Notre Dame Journal of Formal Logic
Volume 51, Number 3, 2010

Pure Second-Order Logic with Second-Order Identity

Alexander Paseau

Abstract Pure second-order logic is second-order logic without functional or first-order variables. In "Pure Second-Order Logic," Denyer shows that pure second-order logic is compact and that its notion of logical truth is decidable. However, his argument does not extend to pure second-order logic with second-order identity. We give a more general argument, based on elimination of quantifiers, which shows that any formula of pure second-order logic with second-order identity is equivalent to a member of a circumscribed class of formulas. As a corollary, pure second-order logic with second-order identity is compact, its notion of logical truth is decidable, and it satisfies a pure second-order analogue of model completeness. We end by mentioning an extension to *n*th-order pure logics.

1 Introduction

Pure second-order logic is second-order logic without the ability to express first-order quantification. Perhaps its chief interest stems from the fact that, so far as the ability to generalize goes, it is the mirror image of first-order logic. In its standard application, first-order logic captures the logical features of objects and properties when generalization over objects but not properties is allowed; pure second-order logic captures the logical features of objects and properties when generalization over properties but not objects is allowed; and second-order logic combines the two by allowing generalization over both objects and properties. The study of pure second-order logic is thus a natural complement to the study of first- and second-order logic.[1]

I owe the expression 'pure second-order logic' to Denyer [2], but there is a difference between my usage and his. Denyer takes pure second-order logic as second-order logic without first-order variables. In my usage, in contrast, pure second-order

Received October 31, 2008; accepted December 19, 2009; printed June 16, 2010
2010 Mathematics Subject Classification: Primary, 03B15
Keywords: second-order logic, *n*th-order logic, elimination of quantifiers, compactness, decidability of validity, model completeness
 10.1215/00294527-2010-021

logic also lacks functional variables and functional quantification; its only variables and quantifiers are for predicates. My terminological choice stems from the fact that in the presence of a constant, first-order quantification may be simulated by functional quantification: simply construe '$\forall x\varphi(x)$' as '$\forall f\varphi(f(c))$' and '$\exists x\varphi(x)$' as '$\exists f\varphi(f(c))$'.

Denyer [2] shows that in pure second-order logic without second-order identity the notion of logical truth is decidable. His argument is based on the notion of the φ-condensate, $M^{\varphi\text{-cond}}$, of a model M of a sentence φ in this language. This structure takes as its domain the subset of M's domain consisting of the interpretation of the terms in φ, and it interprets other symbols as the restrictions of their M-interpretations to this new domain. Thus,

$$\begin{aligned}
\mathrm{dom}(M^{\varphi\text{-cond}}) &= \{\tau^M : \tau \text{ is a term in } \varphi\},\\
c^{M^{\varphi\text{-cond}}} &= c^M, \text{ for constant } c \text{ in } \varphi,\\
f^{M^{\varphi\text{-cond}}} &= f^M \cap (\mathrm{dom}(M^{\varphi\text{-cond}}))^{k+1} \text{ for } k\text{-adic function symbol } f,\\
R^{M^{\varphi\text{-cond}}} &= R^M \cap (\mathrm{dom}(M^{\varphi\text{-cond}}))^{k} \text{ for } k\text{-adic predicate } R,\\
X^{M^{\varphi\text{-cond}}} &= X^M \cap (\mathrm{dom}(M^{\varphi\text{-cond}}))^{k} \text{ for } k\text{-adic variable } X.
\end{aligned}$$

Dom($M^{\varphi\text{-cond}}$) is nonempty since any sentence φ in this language contains at least one term. Although $M^{\varphi\text{-cond}}$ might not be a realization of the given language because its domain need not be closed under functional application, by extending the interpretation of any function signs if necessary we can turn $M^{\varphi\text{-cond}}$ into a realization. In any case, we can define the notion of quasi satisfaction between $M^{\varphi\text{-cond}}$ and a sentence in the standard way, allowing function symbols to denote partial functions.[2] In fact, Denyer shows by induction on the complexity of φ that M satisfies φ if and only if $M^{\varphi\text{-cond}}$ quasi satisfies φ.[3] Since $M^{\varphi\text{-cond}}$'s domain is finite (because φ is of finite length), $M^{\varphi\text{-cond}}$'s quasi satisfaction of φ is a decidable property. Thus, if φ is not a logical truth, it can be quasi-falsified in a model with domain of size no greater than the number of terms in φ. It follows that the notion of pure second-order logical truth is decidable. As Denyer also shows, the resulting logic is compact.

