Model-Theoretic Forcing in Transition Algebra

Go Hashimoto
Kyushu University, Fukuoka, Japan

Daniel Găină
Kyushu University, Fukuoka, Japan

Abstract

We study Löwenheim-Skolem and Omitting Types theorems in Transition Algebra, a logical system obtained by enhancing many sorted first-order logic with features from dynamic logic. The sentences we consider include compositions, unions, and transitive closures of transition relations, which are treated similarly to actions in dynamic logics to define necessity and possibility operators. We show that Upward Löwenheim-Skolem theorem, any form of compactness, and joint Robinson consistency property fail due to the expressivity of transitive closures of transitions. In this non-compact many-sorted logical system, we develop a forcing technique method by generalizing the classical method of forcing used by Keisler to prove Omitting Types theorem. Instead of working within a single signature, we work with a directed diagram of signatures, which allows us to establish Downward Löwenheim-Skolem and Omitting Types theorems despite the fact that models interpret sorts as sets, possibly empty. Building on a complete system of proof rules for Transition Algebra, we extend it with additional proof rules to reason about constructor-based and/or finite transition algebras. We then establish the completeness of this extended system for a fragment of Transition Algebra obtained by restricting models to constructor-based and/or finite transition algebras.

2012 ACM Subject Classification Theory of computation → Logic and verification

Keywords and phrases institutional model theory, algebraic specification, transition algebra, forcing, omitting types property, Löwenheim-Skolem properties, completeness

Digital Object Identifier 10.4230/LIPIcs.MFCS.2025.55

Related Version *Full paper, including proofs*: `http://arxiv.org/abs/2506.22828` [20]

Funding The work presented in this paper has been partially supported by Japan Society for the Promotion of Science, grant number 23K11048.

1 Introduction

Transition Algebra [19], abbreviated TA, offers a logical framework for modeling and reasoning about state transitions in systems, particularly in concurrency theory and computer science. Closely related to modal logics and algebraic specification languages executable by rewriting, it provides a structured approach to describing system evolution based on predefined rules. The key concepts in transition algebra are (a) states, which are characterized using equational theories, and (b) transitions, which are represented as labeled relations between states. Transition algebra can be broadly viewed as an extension of many-sorted first-order logic, incorporating features from dynamic logic. It includes algebraic operators that define how transitions combine, such as (a) Sequential composition ($\fatsemi$): if a system moves from state t_1 to t_2 according to transition rule $\mathfrak{a}_1$, and then from t_1 to t_2 under transition rule $\mathfrak{a}_2$, this is written as $\mathfrak{a}_1 \fatsemi \mathfrak{a}_2$, (b) Choice ($\cup$): represents nondeterministic choices between transitions. (c) Iteration ($*$): denotes repeated transitions, analogous to the Kleene star in formal language theory. Iteration ($*$), has been employed in the literature to enhance the expressive power of first-order logic, albeit at the expense of compactness. Notably, Fagin's work on Transitive Closure Logic (TCL) has been extensively studied within the frameworks of finite model theory and descriptive complexity [9, 10]. However, apart from the work on TA [19], we are

© Go Hashimoto and Daniel Găină;
licensed under Creative Commons License CC-BY 4.0
50th International Symposium on Mathematical Foundations of Computer Science (MFCS 2025).
Editors: Paweł Gawrychowski, Filip Mazowiecki, and Michał Skrzypczak; Article No. 55; pp. 55:1–55:18
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

not aware of any model-theoretic studies of first-order logic with iteration beyond the scope of finite model theory. Moreover, when sorts – possibly infinitely many – and models interpreting those sorts as sets – possibly empty – are permitted, extending existing techniques to such a framework poses significant challenges and lacks a straightforward solution.

In [19], it is shown that transition algebra possesses an essential property for modularization, known as the *satisfaction condition*. This condition states that truth remains invariant under changes in notation. Here, a change in notation refers to the translation between local languages constructed from vocabularies, often called signatures. Consequently, transition algebra is formalized as an institution [11], a category-theoretic framework that captures the notion of logic by incorporating its syntax, semantics, and the satisfaction relation between them. Similarly, the logic underlying CafeOBJ [8] satisfies the satisfaction condition for signature morphisms, allowing it to be formalized as an institution. However, its models cannot distinguish between different transitions occurring between the same states, as transitions are represented by preorder relations. In contrast, Rewriting Logic [25] – the foundation of Maude [4] – offers greater expressivity by treating transitions as arrows in a category, where states serve as objects. This expressiveness comes at a cost, as the satisfaction condition does not hold in Rewriting Logic. On the other hand, transition algebra retains the modular properties of the preorder algebras in CafeOBJ due to its institutional structure while achieving the same level of expressivity as Rewriting Logic.

Given a set of properties formalized as first-order sentences, practitioners seek to study a restricted class of transition algebras that satisfy these properties. In fields such as formal methods and functional programming – where computation and reasoning rely on abstract data types and term rewriting – many algebraic structures are naturally described using constructor operators, such as numbers with 0 and succ, or lists with nil and cons. Pattern matching on constructors further simplifies function definitions. A related example is Finite Model Theory [24, 12], which focuses on logical structures with a finite domain. This area is particularly relevant to database theory, complexity theory, and various computer science applications. Building on a complete system of proof rules for TA, such as the one proposed in [19], we extend it with additional proof rules to reason about constructor-based and/or finite transition algebras. We refer the reader to [2, 1, 17] for theoretical foundations of constructor-based models and to [27] for a formal method developed for reasoning about such models. We then establish the completeness of this extended system for the fragment of TA obtained by restricting the semantics to constructor-based and/or finite transition algebras.

In this work, we use Omitting Types Theorem (OTT) to extend completeness of TA to a fragment obtained by restricting the semantics to constructor-based transition algebras with finite carrier sets. The OTT states that if a first-order theory Φ does not require a certain type (a set of formulas describing a potential element) to be realized, then there exists a model of Φ that omits this type. This ensures the existence of models that exclude specific unwanted elements. A key challenge arises from the fact that models with potentially empty carrier sets are allowed in TA. In this model-theoretic framework, extending the initial signature with an infinite number of constants cannot always be achieved while preserving the satisfiability of the underlying theory. To address this, we employ a forcing technique introduced in [19] to prove completeness of TA. Forcing was originally introduced by Paul Cohen [5, 6] in set theory to show the independence of the continuum hypothesis from the other axioms of Zermelo-Fraenkel set theory. Robinson [28] developed an analogous theory of forcing in model theory. The forcing technique has been studied in institutional model theory at an abstract, category-theoretic level, in works such as [18, 14, 15]. The forcing method proposed in [19] generalizes classical forcing from a single signature to a category of signatures, enabling the dynamic addition of both constants and sentences while preserving satisfiability. Notably, this dynamic forcing method is particularly suited for logical systems that are not compact, such as TA.

Additionally, we show that TA theories have some control over the cardinality of their models. The Upward Löwenheim–Skolem Theorem does not hold in general, as there exist theories that uniquely determine a model up to isomorphism, provided that the model's size is exactly the cardinality of the power of its underlying signature.[1] It is interesting to note that the Downward Löwenheim–Skolem Theorem does hold, as for any model whose size exceeds the power of its underlying signature, one can construct an elementary submodel whose size matches the power of the signature.

2 Transition algebra

In this section, we recall *the logic of many-sorted transition algebras* [19], or *transition algebra* (TA), for short. We present, in order: signatures, models, sentences, and the TA satisfaction relation.

Signatures. The signatures we consider are ordinary algebraic signatures endowed with polymorphic transition labels. We denote them by triples of the form (S, F, L), where:

- (S, F) is a many-sorted algebraic signature consisting of a set of *sorts* S and a set of *function symbols* $F = \{\sigma : w \to s \mid w \in S^*$ and $s \in S\}$;
- L is a set whose elements we call *transition labels*.

Given a function symbol $\sigma : w \to s \in F$, we refer to $w \in S^*$ as its *arity* and $s \in S$ as its *sort*. When w is the empty arity, we may speak of $\sigma\colon \to s$ as a *constant* of sort s.

Throughout the paper, we let Σ, Σ', and Σ_i range over signatures of the form (S, F, L), (S', F', L'), and (S_i, F_i, L_i), respectively.

As usual in institution theory [7, 29], important constructions such as signature extensions with constants as well as open formulae and quantifiers are realized in a multi-signature setting, so moving between signatures is common. A *signature morphism* $\chi\colon \Sigma \to \Sigma'$ consists of a triple $\chi = (\chi^{st}, \chi^{op}, \chi^{lb})$, where (a) $\chi^{st} : S \to S'$ is a function mapping sorts, (b) $\chi^{op} : F \to F'$ maps each function symbol $\sigma : s_1 \dots s_n \to s \in F$ to a function symbol $\chi^{op}(\sigma) : \chi^{st}(s_1) \dots \chi^{st}(s_n) \to \chi(s) \in F'$ and (c) $\chi^{lb} : L \to L'$ is a function mapping labels. For convenience, we typically omit the superscripts *st*, *op* and *lb* in the notation.

▶ Remark 1. Signature morphisms compose componentwise. This composition is associative and has identities, forming a category Sig of signatures.

