(A \wedge B) = Mod(A) \cap Mod(B)$.*
2. *$Mod(A \vee B) \supseteq Mod(A) \cup Mod(B)$.*
3. *$Mod(A \rightarrow B) \supseteq Mod(\neg A) \cup Mod(B)$.*

Proof. For $Mod(A \wedge B) = Mod(A) \cap Mod(B)$:
($\subseteq$) By Lemma 2.18, $A \in Th(Mod(A \wedge B))$ and $B \in Th(Mod(A \wedge B))$. Hence, $Mod(A \wedge B) \subseteq Mod(A)$ and $Mod(A \wedge B) \subseteq Mod(B)$.
($\supseteq$) By Lemma 2.18, $A \wedge B \in Th(Mod(A, B))$, so that $Mod(\{A\} \cup \{B\}) \subseteq Mod(A \wedge B)$. Since $Mod(\{A\} \cup \{B\}) = Mod(A) \cap Mod(B)$, the theorem holds.
For $Mod(A \vee B) \supseteq Mod(A) \vee Mod(B)$:
By Lemma 2.18 (3), $A \vee B \in Th(Mod(A))$ and $A \vee B \in Th(Mod(B))$. Hence, $Mod(A) \subseteq Mod(A \vee B)$ and $Mod(B) \subseteq Mod(A \vee B)$.
For $Mod(A \rightarrow B) \supseteq Mod(\neg A) \cup Mod(B)$:
Since $Mod(\neg A) \cup Mod(B) \subseteq Mod(A \rightarrow B)$ if $A \rightarrow B \in Th(Mod(B))$ and $A \rightarrow B \in Th(Mod(\neg A))$, application of Lemma 2.18 yields the claim. □

It is not generally true that $Mod(A \vee B) \subseteq Mod(A) \cup Mod(B)$. This requires the Negation property. For the probability semantics F_{Mod_3} it does not hold. There are probability functions p and formulas A such that $p(\neg A) = p(A) = \frac{1}{2}$, but for every probability function p it is $p(A \vee \neg A) = 1$.

4.3. Adequacy for Propositional Model Mappings

To obtain familiar connections between semantics and syntax of classical logic, we have to demand that Mod is a propositional model mapping (Definition 3.11). The Negation property $Mod(\neg A) = Mod(A)^c$ is of special importance. This property does not follow from the fact that $Th \circ Mod$ is a propositional consequence operation. However, if we presuppose the Negation property, then Mod is a propositional model mapping is equivalent to $Th \circ Mod$ is a propositional consequence operation.

First, we observe that if a propositional model mapping is adequate for some consequence operation Cn, then Cn is a propositional consequence operation.

Theorem 4.9. *If Mod is a propositional model mapping, then the inference operation $Th \circ Mod$ is a propositional consequence operation. Furthermore, $Th \circ Mod$ is consistent.*

Proof.
- We have $A \in Th(Mod(T))$ iff $Mod(T) \subseteq Mod(A)$ iff $Mod(T) \cap Mod(A)^c = \emptyset$ iff $Mod(T) \cap Mod(\neg A) = \emptyset$ iff $Mod(T \cup \{\neg A\}) = \emptyset$ iff $Th(Mod(T \cup \{\neg A\})) = L_{AL}$.
- $C \in Th(Mod(T, \{A \wedge B\}))$ iff $Mod(T \cup \{A \wedge B\}) \subseteq Mod(C)$ iff $Mod(T) \cap Mod(A \wedge B) \subseteq Mod(C)$ iff $Mod(T) \cap Mod(A) \cap Mod(B) \subseteq Mod(C)$ iff $Mod(T \cup \{A, B\}) \subseteq Mod(C)$ iff $C \in Th(Mod(T, \{A, B\}))$.
- $C \in Th(Mod(T, \{A \vee B\}))$ iff $Mod(T \cup \{A \vee B\}) \subseteq Mod(C)$ iff $Mod(T) \cap Mod(A \vee B) \subseteq Mod(C)$ iff $Mod(T) \cap (Mod(A) \cup Mod(B)) \subseteq Mod(C)$ iff $(Mod(T) \cap Mod(A)) \cup (Mod(T) \cap Mod(B)) \subseteq Mod(C)$ iff $Mod(T \cup \{A\}) \subseteq Mod(C)$ and $Mod(T \cup \{B\}) \subseteq Mod(C)$ iff $C \in Th(Mod(T, \{A\})) \cap Th(Mod(T, \{B\}))$.
- $A \to B \in Th(Mod(T))$ iff $Mod(T) \subseteq Mod(A \to B)$ iff $Mod(T) \subseteq Mod(\neg A) \cup Mod(B)$ iff $Mod(T) \subseteq Mod(A)^c \cup Mod(B)$ iff $Mod(T) \cap Mod(A) \subseteq Mod(B)$ iff $Mod(T \cup \{A\}) \subseteq Mod(B)$ iff $B \in Th(Mod(T, \{A\}))$.
- Suppose that $Th \circ Mod$ is not consistent. Then $A, \neg A \in Th(Mod(\emptyset))$, so that $Mod(\emptyset) \subseteq Mod(A) \cap Mod(A)^c$. Since $Mod(\emptyset) = Str \neq \emptyset$ and $Mod(A) \cap Mod(A)^c = \emptyset$, $Th \circ Mod$ is consistent. □