Our goal is to investigate whether these results still hold if we add second-order identity to pure second-order logic. Given that the logical features of properties and relations include facts about their identity and difference, this is a natural extension; it is natural to suppose that the ability to quantify over predicates goes hand in hand with the ability to state these predicates' identity and distinctness. In the presence of first-order variables and quantifiers, second-order identity may of course be defined. For instance, we may define it as follows for the monadic case:[4]

$$\forall X\forall Y(X = Y \leftrightarrow \forall x(Xx \leftrightarrow Yx)).$$

However, in the absence of first-order or functional quantification second-order identity cannot be defined and must be taken as primitive. Denyer's argument cannot be extended to this more general setting. In fact, the equivalence between M satisfying φ and $M^{\varphi\text{-cond}}$ quasi-satisfying φ no longer holds. For example, the sentence

$$\forall X\forall Y((Xc \wedge Yc) \rightarrow X = Y)$$

of pure second-order logic with second-order identity is false in the model M with domain $\{a, b\}$ and $c^M = a$. But it is quasi-true (indeed true) in its condensate, which has domain $\{a\}$ and also interprets the constant c as a. Moreover, once second-order identity is introduced some sentences in the new language do not contain any terms, and condensates of their models consequently have empty domains.

We shall show that the analogous results are in fact true for pure second-order logic with second-order identity by establishing a quantifier elimination result. This technique is not usually applied to higher-order logics and it will be instructive to apply it here. The results turn out to be corollaries of this more general characterization of the logic.

2 Pure Second-order Logic with Second-order Identity

Our aim is to show that any formula of pure second-order logic with second-order identity is logically equivalent to a basic formula. We assume initially that the language contains no function symbols and later sketch how to extend the result to cover this case. The proof is an adaptation of the method of first-order quantifier elimination for the case at hand; see Chang & Keisler [1, §1.5] for a simpler argument restricted to the first-order case along similar lines. The semantics is the standard one for second-order logic, restricted to the language of pure second-order logic with second-order identity. We begin with some definitions.

2.1 Definition of basic formula We are working in a pure second-order language with second-order identity with signature $\langle (c_i)_{i\in C}, (R_i)_{i\in R}\rangle$. We define basic formulas to be Boolean combinations of atomic formulas and size sentences. The atomic formulas are

$c_i = c_j$	for any constants c_i and c_j,
$R_i(c^k)$	for any k-adic predicate constant R_i and k-tuple of constants c^k,
$X_i(c^k)$	for any k-adic predicate variable X_i and k-tuple of constants c^k,
$R_i = R_j$	for any k-adic predicate constants R_i and R_j,
$X_i = X_j$	for any k-adic predicate variables X_i and X_j,
$R_i = X_j$	for any k-adic predicate constant R_i and predicate variable X_j,
$X_j = R_i$	for any k-adic predicate constant R_i and predicate variable X_j.

The size sentences are the sentences σ_n for $n \geq 1$; that is,

$$\exists X_1 \ldots \exists X_{2^n}(\wedge_{1\leq i\neq j\leq 2^n}\neg X_i = X_j)$$

where the predicate variables are monadic. For $n \geq 1$, σ_n is a formalization of the claim that there are no fewer than 2^n unary properties. Any model in which σ_n is true therefore has a domain of n or more elements. For convenience, we let σ_0 be $\forall X(X = X)$, with X of some arbitrarily chosen adicity.

2.2 Definition of arrangement formula Any given sentence φ of pure second-order logic with second-order identity contains l constants $c_1, \ldots, c_l$, m predicate constants $R_1, \ldots, R_m$ (each of some adicity) and n predicate variables $X_1, \ldots, X_n$ (each of some adicity), where l, m, n are nonnegative integers, at least one of which is positive. Let L_φ be the language of pure second-order logic with second-order identity whose nonlogical vocabulary is the nonlogical vocabulary of φ. Let an L_φ-*arrangement* consist of a set D of size $l + m + n$, an assignment of an element of this set to each constant c_i in φ, an assignment of an element of $\mathbb{P}(D^k)$ to each R_i of adicity k in φ, and an assignment of an element of $\mathbb{P}(D^k)$ to each X_i of adicity k in φ. Given that D is of size $l + m + n$, we can choose distinct assignments for all the constants if desired, and different assignments for all the predicates and predicate variables if desired. We then define an L_φ*-arrangement formula* to be the conjunction of all the atomic or negated atomic facts about the assignments of $c_1, \ldots, c_l$, $R_1, \ldots, R_m$, $X_1, \ldots, X_n$ in an L_φ-arrangement. Note that this conjunction is finite; that it is a formula of L_φ; and that, for each φ, there are finitely many L_φ-arrangements up to isomorphism (since $l+m+n$ is finite) and consequently that there are finitely many L_φ-arrangement formulas.