Models. Given a signature Σ, a Σ-*model* $\mathfrak{A}$ consists of:

- an (S, F)-algebra $\mathfrak{A}$, where each sort $s \in S$ is interpreted by $\mathfrak{A}$ as a set $\mathfrak{A}_s$ and each function symbol $\sigma : s_1 \dots s_n \to s \in F$ is interpreted by $\mathfrak{A}$ as a function $\sigma^{\mathfrak{A}} : \mathfrak{A}_{s_1} \times \dots \times \mathfrak{A}_{s_n} \to \mathfrak{A}_s$;
- an interpretation of each label $\varrho \in L$ as a *many-sorted transition relation* $\varrho^{\mathfrak{A}} \subseteq \mathfrak{A} \times \mathfrak{A}$, where $\varrho^{\mathfrak{A}} = \{\varrho^{\mathfrak{A}}_s\}_{s \in S}$ and $\varrho^{\mathfrak{A}}_s \subseteq \mathfrak{A}_s \times \mathfrak{A}_s$ for all sorts $s \in S$. [2]

[1] The power of a signature is the cardinality of the set of all sentences that can be formed in its language.

[2] In principle, relations of any arity could be defined, and the results presented in this paper would still hold. However, in algebraic specification languages that are executable via rewriting, the only relations considered are transitions, which are defined between pairs of elements of the same sort. Furthermore, the iteration operator $(*)$ can be applied exclusively to transitions.

A *homomorphism* $h : \mathfrak{A} \to \mathfrak{B}$ over a signature Σ is an algebraic (S,F)-homomorphism that preserves transitions: $h(\varrho^{\mathfrak{A}}) \subseteq \varrho^{\mathfrak{B}}$ for all $\varrho \in L$. It is easy to see that Σ-homomorphisms form a category, which we denote by $\mathsf{Mod}^{\mathsf{TA}}(\Sigma)$, under their obvious componentwise composition.

▶ Remark 2. Every signature morphism $\chi\colon \Sigma \to \Sigma'$ determines a *model-reduct functor* $\_\!\upharpoonright_{\chi}\colon \mathsf{Mod}^{\mathsf{TA}}(\Sigma') \to \mathsf{Mod}^{\mathsf{TA}}(\Sigma)$ such that:

- for every Σ'-model $\mathfrak{A}'$, $(\mathfrak{A}'\!\upharpoonright_{\chi})_s = \mathfrak{A}'_{\chi(s)}$ for each sort $s \in S$, $\sigma^{(\mathfrak{A}'\upharpoonright_{\chi})} = \chi(\sigma)^{\mathfrak{A}'}$ for each symbol $\sigma \in F$, and $\varrho^{(\mathfrak{A}'\upharpoonright_{\chi})} = \chi(\varrho)^{\mathfrak{A}'}$ for each label $\varrho \in L$; and
- for every Σ'-homomorphism $h'\colon \mathfrak{A}' \to \mathfrak{B}'$, $(h'\!\upharpoonright_{\chi})_s = h'_{\chi(s)}$ for each $s \in S$.

Moreover, the mapping $\chi \mapsto \_\!\upharpoonright_{\chi}$ is functorial.

For any signature morphism $\chi : \Sigma \to \Sigma'$, any Σ-model $\mathfrak{A}$ and any Σ'-model $\mathfrak{A}'$ if $\mathfrak{A} = \mathfrak{A}'\!\upharpoonright_{\chi}$, we say that $\mathfrak{A}$ is the χ-*reduct* of $\mathfrak{A}'$, and that $\mathfrak{A}'$ is a χ-*expansion* of $\mathfrak{A}$. For example, for a many-sorted set X (say, of variables) that is disjoint from the set of constants in Σ, consider the inclusion morphism $\iota_X : \Sigma \hookrightarrow \Sigma[X]$, where $\Sigma[X] = (S, F[X], L)$ is the signature obtained from Σ by adding the elements of X to F as new constants of appropriate sort. Then an expansion of a Σ-model $\mathfrak{A}$ along ι_X can be seen as a pair $\mathfrak{A}' = \langle \mathfrak{A}, f : X \to \mathfrak{A}\rangle$, where f is a valuation of X in $\mathfrak{A}$.

As in many-sorted algebra, there is a special, initial model in $\mathsf{Mod}^{\mathsf{TA}}(\Sigma)$, which we denote by T_Σ, whose elements are ground terms built from function symbols, and whose transitions are all empty. The Σ-model $T_\Sigma(X)$ of terms with variables from X is defined as the ι_X-reduct of $T_{\Sigma[X]}$; i.e., $T_\Sigma(X) = T_{\Sigma[X]}\!\upharpoonright_{\iota_X}$.

Sentences. Given a signature Σ, the set A of *actions* is defined by the following grammar:

$$\mathfrak{a} \;::=\; \varrho \;\mid\; \mathfrak{a} \fatsemi \mathfrak{a} \;\mid\; \mathfrak{a} \cup \mathfrak{a} \;\mid\; \mathfrak{a}^*$$

where ϱ is a transition label. We let A denote the set of all actions and extend our notational convention for signature components to their corresponding sets of actions. Specifically, we use: (a) A' to denote the set of actions over a signature Σ', (b) A_i to denote the set of actions over a signature Σ_i, and similarly for other variations. Moreover, through a slight abuse of notation, we also denote by $\chi\colon A \to A'$ the canonical map determined by a signature morphism $\chi\colon \Sigma \to \Sigma'$.

To define sentences, we assume a countably infinite set of *variable names* $\{v_i \mid i < \omega\}$. A *variable* for a signature Σ is a triple $\langle v_i, s, \Sigma\rangle$, where (a) v_i is a variable name and (b) s is a sort in Σ. The third component is used only to ensure that variables are distinct from the constants declared in Σ, which is essential when dealing with quantifiers. The set $\mathsf{Sen}^{\mathsf{TA}}(\Sigma)$ of *sentences* over Σ is given by the following grammar:

$$\phi ::= t = t' \mid \mathfrak{a}(t_1, t_2) \mid \neg\phi \mid \vee\Phi \mid \exists X \cdot \phi'$$

where (a) t and t' are Σ-terms of the same sort; (b) $\mathfrak{a}$ is any action; (c) t_1 and t_2 are Σ-terms of the same sort; (d) Φ is a finite set of Σ-sentences; and (e) X is a finite block of Σ-variables, that is, X is a finite set of variables such that if $\langle v_i, s_1, \Sigma\rangle, \langle v_j, s_2, \Sigma\rangle \in X$ and $s_1 \neq s_2$ then $i \neq j$; and (f) ϕ' is a $\Sigma[X]$-sentence. A sentence $\mathfrak{a}(t_1, t_2)$ is called a *transition rule*, which can also be written in infix notation $t_1 \; \mathfrak{a} \; t_2$. An *atomic sentence* is a ground equation $t = t'$ or a ground transition rule of the form $t_1 \; \varrho \; t_2$, where $\varrho \in L$ is a transition label.

Besides the above core connectives, we also make use of the following convenient (and standard) abbreviations: $\wedge\Phi := \neg \vee_{\phi\in\Phi} \neg\phi$ for finite conjunctions; $\bot := \vee\emptyset$ for falsity; $\top := \wedge\emptyset = \neg\bot$ for truth; $\phi_1 \Rightarrow \phi_2 := \neg\phi_1 \vee \phi_2$ for implications; and $\forall X \cdot \phi' := \neg\exists X \cdot \neg\phi'$ for universally quantified sentences.

▶ Remark 3. Any signature morphism $\chi\colon \Sigma \to \Sigma'$ can be canonically extended to a *sentence-translation function* $\chi\colon \mathsf{Sen}^{\mathsf{TA}}(\Sigma) \to \mathsf{Sen}^{\mathsf{TA}}(\Sigma')$ given by:

- $\chi(t = t') \coloneqq (\chi(t) = \chi(t'))$;
- $\chi(t_1\ \mathfrak{a}\ t_2) \coloneqq \chi(t_1)\ \chi(\mathfrak{a})\ \chi(t_2)$;
- $\chi(\neg\phi) \coloneqq \neg\chi(\phi)$;
- $\chi(\vee\Phi) \coloneqq \vee\chi(\Phi)$; and
- $\chi(\exists X \cdot \phi') \coloneqq \exists X' \cdot \chi'(\phi')$, where $X' = \{\langle v_i, \chi(s), \Sigma'\rangle \mid \langle v_i, s, \Sigma\rangle \in X\}$ and $\chi' : \Sigma[X] \to \Sigma'[X']$ is the extension of $\chi : \Sigma \to \Sigma'$ mapping each $\langle v_i, s, \Sigma\rangle \in X$ to $\langle v_i, \chi(s), \Sigma'\rangle \in X'$.

Moreover, this sentence-translation mapping is functorial in χ.

For the sake of simplicity, we identify variables only by their name and sort, provided that there is no danger of confusion. Using this convention, each inclusion morphism $\iota\colon \Sigma \hookrightarrow \Sigma'$ determines an inclusion function $\iota\colon \mathsf{Sen}^{\mathsf{TA}}(\Sigma) \hookrightarrow \mathsf{Sen}^{\mathsf{TA}}(\Sigma')$, which corresponds to the approach of classical model theory. This simplifies the presentation greatly. A situation when we cannot apply this convention arises when translating a Σ-sentence $\exists X \cdot \phi$ along the inclusion $\iota_X : \Sigma \hookrightarrow \Sigma[X]$.