Remark 2 (Continuation of Remark 1). We can also use the alternative definition of propositional model mappings concerning closure properties of the sets $Th(N)$ (Remark 1). Theorem 4.9 then becomes a special case of Theorem 3.2 (1) of [10]. However, as the proof of this part of the theorem was left to the reader in [10], we proved it above.

Example. Since Mod_1 and Mod_2 are propositional model mappings, by Theorem 4.9, F_{Mod1} and F_{Mod2} are propositional consequence operations. However, Mod_3 is not a propositional model mapping, so that we cannot apply Theorem 4.9 to establish that F_{Mod3} is a propositional consequence operation.

In the presence of the Negation property familiar semantical equivalents of syntactical concepts can be given. Consistency of T is equivalent to $Mod(T) \neq \emptyset$. $Th(N)$ is maximally consistent for every $N \in Str$.

The following theorem requires that $Mod(A) \cap Mod(\neg A) = \emptyset$.

Theorem 4.10. *Suppose that Mod is Cn adequate and let Mod be a propositional model mapping. Then*

1. $Mod(L_{AL}) = \emptyset$.
2. *T is Cn-consistent iff $Mod(T) \neq \emptyset$.*
3. *$Th(M)$ is consistent iff $M \neq \emptyset$.*
4. *T is complete iff for $N, N' \in Mod(T)$: $N \equiv N'$.*

Proof. (1) is a corollary of Theorem 4.3 part 3. For (3): If $M = \emptyset = Mod(L_{AL})$, then $Th(M) = Th(Mod(L_{AL})) = L_{AL}$. If $M \neq \emptyset$, then since $M \subseteq Mod(Th(M))$, $Mod(Th(M)) \neq Mod(L_{AL})$. (4) follows from $Mod(L_{AL}) = \emptyset$ and Theorem 4.5. □

To show that a set is maximally consistent, if and only if it is the theory of a single model, the Negation property is needed. The completeness of $Th(\{N\})$ requires that $Mod(A) \cup Mod(\neg A) = Str$.

Theorem 4.11. *Suppose that Mod is Cn adequate and let Mod be a propositional model mapping. Then*

1. *$Th(\{N\})$ is complete.*
2. *T is maximally consistent iff there exists an $N \in Str$ such that $T = Th(\{N\})$.*

Proof. For (1), let $B \notin Cn(Th(\{N\}))$. Then $B \notin Th(\{N\})$, so that $N \notin Mod(B)$. Then, since $Mod(B) \cup Mod(\neg B) = Str$, $N \in Mod(\neg B)$. Hence, $\neg B \in Th(\{N\})$, so that $\neg B \in Cn(Th(\{N\}))$.

For (2): The direction from left to right is stated in Corollary 4.6. Conversely, it is sufficient to show that $Th(\{N\})$ is a consistent and complete theory. For consistency, observe that since $\{N\} \neq \emptyset$, $Th(\{N\})$ is consistent (Lemma 4.10 (3)). Part (1) of the present theorem states that $Th(\{N\})$ is complete. □

That this semantical formulation is not possible without $Mod(A) \cup Mod(\neg A) = Str$ becomes obvious, if we consider the probability semantics Mod_3. The theory of some probability functions p is not complete and hence not maximally consistent. There are formulas A such that $p(A) \neq 1$ and $p(\neg A) \neq 1$.