Intuitively, an L_φ-arrangement formula is a summary of all the facts about an L_φ-arrangement expressible by atomic formulas of the language L_φ and their Boolean combinations. An example will help clarify our definition. Let φ be the sentence

$$\exists X((R = X \leftrightarrow Xc) \vee Xd).$$

In this instance, $l = 2$ and $m = n = 1$. An L_φ-arrangement is then a set D of four elements, with c and d each denoting one of these elements (possibly the same, possibly different) and R and X each denoting subsets of D (possibly the same, possibly different). There are several such arrangements but only a finite number of them up to isomorphism. One such arrangement has the element assigned to c being a member of the subset assigned to X but not R, and has the element assigned to d being a member of neither (so that the elements assigned to c and d are distinct, as are the subsets assigned to X and R). The arrangement formula for this particular arrangement is

$$\neg c = d \wedge \neg Rc \wedge Xc \wedge \neg Rd \wedge \neg Xd \wedge \neg X = R.$$

The arrangement formula captures all the facts about this particular arrangement in this restricted language. The rough idea behind the proof of our main theorem is that existentially quantifying over such a formula has at most the effect of adding a size constraint on the domain.

In connection with this definition, we now prove a lemma.

Lemma 2.1 *Let $\xi_1, \ldots, \xi_n$ be literals of the language L_φ for some formula φ. (A literal is an atomic formula or its negation.) If their conjunction $\wedge_{i\leq n}(\xi_i)$ is satisfiable, it is equivalent to a disjunction of L_φ-arrangement formulas.*

Proof If $\wedge_{i\leq n}(\xi_i)$ is satisfiable then there is some L_φ-arrangement in which it is true. Let $(A_i)_{i\leq N}$ be the finitely many L_φ-arrangement formulas corresponding to the L_φ-arrangements in which $\wedge_{i\leq n}(\xi_i)$ is true. We show that $\wedge_{i\leq n}(\xi_i)$ is equivalent to $\vee_{i\leq N}(A_i)$. Clearly, $\vee_{i\leq N}(A_i)$ entails $\wedge_{i\leq n}(\xi_i)$. Conversely, consider a model in which $\wedge_{i\leq n}(\xi_i)$ is true. Choose an arbitrary interpretation in this model for any constants, predicates, and predicate variables that are in φ but not in $\wedge_{i\leq n}(\xi_i)$, and consider the formula A which is a conjunction of all the L_φ-literals that are true in the resulting interpretation. A is an L_φ-arrangement formula corresponding to an L_φ-arrangement in which $\wedge_{i\leq n}(\xi_i)$ is true; hence $\vee_{i\leq N}(A_i)$ is true in this model since A is a disjunct in $\vee_{i\leq N}(A_i)$. Thus $\wedge_{i\leq n}(\xi_i)$ entails $\vee_{i\leq N}(A_i)$. □

We require one more definition.

Definition 2.2 Let A be an L_φ-arrangement formula. We define $A^{\mathbf{X}_i}$ to be the formula obtained from A by deleting all the conjuncts (if any) containing the variable X_i.

We are now in a position to prove our key lemma.

Lemma 2.3 *Let A be an L_φ-arrangement formula. Then $\exists X_i(A)$ is equivalent to $\sigma_k \wedge A^{\mathbf{X}_i}$ for some size sentence σ_k.*

Proof If X_i does not appear in A, let $\sigma_k = \sigma_0$. If X_i appears in A, let k be the smallest size of the domain of any model of $\exists X_i(A)$. Since A is an L_φ-arrangement formula, it has finite models; hence $\exists X_i(A)$ does too, which shows that k is finite. Clearly, $\exists X_i(A)$ entails $\sigma_k \wedge A^{\mathbf{X}_i}$. For the other direction, note first that we are done if $X_i = R_j$ or $R_j = X_i$ or $X_i = X_j$ or $X_j = X_i$ appears as an unnegated conjunct in A. For then since A is satisfiable, $A^{\mathbf{X}_i}$ entails $\exists X_i(A)$. Thus we may assume that all the identity literals involving X_i in A are negations.

We give the proof for the case in which X_i is of adicity 1. Suppose that $\exists X_i(A)$ contains the constants $c_1, \ldots, c_l$, and that for some nonnegative n, A states that there are n distinct properties (denoted by predicate constants or predicate variables) that agree with X_i on the constants $c_1, \ldots, c_l$ but are all distinct from X_i. Since there is a model of $\exists X_i(A)$ of size k, it follows that $2^{k-l} > n$. Now any model of $\sigma_k \wedge A^{\mathbf{X}_i}$ has a domain with at least k elements. Since $2^{k-l} > n$, any such model's domain has a subset distinct from the n distinct properties that agree with the interpretation of X_i on the interpretation of the constants $c_1, \ldots, c_l$. Hence this model satisfies $\exists X_i(A)$, and so $\sigma_k \wedge A^{\mathbf{X}_i}$ entails $\exists X_i(A)$. The case in which X_i is of adicity m is proved similarly, considering $2^{k^m - l^m}$ instead of its special case 2^{k-l} when $m = 1$. □