Satisfaction relation. Actions are interpreted as binary relations in models. Given a model $\mathfrak{A}$ over a signature Σ, and actions $\mathfrak{a}, \mathfrak{a}_1, \mathfrak{a}_2 \in A$, we have:

- $(\mathfrak{a}_1 \mathbin{⨟} \mathfrak{a}_2)^{\mathfrak{A}} = \mathfrak{a}_1^{\mathfrak{A}} \mathbin{⨟} \mathfrak{a}_2^{\mathfrak{A}}$ (i.e., diagrammatic composition of binary relations);
- $(\mathfrak{a}_1 \cup \mathfrak{a}_2)^{\mathfrak{A}} = \mathfrak{a}_1^{\mathfrak{A}} \cup \mathfrak{a}_2^{\mathfrak{A}}$ (the union of binary relations); and
- $(\mathfrak{a}^*)^{\mathfrak{A}} = (\mathfrak{a}^{\mathfrak{A}})^*$ (the reflexive and transitive closure of binary relations).

We define the *satisfaction relation* between models and sentences as follows:

- $\mathfrak{A} \models t = t'$ iff $t^{\mathfrak{A}} = t'^{\mathfrak{A}}$;
- $\mathfrak{A} \models t_1\ \mathfrak{a}\ t_2$ iff $(t_1^{\mathfrak{A}}, t_2^{\mathfrak{A}}) \in \mathfrak{a}^{\mathfrak{A}}$;
- $\mathfrak{A} \models \neg\phi$ iff $\mathfrak{A} \not\models \phi$;
- $\mathfrak{A} \models \vee\Phi$ iff $\mathfrak{A} \models \phi$ for some sentence $\phi \in \Phi$; and
- $\mathfrak{A} \models \exists X \cdot \phi'$ iff $\mathfrak{A}' \models \phi'$ for some expansion $\mathfrak{A}'$ of $\mathfrak{A}$ to the signature $\Sigma[X]$.

▶ **Proposition 4** (Satisfaction condition). *For all signature morphisms $\chi : \Sigma \to \Sigma'$, all Σ'-models $\mathfrak{A}$ and all Σ-sentences ϕ we have: $\mathfrak{A}\restriction_{\chi} \models \phi \quad$ iff $\quad \mathfrak{A} \models \chi(\phi)$.*

The satisfaction condition for TA is given in [19].

Logical framework. Throughout this paper, we work within a fragment $\mathcal{L}$ of TA which is obtained by (a) restricting the category of signatures – without prohibiting signature extensions with constants – and (b) discarding a subset of sentence or/and action operators from the grammar which is used to define sentences in TA. When there is no danger of confusion we drop (a) the subscript $\mathcal{L}$ from the notations $\mathsf{Sig}^{\mathcal{L}}$ and $\mathsf{Sen}^{\mathcal{L}}$, and (b) the subscript TA from the notation $\mathsf{Mod}^{\mathsf{TA}}$. We denote by $\mathsf{Sen}_0 : \mathsf{Sig} \to \mathbb{S}\mathsf{et}$ the sub-functor of $\mathsf{Sen} : \mathsf{Sig} \to \mathbb{S}\mathsf{et}$, which maps each signature Σ in $\mathcal{L}$ to the set $\mathsf{Sen}_0(\Sigma)$ of atomic sentences from $\mathsf{Sen}(\Sigma)$.

3 Related concepts

We define the necessary concepts and present the existing results for presenting our advancements. We make the following notational conventions:

- Recall that in any category $\mathcal{C}$, the notation $|\mathcal{C}|$ refers to the class of objects in $\mathcal{C}$.

- Let $\mathbb{C}\mathsf{lass}$ denote the large category whose objects are classes and whose morphisms are class functions. [3]
- Assume a set C and a cardinal α. Let $\mathcal{P}_\alpha(C) \coloneqq \{C_1 \subseteq C \mid \mathsf{card}(C_1) < \alpha\}$, the set of all subsets of C of cardinality strictly less than α. We write $C_1 \subseteq_\alpha C$ if $C_1 \in \mathcal{P}_\alpha(C)$.

Presentations. Let Σ be a signature, and assume a class of Σ-models $\mathbb{M}$ and a set of Σ-sentences Φ.

- We write $\mathbb{M} \models \Phi$ if $\mathfrak{A} \models \Phi$ for all models $\mathfrak{A} \in \mathbb{M}$.
- Let $\mathbb{M}^\bullet = \{\varphi \in \mathsf{Sen}(\Sigma) \mid \mathbb{M} \models \Phi\}$, the set of sentences satisfied by all models in $\mathbb{M}$.
- Let $\Phi^\bullet = \{\mathfrak{A} \in |\mathsf{Mod}(\Sigma)| \mid \mathfrak{A} \models \Phi\}$, the class of models which satisfy Φ.
- Let $\mathsf{Mod}(\Sigma, \Phi)$ be the full subcategory of $\mathsf{Mod}(\Sigma)$ of models which satisfy Φ.

A *presentation* is a pair (Σ, Φ) consisting of a signature Σ and a set of Σ-sentences Φ. A *theory* is a presentation (Σ, Φ) such that $\Phi = \Phi^{\bullet\bullet}$. A *presentation morphism* $\chi : (\Sigma, \Phi) \to (\Sigma', \Phi')$ consist of a signature morphism $\chi : \Sigma \to \Sigma'$ such that $\Phi' \models_{\Sigma'} \chi(\Phi)$. A *theory morphism* is a presentation morphism between theories.

Substitutions. Let Σ be a signature, C_1 and C_2 two S-sorted sets of new constants for Σ . A substitution $\theta : C_1 \to C_2$ over Σ is a mapping from C_1 to $T_\Sigma(C_2)$. As in case of signature morphisms, a substitution $\theta : C_1 \to C_2$ determines

- a sentence functor $\mathsf{Sen}(\theta) : \mathsf{Sen}(\Sigma[C_1]) \to \mathsf{Sen}(\Sigma[C_2])$, which preserves Σ and maps each constant $c \in C_1$ to a term $\theta(c)$, and
- a reduct functor $\upharpoonright_\theta : \mathsf{Mod}(\Sigma[C_2]) \to \mathsf{Mod}(\Sigma[C_1])$, which preserves the interpretation of Σ and assigns to each constant $c \in C_1$ the interpretation of the term $\theta(c) \in T_\Sigma(C_2)$ in the category $\mathsf{Mod}(\Sigma[C_2])$, that is, for all $\mathfrak{A} \in |\mathsf{Mod}(\Sigma[C_2])|$ we have (a) $(\mathfrak{A}\upharpoonright_\theta)_s = \mathfrak{A}_s$ for all sorts s in Σ, (b) $x^{(\mathfrak{A}\upharpoonright_\theta)} = x^{\mathfrak{A}}$ for all function or transition symbols x in Σ, and (c) $c^{(\mathfrak{A}\upharpoonright_\theta)} = \theta(c)^{\mathfrak{A}}$ for all constants $c \in C_1$.

As in the case of signature morphisms, we use θ to denote both the substitution $\theta : C_1 \to C_2$ and the functor $\mathsf{Sen}(\theta) : \mathsf{Sen}(\Sigma[C_1]) \to \mathsf{Sen}(\Sigma[C_2])$. The following result is a straightforward generalization of [7, Proposition 5.6].

▶ **Proposition 5** (Satisfaction condition for substitutions). *For all substitutions $\theta : C_1 \to C_2$, all $\Sigma[C_2]$-models $\mathfrak{A}$ and all $\Sigma[C_1]$-sentences ϕ we have: $\mathfrak{A}\upharpoonright_\theta \models \phi$ iff $\mathfrak{A} \models \theta(\phi)$.*

A *reachable (transition) algebra* [26] defined over a signature Σ is a (transition) Σ-algebra $\mathfrak{A}$ such that the unique homomorphism $h : T_\Sigma \to \mathfrak{A}$ is surjective. By [17, Proposition 8], a Σ-algebra $\mathfrak{A}$ is reachable iff for all sets of new constants C for Σ and all expansions $\mathfrak{B}$ of $\mathfrak{A}$ to $\Sigma[C]$, there exists a substitution $\theta : C \to \emptyset$ such that $\mathfrak{A}\upharpoonright_\theta = \mathfrak{B}$.

4 Compactness and Upward Löwenheim-Skolem Property

TA is very expressive and its sentences can control the cardinality of all its models under the the following assumptions:

- Generalized Continuum Hypothesis (GCH) holds, that is, for any infinite cardinal number α, the next larger cardinal is exactly 2^α; and
- there are no inaccessible cardinals (¬IC).

[3] By contrast, a small category is one in which both the collection of objects and the collection of morphisms form sets, rather than proper classes.

▶ **Theorem 6** (Categoricity). *Assume that $\mathcal{L} = \mathsf{TA}$. Let $\mathfrak{A}$ be a model defined over a signature $\Sigma = (S, F, L)$. There exists a sort preserving inclusion of signatures $\iota^\circ : \Sigma \hookrightarrow \Sigma^\circ$ and a reachable ι°-expansion $\mathfrak{A}^\circ$ of $\mathfrak{A}$ such that any model of $(\Sigma^\circ, Th(\mathfrak{A}^\circ))$ is isomorphic to $\mathfrak{A}^\circ$, where $Th(\mathfrak{A}^\circ) = \{\phi \in \mathsf{Sen}(\Sigma^\circ) \mid \mathfrak{A}^\circ \models \phi\}$.*

Theorem 6 is proven assuming ZFC + GCH + ¬IC. These axioms are consistent relative to ZFC – that is, if ZFC is consistent, then so is the extended theory. Consequently, Theorem 6 (and its conclusion) cannot be refuted within ZFC. This illustrates that certain logical properties cannot be established within TA from ZFC alone. The following example shows that joint Robinson consistency property fails in TA.