The Negation property guarantees that every consistent set can be extended to a maximally consistent set.

Corollary 4.12. *Let Mod be Cn-adequate and T consistent. Then there exists a maximally consistent T' such that $T \subseteq T'$.*

Proof. Let $N \in Mod(T)$. Then $T' = Th(\{N\})$ is maximally consistent. □

The converse of Theorem 4.9 is not true. The fact that $Th \circ Mod$ is a propositional consequence operation does not imply the Negation property: First, if $Th \circ Mod$ is a propositional consequence operation, then $Mod(A) \cap Mod(\neg A) = \emptyset$ is equivalent with $Mod(L_{AL}) = \emptyset$. Yet, the consistency of $Th \circ Mod$ does not imply $Mod(L_{AL}) = \emptyset$. I.e., $Mod(L_{AL}) \neq Str$, does not imply $Mod(L_{AL}) = \emptyset$. Second, the consistency of a propositional $Th \circ Mod$ does not imply $Mod(A) \cup Mod(\neg A) = Str$.

The key tool to invent counterexamples to the converse of Theorem 4.9 is the following lemma which was proposed by one of the anonymous reviewers. It is essentially a reformulation of the fact that every generator set determines the same consequence operation as the corresponding closure system (Lemma 2.11). If the theory of a model is closed under the consequence operation $Th \circ Mod$, this model can be added to the set of structures without changing the consequence operation.

Lemma 4.13. *Let Str, Str' be nonempty sets, and let $\models_{Str}$, $\models_{Str'}$ be satisfaction relations. Let Mod, Mod' and Th, Th' be the corresponding mappings. Then the following statements are equivalent:*

1. *$Th \circ Mod = Th' \circ Mod'$.*
2. *For every $N \in Str' \backslash Str$, $Th(N)$ is closed with respect to $Th \circ Mod$.*

Proof. (1) ⇒ (2) $Th(N)$ is closed with respect to $Th' \circ Mod' = Th \circ Mod$.
(2) ⇒ (1) If $Th(N)$ is closed with respect to $Th \circ Mod$, then there exists an $M \subseteq Str$ such that $Th(M) = Th(N)$. □

Example. Suppose that $Th \circ Mod$ is a propositional consequence operation. The following consequence operations are counterexamples to the converse of Theorem 4.9.

1. The set L is closed under $Th \circ Mod$, so we can add structures N such that $Th(N) = L$, without changing the consequence operation. In this case, $Mod(A) \cap Mod(\neg A) \neq \emptyset$.
2. The set $Th(Mod(\emptyset))$ is closed but in general not complete, so we can add structures N such that $Th(N) = Th(Mod(\emptyset))$ without changing the consequence operation. In this case, $Mod(A) \cup Mod(\neg A) \neq Str$.
3. F_{Mod3} is identical with classical logic and consequently a propositional consequence operation. Every Boolean valuation is a probability function. Moreover, every theory of a probability function $Th(p)$ is closed under classical consequence. According to Lemma 4.13, classical consequence does not change by adding probability functions to the set of all Boolean valuations.

If we presuppose the Negation property $Mod(\neg A) = Mod(A)^c$, the converse of Theorem 4.9 holds.

Theorem 4.14. *If $Th \circ Mod$ is a propositional consequence operation, and Mod satisfies the condition $Mod(\neg A) = Mod(A)^c$, then Mod is a propositional model mapping.*

Proof. According to Lemma 4.8, it remains to show the following:
$Mod(A \vee B) \subseteq Mod(A) \cup Mod(B)$:
Let $N \in Mod(A \vee B)$, and $N \notin Mod(A)$. Then $N \in Mod(\neg A)$. By Lemma 2.18 (3), $B \in Th(Mod(\{A \vee B, \neg A\}))$ and therefore $Mod(\{A \vee B, \neg A\}) = Mod(A \vee B) \cap Mod(\neg A) \subseteq Mod(B)$. Thus, $N \in Mod(B)$.
$Mod(A \to B) \subseteq Mod(\neg A) \cup Mod(B)$:
We have $Mod(A \to B) \subseteq Mod(\neg A) \cup Mod(B)$ iff $Mod(A \to B) \cap Mod(A) \subseteq Mod(B)$ iff $B \in Th(Mod(A \to B, A))$ iff $A \to B \in Th(Mod(A \to B))$. The last statement holds because of reflexivity. □

Remark 3 (Continuation of Remark 1). Theorem 4.14 can be interpreted as a special case of Theorem 3.2 (1) of [10]. An alternative proof of Theorem 4.14 consists of the observation that according to Remark 1, Theorem 4.14 is equivalent to Lemma 2.19.