With these definitions and lemmas in place, we now show by induction that any formula of the language is logically equivalent to a basic formula. The atomic case is given since atomic formulas are basic, as are the inductive steps involving the Boolean operations, since the set of basic formulas is closed under these operations. It remains to prove that any existentially quantified formula of the language is logically equivalent to a Boolean combination of basic formulas. Consider a formula of the language of the form $\exists X_q(\psi)$ in which we may assume that ψ is a basic formula. Putting ψ in disjunctive normal form, we see that

$$\psi \equiv \vee_{i \leq m}(\wedge_{j \leq n}\xi_{ij} \wedge \Sigma_i),$$

where each ξ_{ij} is a literal and each Σ_i is a Boolean combination of size sentences. Thus, by the first lemma,

$$\psi \equiv \vee_{i \leq m}(\vee_{j \leq k}A_{ij} \wedge \Sigma_i),$$

for some arrangement formulas A_{ij}. (If $(\wedge_{j \leq n}(\xi_{ij})$ is unsatisfiable let Σ_i be $\sigma_1 \wedge \neg\sigma_1$.) Hence, since Σ_i is a closed formula,

$$\begin{aligned} \exists X_q(\psi) &\equiv \exists X_q[\vee_{i \leq m}(\vee_{j \leq k}A_{ij} \wedge \Sigma_i)] \\ &\equiv [\vee_{i \leq m}\exists X_q(\vee_{j \leq k}A_{ij} \wedge \Sigma_i)] \\ &\equiv [\vee_{i \leq m}(\exists X_q \vee_{j \leq k} A_{ij} \wedge \Sigma_i)] \\ &\equiv [\vee_{i \leq m}(\vee_{j \leq k}\exists X_q A_{ij} \wedge \Sigma_i)]. \end{aligned}$$

But, by the second lemma, $\exists X_q A_{ij}$ is equivalent to $\sigma_l \wedge A_{ij}^{\mathbf{X}_q}$ for some σ_l, which is a Boolean combination of basic formulas. Thus we have proved the following theorem.

Theorem 2.4 *Any formula φ of pure second-order logic with second-order identity with constant terms is equivalent to a Boolean combination of atomic formulas of L_φ and size sentences.*

We sketch an extension of the theorem. Let φ be a sentence of pure second-order logic with second-order identity with functional terms (but no functional variables, of course). We may ‘defunctionalize’ φ by uniformly replacing each distinct functional term in φ with a distinct constant term. For example, the formula

$$\forall X[X(f(c_1)) \leftrightarrow (R_1c_1 \vee R_2(g(c_2)) \vee R_3(h(c_3, f(c_1))) \vee R_4(f(c_1)))]$$

may be defunctionalized as

$$\forall X[Xe_1 \leftrightarrow (R_1c_1 \vee R_2e_2 \vee R_3e_3 \vee R_4e_1)],$$

where e_1, e_2, and e_3 are new constants. Let $\varphi(\tau_1, \ldots, \tau_m, d_1, \ldots, d_n)$ be the original formula containing m distinct functional terms $\tau_1, \ldots, \tau_m$ and n distinct constants $d_1, \ldots, d_n$, and suppose that $\varphi(e_1, \ldots, e_m, d_1, \ldots, d_n)$ is a defunctionalization of $\varphi(\tau_1, \ldots, \tau_m, d_1, \ldots, d_n)$, where $e_1, \ldots, e_m, d_1, \ldots, d_n$ are $m + n$ distinct constants. (In the usual terminology: $\varphi(e_1, \ldots, e_m, d_1, \ldots, d_n)$ is $\varphi(\tau_1\backslash e_1, \ldots, \tau_m\backslash e_m, d_1, \ldots, d_n)$.) If $\varphi(e_1, \ldots, e_m, d_1, \ldots, d_n)$ is logically equivalent to $\psi(e_1, \ldots, e_m, d_1, \ldots, d_n)$ then $\varphi(\tau_1, \ldots, \tau_m, d_1, \ldots, d_n)$ is logically equivalent to $\psi(\tau_1, \ldots, \tau_m, d_1, \ldots, d_n)$. We leave the easy inductive proof of this fact to the reader (note the crucial absence of first-order or functional quantifiers and variables in the language). Since any defunctionalized formula φ is logically equivalent to a basic formula ψ, the theorem therefore generalizes to all formulas in the language of pure second-order logic with second-order identity, with the proviso that the class of basic formulas must be expanded to admit functional terms. (This expansion is easily achieved: in the definition of basic formula replace constants c_i, c_j, and c^k with terms τ_i, τ_j, and τ^k in the first three clauses.)