▶ **Example 7** (Robinson Consistency). Consider a complete (first-order) theory (Σ, T) with at least two non-isomorphic models $\mathfrak{A}$ and $\mathfrak{B}$. Construct a pushout as illustrated on the left side of the diagram below.

$$\begin{tikzcd}
\Sigma^\circ \arrow[r, "\chi_2"] & \Sigma' \\
\Sigma \arrow[u, hook, "\iota^\circ"] \arrow[r, hook, "\iota^\circ"] & \Sigma^\circ \arrow[u, "\chi_1"']
\end{tikzcd}
\qquad
\begin{tikzcd}
(\Sigma^\circ, Th(\mathfrak{B}^\circ)) \arrow[r, "\chi_2"] & (\Sigma', \chi_1(Th(\mathfrak{A}^\circ)) \cup \chi_2(Th(\mathfrak{B}^\circ))) \\
(\Sigma, T) \arrow[u, hook, "\iota^\circ"] \arrow[r, hook, "\iota^\circ"] & (\Sigma^\circ, Th(\mathfrak{A}^\circ)) \arrow[u, "\chi_1"']
\end{tikzcd}$$

Since $\mathfrak{A} \models T$ and $\mathfrak{A}^\circ\upharpoonright_\Sigma = \mathfrak{A}$, we have $T \subseteq Th(\mathfrak{A}^\circ)$. Similarly, since $\mathfrak{B} \models T$ and $\mathfrak{B}^\circ\upharpoonright_\Sigma = \mathfrak{B}$, we have $T \subseteq Th(\mathfrak{B}^\circ)$. However, $\chi_1(Th(\mathfrak{A}^\circ)) \cup \chi_2(Th(\mathfrak{B}^\circ))$ is not consistent, because any model of $\chi_1(Th(\mathfrak{A}^\circ)) \cup \chi_2(Th(\mathfrak{B}^\circ))$, by Theorem 6, would imply that $\mathfrak{A}$ is isomorphic to $\mathfrak{B}$.

Typically, the cardinality of a model $\mathcal{A}$ over a signature $\Sigma = (S, F, L)$ is defined as the sum of the cardinalities of its carrier sets, that is, $\mathsf{card}(\mathfrak{A}) = \sum_{s \in S} \mathsf{card}(\mathfrak{A}_s)$. The following example illustrates that the classical Upward Löwenheim-Skolem (ULS) property does not hold when sorts are treated as unary predicates rather than as distinct domains.

▶ **Example 8.** Let Σ be a signature with countably infinitely many sorts $\{s_n\}_{n<\omega}$, where each sort s_n has exactly one constant symbol $c_n :\to s_n$, and there are no transition labels. Define the set of equations $\Gamma := \{\forall x_n \cdot (x_n = c_n) \mid n < \omega\}$, which asserts that each sort s_n contains exactly one element.

Note that Σ is a first-order signature, and Γ is a set of Σ-sentences that has an infinite model. However, Γ has no models of cardinality greater than ω, indicating that the ULS property, as traditionally stated, fails to apply in this setting. The following definition of the ULS property applies to many-sorted first-order logic.

▶ **Definition 9** (Upward Löwenheim-Skolem Property). α-Upward Löwenheim-Skolem *property holds for an infinite cardinal α whenever for all signatures $\Sigma = (S, F, L)$, all Σ-models $\mathfrak{A}$ and all sorts $s \in S$ such that $\alpha \geq \mathsf{card}(\mathsf{Sen}(\Sigma))$ and $\alpha \geq \mathsf{card}(\mathfrak{A}_s) \geq \omega$ there exists an elementary extension $\mathfrak{B}$ of $\mathfrak{A}$ such that $\mathsf{card}(\mathfrak{B}_s) = \alpha$.*

Recall that α*-compactness* holds whenever for all signatures Σ and all sets of Σ-sentences Φ, Φ has a model iff all subsets $\Psi \in \mathcal{P}_\alpha(\Phi)$ have a model. From Theorem 6, one can show that both ULS property and compactness fail.

▶ **Corollary 10.** *Assume that $\mathcal{L} = \mathsf{TA}$. Then (a) α-ULS fails for all cardinals $\alpha > \omega$, and (b) α-compactness fails for all infinite cardinals α.*

Corollary 10 shows that the iteration operator for actions provides enough expressive power to control the cardinality of models. Moreover, compactness fails in all its forms. In this setting, we aim to develop a model construction method that does not rely on compactness, yet still supports the proof of logical properties such as Downward Löwenheim-Skolem (DLS) theorem and OTT. It is worth noting that the results presented in the following sections hold without assuming either GCH or ¬IC.

5 Forcing

In this section, we present a forcing technique, originally proposed in [19], which extends classical forcing from one signature to a category of signatures in a non-trivial way. To illustrate the motivation behind our approach, we recall Example 28 from [19].

▶ **Example 11.** Let Σ be a signature defined as follows: (a) $S \coloneqq \{s_i \mid i \in \omega\}$, (b) $F \coloneqq \{c :\to s_0, d :\to s_0\}$, and (c) $L \coloneqq \{\lambda\}$. Let Φ be a set of Σ-sentences which consists of: (a) $\lambda^*(c,d)$, and (b) $(\exists x_n \cdot \top) \Rightarrow \neg\lambda^n(c,d)$ for all $n \in \omega$, where x_n is a variable of sort s_n.

The first sentence asserts that there is a transition from c to d in a finite number of steps. For each natural number n, the sentence $(\exists x_n \cdot \top) \Rightarrow \neg\lambda^n(c,d)$ states that if the sort s_n is not empty then there is no transition from c to d in exactly n steps. Classical model construction techniques typically involve introducing an infinite number of constants for each sort – commonly referred to as *Henkin constants* – followed by the application of quantifier elimination methods that select witnesses for existentially quantified variables. Let $C = \{C_{s_n}\}_{n<\omega}$ be a collection of Henkin constants, that is, newly introduced constants such that each C_{s_n} is countably infinite for all $n < \omega$. Although the original theory (Σ, Φ) is satisfiable, its extension $(\Sigma[C], \Phi)$ becomes unsatisfiable upon the addition of these constants. This discrepancy highlights the limitations of classical approaches and motivates the need for a novel method to handle the introduction of Henkin constants effectively.

▶ **Definition 12** (Forcing property). A forcing property *is a tuple* $\mathbb{P} = (P, \leq, \Delta, f)$*, where:*

$$\begin{tikzcd} (P,\leq) \arrow[rr, bend left, "f"] \arrow[dr, "\Delta"'] & {} \arrow[d, Rightarrow, "\subseteq"] & \mathbb{S}\mathsf{et} \\ & \mathsf{Sig} \arrow[ur, "\mathsf{Sen}_0"'] & \end{tikzcd}$$

1. $(P, \leq)$ *is a partially ordered set with a least element* 0.
   The elements of P *are traditionally called conditions.*
2. $\Delta : (P, \leq) \to \mathsf{Sig}$ *is a functor, which maps each arrow* $(p \leq q) \in (P, \leq)$ *to an inclusion* $\Delta(p) \subseteq \Delta(q)$.
3. $f : (P, \leq) \to \mathbb{S}\mathsf{et}$ *is a functor from the small category* $(P, \leq)$ *to the category of sets* $\mathbb{S}\mathsf{et}$ *such that* $f \subseteq \Delta; \mathsf{Sen}_0$ *is a natural transformation, that is:* (a) $f(p) \subseteq \mathsf{Sen}_0(\Delta(p))$ *for all conditions* $p \in P$*, and* (b) $f(p) \subseteq f(q)$ *for all relations* $(p \leq q) \in (P, \leq)$.
4. *If* $f(p) \models \phi$ *then* $\phi \in f(q)$ *for some* $q \geq p$*, for all atomic sentences* $\phi \in \mathsf{Sen}_0(\Delta(p))$.

A classical forcing property is a particular case of forcing property such that $\Delta(p) = \Delta(q)$ for all conditions $p, q \in P$. As usual, forcing properties determine suitable relations between conditions and sentences.

▶ **Definition 13** (Forcing relation). *Let* $\mathbb{P} = \langle P, \leq, \Delta, f \rangle$ *be a forcing property.* The forcing relation $\Vdash$ *between conditions* $p \in P$ *and sentences from* $\mathsf{Sen}(\Delta(p))$ *is defined by induction on the structure of sentences, as follows:*

- $p \Vdash \varphi$ *if* $\varphi \in f(p)$*, for all atomic sentences* $\varphi \in \mathsf{Sen}_0(\Delta(p))$.
- $p \Vdash (\mathfrak{a}_1 \fatsemi \mathfrak{a}_2)(t_1, t_2)$ *if* $p \Vdash \mathfrak{a}_1(t_1, t)$ *and* $p \Vdash \mathfrak{a}_2(t, t_2)$ *for some* $t \in T_{\Delta(p)}$.
- $p \Vdash (\mathfrak{a}_1 \cup \mathfrak{a}_2)(t_1, t_2)$ *if* $p \Vdash \mathfrak{a}_1(t_1, t_2)$ *or* $p \Vdash \mathfrak{a}_2(t_1, t_2)$.
- $p \Vdash \mathfrak{a}^*(t_1, t_2)$ *if* $p \Vdash \mathfrak{a}^n(t_1, t_2)$ *for some natural number* $n < \omega$.