4.4. Further Relationships between Propositional *Mod* and Propositional *Th* ∘ *Mod*

We have shown that Mod is a propositional model mapping is stronger than $Th \circ Mod$ is a propositional consequence operation. This gives rise to the following questions:

1. Are there weaker conditions for Mod than those of propositional model mappings, such that these are sufficient and necessary for $Th \circ Mod$ being a propositional consequence operation?
2. What additional properties must $Th \circ Mod$ have in order to guarantee that Mod is a propositional model mapping?
3. If Mod is a propositional model mapping, then $Th \circ Mod$ is a propositional consequence operation. What additional properties does $Th \circ Mod$ have?

In response to the first question, clearly, there are such weaker conditions. Since $Th \circ Mod$ is definable in terms of Mod, that $Th \circ Mod$ is a propositional consequence operation can simply be rewritten in terms of Mod. However, the resulting conditions are too weak for establishing important and well-known connections between syntax and semantics of classical logic. Theorems 4.10 and 4.11, for example, do not hold.

The answer to the second question is that there are no such properties. Lemma 4.13 shows that we can add to every set of structures, models of deductively closed sets without changing the consequence operation. Among them are models of the language L_{AL} and models such that their theory is not maximal. It is consequently not possible to express the Negation property $Mod(\neg A) = Mod(A)^c$ in terms of consequence operations. In fact, there are common and important semantics for classical logic which do not satisfy the condition $Mod(\neg A) = Mod(A)^c$. It is, for instance, the inference operation F_{Mod3}, which is generated by probability semantics, identical with classical logic. Yet—as already discussed—there are probability functions p and sentences A such that $p(A) \neq 1$ and $p(\neg A) \neq 1$. Consequently, semantics of classical logic can be done without the Negation property and with models that have no maximal theory.

We give a partial answer to question three. If we assume finitariness, then the induced inference operation $Th \circ Mod$ is classical logic.

Theorem 4.15. *Suppose that Mod is a propositional model mapping and $Th \circ Mod$ is finitary. Then $Th \circ Mod$ is identical with classical logic.*

Proof. That $Th \circ Mod$ is stronger than classical logic follows from the fact that $Th \circ Mod$ is a finitary propositional consequence operation (Theorem 4.9). According to Theorem 2.22, such a consequence operation is complete. Conversely, observe that for every Boolean valuation α, there exists an $N \in Str$ such that $Th(N) = \{A | \models_\alpha A\}$. Define N such that $Th(N) = Th(Mod(\{p \in Av | \models_\alpha p\}))$. Then: $A \in Th(N)$ iff $\models_\alpha A$. This can be seen by induction on the complexity of formulas employing the truth-functionality of Mod (Mod is a propositional model mapping). Observe that N is well defined, since $Th(N)$

is maximally consistent. According to Theorem 2.16, classical logic is stronger than $Th \circ Mod$. □

5. Conclusions and Future Work

Within our framework of adequacy, we have studied the interactions and connections between syntax and semantics. Semantics is based on the Galois correspondence between models and theories, and syntax is given by the theory of consequence operations. A semantics is adequate for a consequence operation Cn, if and only if the semantical inference operation $Th \circ Mod$ is identical with Cn. At first glance, it may be surprising that soundness ($Cn(T) \subseteq Th \circ Mod(T)$ for all T) and completeness ($Th \circ Mod(T) \subseteq Cn(T)$ for all T) are too weak to establish important connections between standard semantics for classical propositional logic and its sound and complete calculi. However, both, that a contradiction has no model, and that a set is maximally consistent, if and only if it is theory of a single model, require the Negation property $Mod(\neg A) = Mod(A)^c$. This property is not implied by the fact that $Cn = Th \circ Mod$ is a propositional consequence operation—although the converse is true. Consequently, there are semantics for classical propositional logic, for instance, probability semantics, that do not satisfy the Negation property.