3 Corollaries

The theorem has several corollaries. We draw three here. To begin with, in the first-order setting, we say that a model A is an elementary submodel of B if and only if A is a submodel of B and for any sentence $\varphi(x_1, \ldots, x_k)$ with free variables $x_1, \ldots, x_k$ and k-tuple $a^k \in (\mathrm{Dom}(A))^k$,

$$(A, a^k) \models \varphi(x_1, \ldots, x_k) \quad \text{iff} \quad (B, a^k) \models \varphi(x_1, \ldots, x_k).$$

Consider an analogous property for the general second-order case. Suppose that A is a submodel of B, and that for any formula $\varphi(x_1, \ldots, x_k, X_1, \ldots, X_l)$ with free first-order variables $x_1, \ldots, x_k$ and free second-order predicate variables $X_1, \ldots, X_l$, for any k-tuple a^k of elements of Dom(A) and any l-tuple α^l of respective members of $\mathbb{P}((\mathrm{Dom}\ A)^{n_1}), \ldots, \mathbb{P}((\mathrm{Dom}\ A)^{n_l})$, where n_i is the adicity of X_i, the following holds:

$$(A, a^k, \alpha^l) \models \varphi(x_1, \ldots, x_k, X_1, \ldots, X_l) \quad \text{iff} \quad (B, a^k, \alpha^l) \models \varphi(x_1, \ldots, x_k, X_1, \ldots, X_l).$$

Now fix the logic to be pure second-order logic with second-order identity, so that φ has no first-order variables. In that case we say that A is a *qualitative* submodel of B if and only if the above property holds. Just as the label ‘elementary’ is intended to indicate that A and B agree on the claims they make about their domains’ common elements, the label ‘qualitative’ indicates that A and B agree on the claims they make about their domains’ common qualities (properties and relations).

Suppose then that A is a submodel of B and that they are both models of some complete theory in pure second-order logic with second-order identity. Since A is a submodel of B, the atomic cases in the proof that A is a qualitative submodel of

B are given. The propositional cases are easily proved inductively. Finally, by the theorem, any quantified formula is equivalent to a Boolean combination of atomic formulas and size sentences, and since size sentences are closed formulas they are by assumption satisfied by A if and only if they are satisfied by B, as A and B are both models of the same complete theory. It thus follows that

$$(A, \alpha^l) \models \varphi(X_1, \ldots, X_l) \quad \text{iff} \quad (B, \alpha^l) \models \varphi(X_1, \ldots, X_l).$$

Hence any embedding between the models of a complete theory in this logic is qualitative, or to put it another way, any submodel A of a model B that agrees with B on the sentences of the language is a qualitative submodel of B. This is the pure second-order analogue of model completeness. In this sense, then, all complete theories in pure second-order logic with second-order identity are model-complete.

Second, the theorem implies that logical truth for pure second-order logic with second-order identity is decidable. Let φ be a sentence of the language (for simplicity we assume φ has no functional terms). By the theorem, φ is equivalent to φ^{eq}, a Boolean combination of atomic formulas (without predicate variables) and size sentences, and the proof of the theorem shows that finding this φ^{eq} for any given φ is a mechanical procedure. Suppose φ^{eq} contains l constants $c_1, \ldots, c_l$, m predicate constants $R_1, \ldots, R_m$ and k size sentences, $\sigma_{n_1}, \ldots, \sigma_{n_k}$. Let S (for size) be the largest subscript of any size sentence in φ^{eq}; that is, $S = \max(n_1, \ldots, n_k)$. Then φ^{eq} has a model if and only if it has a model with domain of size $\leq l + 2^m + S$.

The proof is based on three simple facts. First, a Boolean combination of basic formulas of the language and that same formula conjoined or disjoined with a Boolean combination of the size sentences $\sigma_{n_1}, \ldots, \sigma_{n_k}$ have the same models of domain size $\geq S = \max(n_1, \ldots, n_k)$ if and only if they have the same models of domain size S. The reason is that the sentence σ_{n_i} only distinguishes between two categories of models: those of domain size $\geq n_i$ and those of domain size $< n_i$. Second, the distinctness of the interpretations of any two constants may be ensured if necessary in any domain of size $\geq l$. Third, the distinctness and identity of the interpretations of the predicate constants $R_1, \ldots, R_m$ in a model M may be ensured by the presence of 2^m elements in the domain, since there are at most 2^m different regions of the form $\pm R_1^M \cap \cdots \cap \pm R_m^M$ (where $+R_i^M$ is R_i^M and $-R_i^M$ is $(\neg R_i)^M$).[5]

Now for any domain of given finite size there are finitely many ways to interpret the nonlogical vocabulary of φ^{eq}. Hence there are only finitely many potential models (up to isomorphism) of φ^{eq} of domain size $\leq l + 2^m + S$. If φ is a logical truth, it will be true in all these models; if φ is not a logical truth, one of these models will be a model of $\neg\varphi^{\mathrm{eq}}$ and thus of $\neg\varphi$. This provides a decision procedure for logical truth in this language.