- *$p \Vdash \neg\phi$ if there is no $q \geq p$ such that $q \Vdash \phi$.*
- *$p \Vdash \vee\Phi$ if $p \Vdash \phi$ for some $\phi \in \Phi$.*
- *$p \Vdash \exists X \cdot \phi$ if $p \Vdash \theta(\phi)$ for some substitution $\theta : X \to T_{\Delta(p)}$.*

The relation $p \Vdash \phi$ in $\mathbb{P}$, is read as p forces ϕ.

A few basic properties of forcing are presented below.

▶ **Lemma 14** (Forcing properties [19]). *Let $\mathbb{P} = (P, \leq, \Delta, f)$ be a forcing property. For all conditions $p \in P$ and all sentences $\phi \in \mathsf{Sen}(\Delta(p))$ we have:*

1. *$p \Vdash \neg\neg\phi$ iff for each $q \geq p$ there is a condition $r \geq q$ such that $r \Vdash \phi$.*
2. *If $p \leq q$ and $p \Vdash \phi$ then $q \Vdash \phi$.*
3. *If $p \Vdash \phi$ then $p \Vdash \neg\neg\phi$.*
4. *We can not have both $p \Vdash \phi$ and $p \Vdash \neg\phi$.*

The second property stated in the above lemma shows that the forcing relation is preserved along inclusions of conditions. The fourth property shows that the forcing relation is consistent, that is, a condition cannot force all sentences. The remaining conditions are about negation.

▶ **Definition 15** (Generic set). *Let $\mathbb{P} = (P, \leq, \Delta, f)$ be a forcing property. A non-empty subset of conditions $G \subseteq P$ is generic if*

1. *G is an ideal, that is: (a) $G \neq \emptyset$, (b) for all $p \in G$ and all $q \leq p$ we have $q \in G$, and (c) for all $p, q \in G$ there exists $r \in G$ such that $p \leq r$ and $q \leq r$; and*
2. *for all conditions $p \in G$ and all sentences $\phi \in \mathsf{Sen}(\Delta(p))$ there exists a condition $q \in G$ such that $q \geq p$ and either $q \Vdash \phi$ or $q \Vdash \neg\phi$ holds.*

We write $G \Vdash \phi$ if $p \Vdash \phi$ for some $p \in G$.

A generic set G describes a reachable model which satisfies all sentences forced by the conditions in G.

▶ Remark 16. Since $\Delta : (G, \leq) \to \mathsf{Sig}$ is a directed diagram of signature inclusions, one can construct a co-limit $\mu : \Delta \Rightarrow \Delta_G$ of the functor $\Delta : (G, \leq) \to \mathsf{Sig}$ such that $\mu_p : \Delta(p) \to \Delta_G$ is an inclusion for all $p \in G$. [4]

The results which leads to DLS and OTT are developed over a signature Δ_G such as the one described in Remark 16.

▶ **Definition 17** (Generic model). *Let $\mathbb{P} = (P, \leq, \Delta, f)$ be a forcing property and $G \subseteq P$ a generic set. A model $\mathfrak{A}$ defined over Δ_G is a* generic model *for G iff for every sentence $\phi \in \bigcup_{p \in G} \mathsf{Sen}(\Delta(p))$, we have $\mathfrak{A} \models \phi$ iff $G \Vdash \phi$.*

The notion of generic model is the semantic counterpart of the definition of generic set. The following result shows that every generic set has a generic model.

▶ **Theorem 18** (Generic Model Theorem [19]). *Let $\mathbb{P} = (P, \leq, \Delta, f)$ be a forcing property and $G \subseteq P$ a generic set. Then there a generic model $\mathfrak{A}$ for G which is reachable.*

[4] Note that Δ_G refers both to the vertex of the colimit of the diagram Δ, and the constant functor from the poset $(G, \leq)$ to the category Sig, which maps every object in G to the signature Δ_G.

6 Semantic Forcing

As an example of forcing property, we introduce *semantic forcing* and study its properties. Let us fix an arbitrary signature Σ and let $\alpha := \mathsf{card}(\mathsf{Sen}(\Sigma))$ be the *power* of Σ. Consider an $\mathcal{S}$-sorted set of new constants $C = \{C_s\}_{s\in S}$ such that $\mathsf{card}(C_s) = \alpha$ for all sorts $s \in S$. These constants, traditionally known as Henkin constants, are used in the construction of models. Throughout this section we assume the following:

1. a partially ordered set $(\mathcal{K}, \preceq)$ with a least element $0_{\mathcal{K}}$;
2. a signature functor $\Omega : (\mathcal{K}, \preceq) \to \mathsf{Sig}$ which maps (a) $0_{\mathcal{K}}$ to Σ, (b) each element $\kappa \in \mathcal{K}$ to a signature $\Sigma[C_\kappa]$, where C_κ is a subset of C with cardinality strictly less than α, that is, $C_\kappa \in \mathcal{P}_\alpha(C)$, and (c) each relation $(\kappa \preceq \ell) \in (\mathcal{K}, \preceq)$ to an inclusion of signatures $\Sigma[C_\kappa] \subseteq \Sigma[C_\ell]$;

$$\begin{tikzcd}
(\mathcal{K},\preceq) \arrow[rr, bend left, "\mathcal{M}"] \arrow[dr, bend right, "\Omega"'] & {} \arrow[d, Rightarrow, "\subseteq"] & \mathbb{C}\mathsf{lass}^{op} \\
& \mathsf{Sig} \arrow[ur, bend right, "|\mathsf{Mod}|"'] &
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathcal{M}(\kappa) \arrow[d, dotted, "\subseteq"'] & \mathcal{M}(\ell) \arrow[l, "\upharpoonright_{\Omega(\kappa)}"'] \arrow[d, dotted, "\subseteq"] \\
|\mathsf{Mod}(\Omega(\kappa))| & |\mathsf{Mod}(\Omega(\ell))| \arrow[l, "\upharpoonright_{\Omega(\kappa)}"]
\end{tikzcd}$$

3. a model functor $\mathcal{M} : (\mathcal{K}, \preceq) \to \mathbb{C}\mathsf{lass}^{op}$ such that the following hold: [5]
   a. $\mathcal{M} \subseteq \Omega \mathbin{⨟} |\mathsf{Mod}|$ is a natural transformation, that is, (a) $\mathcal{M}(\kappa) \subseteq |\mathsf{Mod}(\Omega(\kappa))|$ for all elements $\kappa \in \mathcal{K}$, and (b) $\mathcal{M}(\ell)\upharpoonright_{\Omega(\kappa)} \subseteq \mathcal{M}(\kappa)$ for all relations $(\kappa \preceq \ell) \in (\mathcal{K}, \preceq)$;
   b. Assume an element $\kappa \in \mathcal{K}$, a model $\mathfrak{A} \in \mathcal{M}(\kappa)$ and a set of constants $C_1 \subseteq_\alpha C \setminus C_\kappa$. For all expansions $\mathfrak{B}$ of $\mathfrak{A}$ to $\Sigma[C_\kappa \cup C_1]$, there exists $\ell \succeq \kappa$ such that $C_\ell = C_\kappa \cup C_1$ and $\mathfrak{B} \in \mathcal{M}(\ell)$.

$$\begin{tikzcd}
\mathcal{M}(\kappa) \ni \mathfrak{A} \arrow[r, dotted, no head] & \Sigma[C_\kappa] \arrow[r, hook] & \Sigma[C_\kappa \cup C_1] = \Sigma[C_\ell] \arrow[r, dotted, no head] & \mathfrak{B} \in \mathcal{M}(\ell) \arrow[lll, dotted, bend left=12, "\upharpoonright_{\Sigma[C_\kappa]}"]
\end{tikzcd}$$

The condition 3(b) implies that all sets of constants from $\mathcal{P}_\alpha(C)$ appear in some signature from the image of Ω.

▶ **Example 19.** Let $\mathfrak{A}$ be a model defined over a signature Σ.

1. $\mathcal{K} = \{(\Sigma[C'], \mathfrak{A}') \mid C' \in \mathcal{P}_\alpha(C)$ and $\mathfrak{A}' \in |\mathsf{Mod}(\Sigma[C'])|$ such that $\mathfrak{A}'\upharpoonright_\Sigma = \mathfrak{A}\}$, and $(\Sigma[C'], \mathfrak{A}') \preceq (\Sigma[C''], \mathfrak{A}'')$ iff $C' \subseteq C''$ and $\mathfrak{A}''\upharpoonright_{\Sigma[C']} = \mathfrak{A}'$;
2. $\Omega : (\mathcal{K}, \preceq) \to \mathsf{Sig}$ is the forgetful functor mapping each $(\Sigma[C'], \mathfrak{A}')$ to $\Sigma[C']$;
3. $\mathcal{M} : (\mathcal{K}, \preceq) \to \mathbb{C}\mathsf{lass}^{op}$ is the model functor which maps each $(\Sigma[C'], \mathfrak{A}') \in \mathcal{K}$ to $\{\mathfrak{A}'\}$.