It seems to be a worthy enterprise to extend the present work to other logics such as first-order logic, higher-order logic and paraconsistent logic. The first—but not simple—task is to define the suitable class of consequence operations for the logic in question. Thereafter, one has to investigate how this class determines properties of its adequate semantics (Sect. 4.2). This leads to representation theorems in the style of Theorems 4.9 and 4.14. The reverse question can also be asked: what properties does a consequence operation have if a certain class of model mappings is adequate for it (Sect. 4.3)? There remains much exploration to be done, how concepts of a certain class of consequence operations connect to concepts of its adequate semantics. Our general results of Sect. 4.1, however, remain valid for every class of consequence operations and model mappings. Although the basic semantical framework employed in this contribution is rather general and widely applicable, it has certain limitations because it is not suitable for nonmonotonic systems.

Acknowledgements

This work was supported by the Austrian Science Fund(FWF):I141-G15 (Project Leader: Gernot Kleiter) as part of the LogICCC programme of the European Science Foundation. Many thanks to three anonymous reviewers for their very detailed, and extremely useful comments. They helped a lot to improve a first version of the paper. I am indebted to Reinhard Kleinknecht for having first introduced me to consequence operations and the notion of adequate model mappings.

References

[1] Barwise, J.K.: Axioms for abstract model theory. Ann. Math. Logic **7**, 221–265 (1974)

[2] Barwise, J.K., Feferman, S. (eds.): Model-Theoretic Logics. Springer, Berlin (1985)

[3] Beziau, J.Y.: La V́eritable Portée Du Théorème de Lindenbaum-Asser. Logique et Analyse, **167**–**168**, 341–359 (1999)

[4] Beziau, J.Y.: From consequence operator to universal logic: a survey of general abstract logic. In: Beziau, J.Y. (ed.) Logica Universalis, pp. 3–19. Birkhäuser, Basel (2007)

[5] Birkhoff, G.: Lattice Theory. American Mathematical Society, Providence (1967)

[6] Cohn, P.M.: Universal Algebra. Harper and Row, New York (1965)

[7] García-Matos, M., Väänänen, J.: Abstract model theory as a framework for universal logic. In: Beziau, J.Y. (ed.) Logica Universalis, pp. 19–33. Birkhäuser, Basel (2007)

[8] Kleinknecht, R.: Lecture Notes Logic 3. University of Salzburg, 2005–2006, unpublished

[9] Lewitzka, S.: a topological approach to universal logic: model-theoretical abstract logics. In: Beziau, J.Y. (ed.) Logica Universalis, pp. 35–61. Birkhäuser, Basel (2007)

[10] Lewitzka, S., Brunner, A.B.M.: Minimally generated abstract logics. Logica Universalis. **3**, 219–241 (2009)

[11] Los, J., Suzko, R.: Remarks on sentential logic. Indag. Mathe. **20**, 177–183 (1958)

[12] Makinson, D.: Bridges from Classical to Nonmonotonic Logic. King's College Publications, London (2005)

[13] Smith, P.: The galois connection between syntax and semantics. http://www.logicmatters.net/resources/pdfs/Galois.pdf, 2010 (Download: September, 3th 2012)

[14] Surma, S.J.: On the axiomatic treatment of the theory of models 1. Universitates Iagellonica Acta Scientarium Litterarumque, Schedae Logicae CCVIII. **4**, 39–45 (1969)

[15] Tarski, A.: On some fundamental concepts of metamathematics. In: Logic, Semantics and Metamathematics (trans: Woodger,J.H), pp. 30–38. Clarendon Press, Oxford (1969)

[16] Tarski, A.: Fundamental concepts of the methodology of the deductive sciences. In: Logic, Semantics and Metamathematics (trans: Woodger,J.H), pp. 60–110. Clarendon Press, Oxford (1969)

[17] Wojcicki, R.: Theory of Logical Calculi. Kluwer Academic Publishers, Dordrecht (1988)

Christian Wallmann
Department of Psychology
University of Salzburg
Hellbrunnerstrasse 34
5020 Salzburg
Austria
e-mail: christian.wallmann2@sbg.ac.at

Received: February 28, 2012.
Accepted: August 1, 2012.