Third, the theorem and its proof imply that pure second order-logic with second-order identity is compact. We give two proofs of this fact, first an elegant proof then a proof that generalizes. Suppose some set of sentences Γ in pure second-order logic with second-order identity is unsatisfiable. Each member γ of Γ is a sentence and thus by the theorem is equivalent to a Boolean combination of sentences of the form $c_i = c_j$, $R_i(c^k)$, $R_i = R_j$, σ_n. Let us call a formula that is both a basic formula and a sentence a basic sentence. The trick is to notice that basic sentences may be "first-orderized." Sentences of the form $c_i = c_j$ are already first-order; any sentence of the form $R_i = R_j$ where $R_i = R_j$ are k-adic predicate constants can be replaced by $\forall x^k(R_i(x^k) \leftrightarrow R_j(x^k))$, where x^k is a k-tuple of distinct first-order variables; and any sentence σ_n can be replaced by its first-order equivalent: $\exists x_1 \ldots \exists x_n(\wedge_{1 \leq i \neq j \leq n} \neg x_i = x_j)$. ($\sigma_0$ may be replaced with $\forall x(x = x)$.) Let γ^* be the first-orderization of a basic sentence γ, and for any model M in pure second-order logic with second-order identity of a basic sentence, let M^* be the associated first-order model, that is, the model with the same domain as M and which interprets the constants and predicate constants in the same way as M. It is easy to verify by induction on formula complexity that if γ is a basic sentence then M is a model for γ if and only if M^* is a model for γ^*. For example, if γ is the sentence '$R_1 = R_2 \wedge R_1 c$' and M is the model in pure second-order logic with second-order identity that satisfies γ with domain $\{1, 2\}$, $R_1^M = R_2^M = \{1\}$ and $c^M = 1$, then M^* is the first-order model with domain $\{1, 2\}$, $R_1^{M^*} = R_2^{M^*} = \{1\}$ and $c^{M^*} = 1$, and M^* satisfies γ^*, which is '$\forall x(R_1(x) \leftrightarrow R_2(x)) \wedge R_1 c$'. Returning to Γ, since by assumption Γ has no (pure second-order logic with second-order identity) model, Γ^* has no (first-order) model. Since first-order logic is compact, it follows that some finite subset of Γ^*, $(\Gamma^*)^{\text{fin}}$, is unsatisfiable in first-order logic. The finite subset of Γ that corresponds to $(\Gamma^*)^{\text{fin}}$ is correspondingly unsatisfiable in pure second-order logic with second-order identity. Hence the logic is compact.

For the second proof of compactness, recall the Henkin-style compactness proof of first-order logic. Any set of first-order sentences Δ all of whose finite subsets are satisfiable can be extended to a maximal set $\Delta^{\max}$ with this property. Given its maximality, $\Delta^{\max}$ is witness-complete (in a possibly augmented language). We may now use $\Delta^{\max}$ to define a familiar term model $M(\Delta^{\max})$. The salient facts about $M(\Delta^{\max})$ are that its domain's elements are equivalence classes of constants, with constants c_i and c_j in the same equivalence class if and only if $c_i = c_j \in \Delta^{\max}$, that $R_i(c^k)$ is true in $M(\Delta^{\max})$ if and only if $R_i(c^k) \in \Delta^{\max}$, and that $\exists x\varphi(x)$ is true in $M(\Delta^{\max})$ if and only if $\varphi(c) \in \Delta^{\max}$ for some constant c.