Example 19 shows that $(\mathcal{K}, \preceq)$ can have a more complex structure than $(\mathcal{P}_\alpha(C), \subseteq)$.

▶ **Example 20.** Let Φ be any set of sentences defined over a signature Σ.

1. $\mathcal{K} = \{(\Sigma[C'], \Phi) \mid C' \in \mathcal{P}_\alpha(C)\}$, and $(\Sigma[C'], \Phi) \preceq (\Sigma[C''], \Phi)$ iff $C' \subseteq C''$;
2. $\Omega : (\mathcal{K}, \preceq) \to \mathsf{Sig}$ is the forgetful functor mapping each $(\Sigma[C'], \Phi)$ to $\Sigma[C']$;
3. $\mathcal{M} : (\mathcal{K}, \preceq) \to \mathbb{C}\mathsf{lass}^{op}$ is the functor which maps each $(\Sigma[C'], \Phi) \in \mathcal{K}$ to $|\mathsf{Mod}(\Sigma[C'], \Phi)|$, the class of $\Sigma[C']$-models which satisfy Φ.

[5] $\mathbb{C}\mathsf{lass}^{op}$ is the opposite category of $\mathbb{C}\mathsf{lass}$, which means: (a) it has the same objects as $\mathbb{C}\mathsf{lass}$, that is, classes, but (b) morphisms are reversed, that is, a morphism $f : A \to B$ in $\mathbb{C}\mathsf{lass}^{op}$ corresponds to a class function $f : B \to A$ in $\mathbb{C}\mathsf{lass}$.

▶ **Definition 21** (Semantic Forcing Property). *The semantic forcing property* $\mathbb{P}(\Omega, \mathcal{M}) = (P, \leq, \Delta, f)$ *is defined as follows:*

1. *A condition* p *is a pair* (κ_p, Φ_p)*, where* $\kappa_p \in \mathcal{K}$ *and* $\Phi_p \subseteq_\alpha \mathsf{Sen}(\Omega(\kappa_p))$ *such that* $\mathcal{M}(\kappa_p) \cap \Phi_p^\bullet \neq \emptyset$*, that is, there exists at least one model in* $\mathcal{M}(\kappa_p)$ *which satisfies* Φ_p*.*
2. $p \leq q$ *iff* $\kappa_p \preceq \kappa_q$ *and* $\Phi_p \subseteq \Phi_q$*, for all conditions* $p = (\kappa_p, \Phi_p), q = (\kappa_q, \Phi_q) \in P$*.*
3. $\Delta : (P, \leq) \to \mathsf{Sig}$ *is the functor mapping each condition* $p = (\kappa_p, \Phi_p) \in P$ *to* $\Omega(\kappa_p) \in |\mathsf{Sig}|$*.*
4. *Let* $f(p) = \mathsf{Sen}_0(\Omega(\kappa_p)) \cap (\mathcal{M}(\kappa_p) \cap \Phi_p^\bullet)^\bullet$*, for all conditions* $p \in P$*. In other words,* $f(p)$ *is the set of atomic sentences satisfied by all models of* Φ_p *from* $\mathcal{M}(\kappa_p)$*.*

From a category theory perspective, several observations are worth noting.

▶ Remark 22. Definition 21 describe the following mathematical structures:

- $\Lambda : (P, \leq) \to (\mathcal{K}, \prec)$, defined by $\Lambda(p) = \kappa_p$ for all $p \in P$, is a monotone function such that $\Delta = \Lambda \mathbin{⨾} \Omega$.
- $\Gamma : (P, \leq) \to \mathbb{S}\mathsf{et}$, defined by $\Gamma(p) = \Phi_p$ for all $p \in P$, is a functor such that the inclusion $\subseteq : \Gamma \Rightarrow (\Lambda \mathbin{⨾} \Omega \mathbin{⨾} \mathsf{Sen})$ is a natural transformation, that is:
  1. $\Gamma(p) \subseteq \mathsf{Sen}(\Delta(p))$ for all conditions $p \in P$, and
  2. $\Gamma(p) \subseteq \Gamma(q)$ for all relations $(p \leq q) \in (P, \leq)$.

$$\begin{tikzcd}
(P,\leq) \arrow[rrr, bend left, "\Gamma"{name=G}] \arrow[dr, "\Lambda"'] & & & \mathbb{S}\mathsf{et} \\
& (\mathcal{K},\preceq) \arrow[r, "\Omega"'{name=O}] & \mathsf{Sig} \arrow[ur, "\mathsf{Sen}"'] &
\arrow[Rightarrow, from=G, to=O, "\subseteq"]
\end{tikzcd}
\qquad
\begin{tikzcd}
(\mathcal{K},\preceq) \arrow[rr, bend left, "\mathcal{M}"{name=M}] \arrow[dr, bend right, "\Omega"'] & & \mathbb{C}\mathsf{lass}^{op} \\
& \mathsf{Sig} \arrow[ur, bend right, "|\mathsf{Mod}|"'] &
\arrow[Rightarrow, from=M, to=2-2, "\subseteq"]
\end{tikzcd}$$

Let $p \in P$ be a condition.

1. We let $\mathsf{Mod}(p)$ denote $\mathcal{M}(\Lambda(p)) \cap \Gamma(p)^\bullet$, the class of models in $\mathcal{M}(\Lambda(p))$ satisfying $\Gamma(p)$.
2. We write $p \models \phi$ iff ϕ is satisfied by all models in $\mathsf{Mod}(p)$, in symbols, $\mathsf{Mod}(p) \models \phi$.

The perspective of category theory can be disregarded if the reader is not familiar with it. However, Remark 22 lays the groundwork for a generalization to an abstract level, as defined by the notion of institution [11].

We define a notion of distance between conditions, which provides a principled method for regulating the addition of Henkin constants from C to the starting signature Σ.

▶ **Definition 23** (Distance & weak forcing). *The distance between two conditions* p *and* q*, with* $p \leq q$*, is defined as the number of constants and sentences that must be added to* p *to obtain* q*, that is,* $d(p, q) := \mathsf{card}(\Delta(q) \setminus \Delta(p)) + \mathsf{card}(\Gamma(q) \setminus \Gamma(p))$*. A condition* p *weakly forces* ϕ*, in symbols,* $p \Vdash^w \phi$*, if for all* $q \geq p$ *there exists* $r \geq q$ *such that* $d(q, r) < \omega$ *and* $r \Vdash \phi$*.*

Using Lemma 14 (2) and the reflexivity of $\leq$, it is not difficult to show that $p \Vdash \phi$ implies $p \Vdash^w \phi$. In the classical setting, weak forcing does not involve any notion of distance: $p \Vdash^w \phi$ if and only if, for all $q \geq p$, there exists $r \geq q$ such that $r \Vdash \phi$; and this, in turn, holds if and only if $p \Vdash \neg\neg\phi$. The notion of distance is introduced in this paper to extend the applicability of the results to all signatures – including those of singular cardinality – by controlling the number of constants introduced during the construction of the generic sets.

▶ **Theorem 24** (Semantic Forcing Theorem). *For all conditions* $p \in P$ *and all sentences* $\phi \in \mathsf{Sen}(\Delta(p))$*, we have: (a)* $p \models \phi$ *iff (b)* $p \Vdash^w \phi$

7 Downward Löwenheim-Skolem Theorem

Consider a semantic forcing property $\mathbb{P}(\Omega, \mathcal{M})$ such that the functors Ω and $\mathcal{M}$ are defined as in Example 19. We use $\mathbb{P}(\Omega, \mathcal{M})$ to give a proof of Downward Löwenheim-Skolem Theorem. The key to proving any result based on forcing is the construction of a generic set for a given condition p.

▶ **Lemma 25** (Existence). *Let $\mathbb{P}(\Omega, \mathcal{M}) = (P, \leq, \Delta, f)$ be a semantic forcing property such that the functors Ω and $\mathcal{M}$ are defined as in Example 19. Any condition $p \in P$ belongs to a generic set G_p.*

Proof. Let $pair : \alpha \times \alpha \to \alpha$ be any bijection such that for all ordinals $i, j, \beta < \alpha$ if $\beta = pair(i, j)$ then $i \leq \beta$. For any condition $q \in P$, let $\gamma(q) : \alpha \to \mathsf{Sen}(\Delta(q))$ be an enumeration of the sentences in $\mathsf{Sen}(\Delta(q))$.

First, we define an increasing chain of conditions $p_0 \leq p_1 \leq \dots \leq p_\beta \leq \dots$ with the following property: for all ordinals $\beta < \alpha$ there exists $n < \omega$ such that $d(p, p_\beta) \leq n \cdot \mathrm{card}(\beta)$.[6] We proceed by induction on ordinals $\beta < \alpha$.

($\beta = 0$) Let $p_0 := p$.