We can replicate this proof for pure second-order logic with second-order identity. Let Δ be a finitely satisfiable set of sentences of this logic, and let Δ^{eq} be the set of basic sentence equivalents of the members of Δ; that is, $\Delta^{\text{eq}} = \{\varphi^{\text{eq}} : \varphi \in \Delta\}$.[6] Clearly, Δ^{eq} is finitely satisfiable. As above, we extend Δ^{eq} to a maximal finitely satisfiable set, $(\Delta^{\text{eq}})^{\max}$, which is witness-complete. Witness-completeness here consists of two conditions: if $\exists X\varphi(X) \in (\Delta^{\text{eq}})^{\max}$ then $\varphi(R_i) \in (\Delta^{\text{eq}})^{\max}$ for some predicate constant R_i (of the same adicity as the predicate variable X); and if $R_i \neq R_j$ then for some k-tuple of constants c^k (where k is the adicity of R_i and R_j) either $R_i c^k \in (\Delta^{\text{eq}})^{\max}$ and $\neg R_j c^k \in (\Delta^{\text{eq}})^{\max}$, or $R_j c^k \in (\Delta^{\text{eq}})^{\max}$ and $\neg R_i c^k \in (\Delta^{\text{eq}})^{\max}$. We then define a pure second-order model $M(\Delta^{\text{eq}})^{\max}$ based on $(\Delta^{\text{eq}})^{\max}$ in the same way as in the first-order case: the domain's elements are equivalence classes of constants, constants c_i and c_j are in the same equivalence class if and only if $c_i = c_j \in (\Delta^{\text{eq}})^{\max}$ and $R_i(c^k)$ is true in $M(\Delta^{\text{eq}})^{\max}$ if and only if $R_i(c^k) \in (\Delta^{\text{eq}})^{\max}$. Unlike the first-order case, however, the resulting pure second-order model need not be a model of $(\Delta^{\text{eq}})^{\max}$. For example, $\neg\exists X\varphi(X)$ and thus all sentences of the form $\neg\varphi(R_i)$ may be in $(\Delta^{\text{eq}})^{\max}$, but $\exists X\varphi(X)$ may nevertheless be true in $M(\Delta^{\text{eq}})^{\max}$ if some indefinable—in the augmented language—k-ary property of the domain satisfies $\varphi(X)$. This is why the Henkin compactness proof does not work for standard second-order logic, which is not compact. But it does work for the case for pure second-order logic with second-order identity, since as we shall now prove the following biconditional obtains for any basic sentence φ:

$$\varphi \in (\Delta^{\mathrm{eq}})^{\max} \quad \text{iff} \quad M(\Delta^{\mathrm{eq}})^{\max} \models \varphi.$$

First note that $(\Delta^{\mathrm{eq}})^{\max}$, being a maximal finitely satisfiable set, has the familiar maximality properties: for any φ, either φ or $\neg\varphi$ but not both are in $(\Delta^{\mathrm{eq}})^{\max}$, $\varphi_1 \wedge \varphi_2 \in (\Delta^{\mathrm{eq}})^{\max}$ if and only if $\varphi_1 \in (\Delta^{\mathrm{eq}})^{\max}$ and $\varphi_2 \in (\Delta^{\mathrm{eq}})^{\max}$, and so on. We therefore need only consider four cases: (i) φ is $c_i = c_j$; (ii) φ is $R_i(c^k)$; (iii) φ is $R_i = R_j$; and (iv) φ is σ_n. All other basic sentences are Boolean compounds of these and their satisfaction of the biconditional follows inductively from the atomic cases by the maximality properties of $(\Delta^{\mathrm{eq}})^{\max}$. Cases (i) and (ii) are given by the definition of $M(\Delta^{\mathrm{eq}})^{\max}$. To prove case (iii), suppose that φ is $R_i = R_j$ and that $R_i = R_j \in (\Delta^{\mathrm{eq}})^{\max}$. If $M(\Delta^{\mathrm{eq}})^{\max} \models \neg R_i = R_j$ then without loss of generality some element of $(\mathrm{dom}(M(\Delta^{\mathrm{eq}})^{\max}))^k$ is in the extension of the interpretation of R_i but not of R_j (where k is the adicity of R_i and R_j), so that $R_i(c^k) \in (\Delta^{\mathrm{eq}})^{\max}$ and $R_j(c^k) \notin (\Delta^{\mathrm{eq}})^{\max}$, hence $\neg R_j(c^k) \in (\Delta^{\mathrm{eq}})^{\max}$. But this is a contradiction, since all the finite subsets of $(\Delta^{\mathrm{eq}})^{\max}$ are satisfiable. Conversely, if $M(\Delta^{\mathrm{eq}})^{\max} \models R_i = R_j$ then $\neg R_i = R_j \notin (\Delta^{\mathrm{eq}})^{\max}$ by the witness-completeness of $(\Delta^{\mathrm{eq}})^{\max}$, and so by the maximality of $M(\Delta^{\mathrm{eq}})^{\max}$, $R_i = R_j \in (\Delta^{\mathrm{eq}})^{\max}$. For the final case (iv), suppose φ is σ_n. If $\sigma_n \in (\Delta^{\mathrm{eq}})^{\max}$ then the first witness-completeness property of $(\Delta^{\mathrm{eq}})^{\max}$ ensures that there are 2^n monadic predicates $R_1, \ldots, R_{2^n}$ such that $(\wedge_{1 \leq i \neq j \leq 2^n} \neg R_i = R_j) \in (\Delta^{\mathrm{eq}})^{\max}$, and by the second witness-completeness property there are at least n constants $c_1, \ldots, c_n$ such that $({}_{1 \leq i \neq j \leq n} \neg c_i = c_j) \in (\Delta^{\mathrm{eq}})^{\max}$. By the definition of $M(\Delta^{\mathrm{eq}})^{\max}$, this model's domain therefore has size $\geq n$ and so $M(\Delta^{\mathrm{eq}})^{\max} \models \sigma_n$. Conversely, if $M(\Delta^{\mathrm{eq}})^{\max} \models \sigma_n$ then there are (at least) n constants in the language such that $\neg c_i = c_j \in (\Delta^{\mathrm{eq}})^{\max}$ for any $i \neq j$ and so $\neg\sigma_n \notin (\Delta^{\mathrm{eq}})^{\max}$; otherwise, $(\Delta^{\mathrm{eq}})^{\max}$ would not be finitely satisfiable. By the maximality of $M(\Delta^{\mathrm{eq}})^{\max}$ it follows that $\sigma_n \in (\Delta^{\mathrm{eq}})^{\max}$.