($\beta \Rightarrow \beta + 1$) Let $(i, j) := pair^{-1}(\beta)$. Notice that $i \leq \beta$, which means p_i is already defined. Condition $p_{\beta+1}$ is obtained by adding a finite set of constants from C and a finite set of sentences to p_β, according to a case distinction detailed below:

- **($q \Vdash \gamma(p_i, j)$ for some $q \geq p_\beta$)** Since $q \Vdash \gamma(p_i, j)$, by Theorem 24, $q \models \gamma(p_i, j)$. It follows that $r := (\Lambda(p_\beta), \Gamma(p_\beta) \cup \{\gamma(p_i, j)\}) \in P$. Since $r \vDash \gamma(p_i, j)$, by Theorem 24, there exists $p_{\beta+1} \geq r$ such that $p_{\beta+1} \Vdash \gamma(p_i, j)$ and $d(r, p_{\beta+1}) < \omega$. By induction hypothesis, there exists $n < \omega$ such that $d(p, p_{\beta+1}) \leq n \cdot \mathsf{card}(\beta + 1)$.
- **($q \nVdash \gamma(p_i, j)$ for every $q \geq p_\beta$)** Let $p_{\beta+1} := p_\beta$. In this case, $p_{\beta+1} \Vdash \neg\gamma(p_i, j)$.

($\beta < \alpha$ is a limit ordinal) At this point, for all $i < \beta$ we have defined $p_i = (\Lambda(p_i), \Gamma(p_i))$, where $\Lambda(p_i) = (\Delta(p_i), \mathfrak{A}_i)$ and $\mathfrak{A}_i$ is an expansion of $\mathfrak{A}$ to $\Delta(p_i)$.

- Let $C_{\Lambda(p_\beta)} := \bigcup_{i<\beta} C_{\Lambda(p_i)}$ and $\Delta(p_\beta) = \Sigma[C_{\Lambda(p_\beta)}]$ and $\Gamma(p_\beta) := \bigcup_{i<\beta} \Gamma(p_i)$. Since $\mathsf{card}(C_{\Lambda(p_i)}) \leq \mathsf{card}(\beta)$ for all $i < \beta$, we get $\mathsf{card}(C_{\Lambda(p_\beta)}) \leq \mathsf{card}(\beta) < \alpha$. Similarly, $\mathsf{card}(\Gamma(p_\beta)) \leq \mathsf{card}(\beta) < \alpha$.
- A model $\mathfrak{A}_i \in \mathcal{M}(\Lambda(p_i))$, where $i < \beta$, can be regarded as a pair $(\mathfrak{A}, g_i : C_{\Lambda(p_i)} \to \mathfrak{A})$, consisting of the model $\mathfrak{A}$ and an interpretation of the constants in $C_{\Lambda(p_i)}$ into the elements of $\mathfrak{A}$ given by the function g_i. Define $g_\beta := \bigcup_{i<\beta} g_i$, let $\mathfrak{A}_\beta := (\mathfrak{A}, g_\beta)$ and $\Lambda(p_\beta) := (\Delta(p_\beta), \mathfrak{A}_\beta)$.
- Let $p_\beta := (\Lambda(p_\beta), \Gamma(p_\beta))$. By its definition, $\mathfrak{A}_\beta\upharpoonright_{\Delta(p_i)} = \mathfrak{A}_i$ for all $i < \beta$. Since $\mathfrak{A}_i \models \Gamma(p_i)$ for all $i < \beta$, by satisfaction condition, $\mathfrak{A}_\beta \models \Gamma(p_\beta)$. Hence, p_β is well-defined. Since for all $i < \beta$ there exists $n_i \in \mathbb{N}$ such that $d(p, p_i) \leq n_i \cdot \mathsf{card}(i)$, we get $d(p, p_\beta) \leq \mathsf{card}(\beta)$.

Secondly, one can easily show that $G_p := \{q \in P \mid q \leq p_\beta \text{ for some } \beta < \alpha\}$ is a generic set. ◀

▶ **Theorem 26** (Downward Löwenheim-Skolem Theorem). *Let $\mathfrak{A}$ be a model over a signature $\Sigma = (S, F, L)$ of power α such that $\mathsf{card}(\mathfrak{A}_s) \geq \alpha$ for some sort $s \in S$. Then there exists an elementary submodel $\mathfrak{B}$ of $\mathfrak{A}$ such that $\mathsf{card}(\mathfrak{B}_s) = \alpha$.*

[6] When α is a singular cardinal, the entire set of constants C may be used along a chain $\{p_i\}_{i<\beta}$ with $\beta < \alpha$. To avoid this uncontrolled increase in the size, we use the distance measure d.

8 Omitting Types Theorem

Consider a semantic forcing property $\mathbb{P}(\Omega, \mathcal{M})$ such that Ω and $\mathcal{M}$ are defined as in Example 20. We use $\mathbb{P}(\Omega, \mathcal{M})$ to give a proof of Omitting Types Theorem in TA.

▶ **Definition 27** (Types). *A type for a signature Σ of power α is a set of sentences T defined over $\Sigma[X]$, where X is a block of variables for Σ such that $\alpha^{\mathsf{card}(X)} \leq \alpha$. A Σ-model $\mathfrak{A}$* realizes *T if $\mathfrak{B} \models T$ for some expansion $\mathfrak{B}$ of $\mathfrak{A}$ to $\Sigma[X]$. $\mathfrak{A}$* omits *T if $\mathfrak{A}$ does not realize T.*

For all infinite cardinals α we have $\alpha < \alpha^{\mathrm{cf}(\alpha)}$, where $\mathrm{cf}(\alpha)$ is the cofinality of α. Consequently, the condition $\alpha^{\mathsf{card}(X)} \leq \alpha$ implies that $\mathsf{card}(X) < \mathrm{cf}(\alpha)$. The converse also holds under the assumption of GCH. The inequality $\alpha^{\mathsf{card}(X)} \leq \alpha$ ensures that the number of mappings from X to the set of ground terms – that is, substitutions $\theta : X \to \emptyset$ – does not exceed α. Note that if α is the cardinality of real numbers and X is countable then $\alpha^{\mathsf{card}(X)} \leq \alpha$ holds.

▶ **Definition 28** (Isolated types). *Let Σ be a signature of power α. A set of sentences $\Phi \subseteq \mathsf{Sen}(\Sigma)$ is said to* isolate *a type $T \subseteq \mathsf{Sen}(\Sigma[X])$ iff there exist:*

- *an S-sorted set of new constants D for Σ with $\mathsf{card}(D) < \alpha$,*
- *a set $\Gamma \subseteq \mathsf{Sen}(\Sigma[D])$ with $\mathsf{card}(\Gamma) < \alpha$ and such that $T \cup \Gamma$ is satisfiable over $\Sigma[D]$, and*
- *a substitution $\theta : X \to D$,*

such that $\Phi \cup \Gamma \models_{\Sigma[D]} \theta(T)$. We say that Φ locally omits *T if Φ does not isolate T.*

Our definitions for type, realization, and isolation align with those provided in [16], with one key difference: we do not restrict the block of variables X to be finite. Definition 28 is similar to the definition of locally omitting types for first-order logic without equality from [22]. If the fragment $\mathcal{L}$ of TA is closed under Boolean operators and first-order quantifiers, the above definition coincides with the classical one for isolated types.

▶ **Lemma 29.** *Assume that (a) $\mathcal{L}$ is semantically closed under Boolean operators and first-order quantifiers, and (b) $\mathcal{L}$ is compact or $\alpha = \omega$. Then $\Phi \subseteq \mathsf{Sen}(\Sigma)$ isolates a type $T \subseteq \mathsf{Sen}(\Sigma[X])$ iff there exists a set of sentences $\Gamma \in \mathcal{P}_\alpha(\mathsf{Sen}(\Sigma[X]))$ such that $T \cup \Gamma$ is satisfiable over $\Sigma[X]$ and $\Phi \cup \Gamma \models_{\Sigma[X]} T$.*

The proof is conceptually identical with the proof of [16, Lemma 45].

▶ **Definition 30** (Omitting Types Property). *Let α be an infinite cardinal. The fragment $\mathcal{L}$ has α-Omitting Types Property (α-OTP) whenever*

1. *for all signatures Σ of power α,*
2. *all satisfiable sets of sentences $\Phi \subseteq \mathsf{Sen}(\Delta)$, and*
3. *all families of types $\{T_i \subseteq \mathsf{Sen}(\Sigma[X_i]) \mid i < \alpha\}$,*

such that Φ locally omits T_i for all $i < \alpha$, there exists a Σ-model of Φ which omits T_i for all $i < \alpha$. If $\alpha = \omega$ then we say that $\mathcal{L}$ has OTP rather than $\mathcal{L}$ has ω-OTP.

▶ **Theorem 31** (Omitting Types Theorem). *Let Σ be a signature of power α. If $\alpha > \omega$ then we assume that fragment $\mathcal{L}$ is compact. Then $\mathcal{L}$ has α-OTP.*

Any fragment $\mathcal{L}$ obtained from TA by (a) discarding a subset of sentence or/and action operators, and (b) restricting the category of signatures – without prohibiting signature extensions with constants – has ω-OTP. Any star-free fragment $\mathcal{L}$ of TA is compact and therefore has α-OTP for all cardinals cardinal $\alpha \geq \omega$. In particular, many-sorted first-order logic has α-OTP for all cardinals cardinal $\alpha \geq \omega$.