This proves that $M(\Delta^{\mathrm{eq}})^{\max}$ is a model of Δ^{eq}, whether or not $M(\Delta^{\mathrm{eq}})^{\max}$ is a model of $(\Delta^{\mathrm{eq}})^{\max}$. It follows that Δ^{eq} is satisfiable, which implies that the original set of sentences Δ is satisfiable too. Pure second-order logic with second-order identity is therefore compact.

4 An Extension

We conclude with a brief remark about pure nth-order logics with pure nth-order identity. By this we mean an nth-order logic whose only variables and quantifiers are nth-order predicate ones. In other words, a pure nth-order logic is an nth-order logic for some $n \geq 2$ without quantifiers or variables of order m for any $m < n$ and without functional variables or quantifiers.[7] We foresee no difficulty in extending our results to pure nth-order logic with nth-order identity. One result whose proof obviously does not extend to the pure nth-order case is our first proof of compactness. The first proof of compactness for pure second-order logic with second-order identity relies on the fact that the first-order fragment of second-order logic is compact. This proof only generalizes to the nth-order case on the assumption that the fragment of nth-order logic with only $(n-1)$th-order quantifiers is compact. In contrast, the second proof of compactness lends itself to direct generalization to this case, assuming our main theorem's generalization.

Notes

1. By "properties" we mean properties and relations of any arity.

2. In the absence of first-order quantification any functionally incomplete $M^{\varphi\text{-cond}}$ can no longer be discriminated from M by the latter's satisfaction of the sentence '$\forall x^k \exists y(f(x^k) = y)$', where x^k is a k-tuple.

3. Denyer does not distinguish between satisfaction and quasi satisfaction but it is clearest to do so.

4. Principles of this kind are often called "Extensionality."

5. Under most circumstances, for instance, if some of the predicate constants $R_1, \ldots, R_m$ are of adicity greater than 1, and in particular if some of the different predicate constants are of different adicity, this is overkill.

6. We may pick a single φ^{eq} for each φ if we like.

7. Let mth-order generality mean the presence of mth-order predicate variables and quantifiers. Any nth-order logic that admits mth-order generality for $m < n$ can express the fact that a dyadic predicate R^{m+1} has an infinite chain:

$$\forall x^m \exists y^m (R^{m+1} x^m y^m) \wedge \neg \exists x^m (R^{m+1} x^m x^m) \\ \wedge \forall x^m \forall y^m \forall z^m ((R^{m+1} x^m y^m \wedge R^{m+1} y^m z^m) \rightarrow R x^m z^m)).$$

(Making the appropriate modifications, any predicate R^{m+1} of adicity ≥ 2 will do instead of a dyadic predicate.) Note also that if an nth-order logic admits both mth-order and $(m+1)$th-order generality for some $m < n$ then it has the ability to express the version of the above sentence with R^{m+1} existentially quantified over. The generalization of our main theorem's proof will therefore not hold for such nth-order logics, since they have sentences with no finite domains. Also, any nth-order logic with mth-order generality for $m < n$ has a fragment which may be thought of as first-order logic moved up $m - 1$ levels. The natural generalization of pure second-order logic is thus nth-order logic with only nth-order predicate variables and quantifiers.

References

[1] Chang, C. C., and H. J. Keisler, *Model Theory*, 3d edition, vol. 73 of *Studies in Logic and the Foundations of Mathematics*, North-Holland Publishing Co., Amsterdam, 1990. Zbl 0697.03022. MR 1059055. 353

[2] Denyer, N., "Pure second-order logic," *Notre Dame Journal of Formal Logic*, vol. 33 (1992), pp. 220–24. Zbl 0760.03001. MR 1167978. 351, 352

Acknowledgments

Thanks to Nick Denyer and an NDJFL referee for comments.

Wadham College
Oxford OX1 3PN
UNITED KINGDOM
alexander.paseau@philosophy.ox.ac.uk