▶ **Lemma 32.** *Assume that $\mathcal{L}$ is closed under Boolean operators and first-order quantifiers. Let $\Sigma = (S, F)$ be a signature, where $S := \{s_n \mid n > 0\}$ and $F := \emptyset$.*

- *Let $\phi_n := \exists z_n \cdot \top \Rightarrow \exists x_1, \ldots, x_n \cdot \bigwedge_{i \neq j} x_i \neq x_j$ be a Σ-sentence, where $n > 0$, where z_n is a variable of sort s_n, and $x_1, \ldots, x_n$ are variables of sort s_1. Note that ϕ_n states that if there exists an element of sort s_n then there exists at least n elements of sort s_1.*
- *Let $T := \{\exists x_1, \ldots, x_n \cdot \bigwedge_{i \neq j} x_i \neq x_j \wedge \bigwedge_{i=1}^{n} y \neq x_i \mid n > 0\}$ be a type with one variable y of sort s_1, expressing that there are infinitely many elements of sort s_1.*

Then $\Phi := \{\phi_n \mid n > 0\}$ locally omits T.

Let Σ, Φ and T be as defined in Lemma 32. Recall that the set of Henkin constants is $C = \{C_{s_n}\}_{n<\omega}$, where $\mathsf{card}(C_{s_n}) = \omega$ for all $n < \omega$. Note that (Σ, Φ) is a satisfiable first-order presentation that does not involve the transitive closure operator $*$, as it contains no relation symbols. By Lemma 32, Φ locally omits T. By Theorem 31, there exists a model $\mathfrak{A}$ for Φ which omits T. Since every model of $(\Sigma[C], \Phi)$ realizes T, the model $\mathfrak{A}$ must be constructed over a proper subset of C, in contrast to the classical first-order logic approach, where results are developed over the entire $\Sigma[C]$. Consequently, the OTT as presented in [23, 3, 21, 14, 16] does not apply in this context.

▶ **Lemma 33.** *Let $\Sigma = (S, F)$ be a signature such that $S := \{s\}$ and $F := \{c :\to s \mid c \in \mathbb{R}\}$, where $\mathbb{R}$ denotes the set of real numbers. Let $T := \{x_i \neq c \mid c :\to s \in F \text{ and } i < \omega\}$ be a type with a countably infinite number of variables from $X := \{x_i \mid i < \omega\}$. Then Φ locally omits T, where Φ is any countable and satisfiable set of Σ-sentences.*

By Theorem 31, there exists a model $\mathfrak{A}$ of Φ which omits T. Consequently, the number of elements in $\mathfrak{A}$ that are not denotations of real numbers – considered unwanted elements – is finite and may therefore be regarded as negligible. Lemma 33 demonstrates that types involving infinitely many variables are not difficult to construct and can naturally arise in applications.

9 ω-Completeness

Although a transition algebra may satisfy a given set of sentences, it may not be relevant to the intended semantics. In many cases, formal methods practitioners are interested in the properties of a restricted class of transition algebras, such as: (a) those reachable through a set of constructor operators, (b) those with a finite number of elements, or (c) those satisfying both conditions. Using Omitting Types Theorem we can extend the completeness result in [19] to fragments obtained from $\mathcal{L}$ by restricting the semantics to the aforementioned transition algebras. Concretely, given a sound and complete proof system for $\mathcal{L}$, we introduce additional proof rules so that the resulting system remains sound and complete for fragments of $\mathcal{L}$ obtained by restricting the class of transition algebras to those of interest.

▶ **Definition 34** (Entailment relation). *An* entailment relation *is a family of binary relations between sets of sentences $\vdash = \{\vdash_\Sigma\}_{\Sigma \in |\mathsf{Sig}^{\mathcal{L}}|}$ with the following properties:*

$$(Monotonicity)\frac{\Phi_1 \subseteq \Phi_2}{\Phi_2 \vdash \Phi_1} \qquad (Transitivity)\frac{\Phi_1 \vdash \Phi_2 \qquad \Phi_2 \vdash \Phi_3}{\Phi_1 \vdash \Phi_3}$$

$$(Union)\frac{\Phi_1 \vdash \varphi_2 \text{ for all } \phi_2 \in \Phi_2}{\Phi_1 \vdash \Phi_2} \qquad (Translation)\frac{\Phi_1 \vdash_\Sigma \Phi_2}{\chi(\Phi_1) \vdash_{\Sigma'} \chi(\Phi_2)} \text{ where } \chi : \Sigma \to \Sigma'$$

An entailment relation $\vdash$ is sound (complete) if $\vdash \subseteq \models$ ($\models \subseteq \vdash$).

Completeness fails when $\mathcal{L}$ is closed under $*$ and includes uncountable signatures [19, Proposition 15]. Therefore, throughout this section, we assume that the fragment $\mathcal{L}$ has only countable signatures if it is closed under $*$. Additionally, we assume that $\mathcal{L}$ is semantically closed under Boolean operators.

Constructor-based algebras. Many algebraic structures can be naturally described using a set of constructor operators. For instance, numbers can be defined using a constant *zero* and a unary function symbol *succ*, while lists can be defined using a constant *nil* and a binary function symbol *cons*.

▶ **Definition 35** (Constructor-based algebras). *Let $\Sigma = (S, F, L)$ be a signature and $F^c \subseteq F$ a subset of constructor operators. We let Σ^c denote the signature of constructors (S, F^c). The constructors create a partition of the set of sorts S.*

1. *A sort $s \in S$ that has a constructor – i.e., there exists a constructor $(\sigma : w \to s) \in F^c$ – is called* constrained. *We denote the set of all constrained sorts by S^c.*
2. *A sort that is not constrained is called* loose. *We denote the set of all loose sorts by S^l.*

A constructor-based transition algebra $\mathfrak{A}$ *is a transition algebra for which there exist (a) a set of loose variable $Y = \{Y_s\}_{s \in S^l}$ and (b) an expansion $\mathfrak{B}$ of $\mathfrak{A}$ to $\Sigma[Y]$, such that $\mathfrak{B}\upharpoonright_{\Sigma^c[Y]}$ is reachable.*

▶ **Example 36** (Lists). Let $\Sigma = (S, F)$ be an algebraic signature such that (a) $S = \{Elt, List\}$ and (b) $F = \{empty :\to List, \_;\_ : List\ Elt \to List, add : List\ List \to List\}$. Let $F^c = \{empty :\to List, \_;\_ : List\ Elt \to List\}$ be a set of constructors.

- *Elt* is a loose sort while *List* is a constrained sort.
- Let $\mathfrak{A}$ be a constructor-based algebra obtained from $T_{\Sigma^c}(\omega)$ by interpreting *add* as follows: (a) $add^{\mathfrak{A}}(\ell, empty) = \ell$, and (b) $add^{\mathfrak{A}}(\ell, \ell'; n) = add^{\mathfrak{A}}(\ell, \ell'); n$ for all lists of natural numbers $\ell, \ell' \in \mathfrak{A}_{List}$ and all natural numbers $n \in \mathfrak{A}_{Elt}$.

By refining the syntax with a subset of constructor operators and restricting the semantics to constructor-based transition algebras, we obtain a new logic $\mathcal{L}^c$ from $\mathcal{L}$. This restriction in semantics alters the satisfaction relation $\models$ between sets of sentences: $\Phi \models^c \phi$ means that all constructor-based transition algebras of Φ are models of ϕ. Consequently, there may exist non-restricted models of Φ that do not satisfy ϕ. In order to obtain an institution, we need to restrict the class of signature morphisms such that the reduct of a constructor-based algebra is again a constructor-based algebra [17, 13, 2]. So, the category of signatures of $\mathcal{L}^c$ has

1. objects of the form $(S, F^c \subseteq F, L)$, where (S, F, L) is a signature in $\mathcal{L}$, and
2. signature morphisms of the form $\chi : (S, F^c \subseteq F, L) \to (S', F'^c \subseteq F', L')$, where (a) $\chi : (S, F, L) \to (S', F', L')$ is a signature morphism in $\mathcal{L}$, (b) constructors are preserved, that is, $\chi(F^c) \subseteq F'^c$, and (c) constructors are reflected, that is, for all $\sigma' : w' \to \chi(s) \in F'^c$ there exists $\sigma : w \to s \in F^c$ such that $\chi(\sigma : w \to s) = \sigma' : w' \to \chi(s)$.

Finite algebras. Computer science applications often focus on properties that hold in finite structures. Therefore, we refine the fragment $\mathcal{L}^c$ defined above to include constructor-based transition algebras, where certain distinguished sorts are interpreted as finite sets. Let $\mathcal{L}^f$ be a fragment of $\mathcal{L}^c$ obtained by (a) refining the syntax with a subset of sorts of finite domains, and (b) by restricting constructor-based models such that the sorts for finite domains are interpreted as finite sets. This means that the category of signatures in $\mathcal{L}^f$ has

1. objects of the form $(S^f \subseteq S, F^c \subseteq F, L)$, where $(S, F^c \subseteq F, L)$ is a signature in $\mathcal{L}^c$, and
2. arrows of the form $\chi : (S^f \subseteq S, F^c \subseteq F, L) \to (S'^f \subseteq S', F'^c \subseteq F', L')$ such that (a) $\chi : (S, F^c \subseteq F, L) \to (S', F'^c \subseteq F', L')$ is a signature morphism in $\mathcal{L}^c$, and (b) the finite sorts are preserved, that is, $\chi(S^f) \subseteq S'^f$.

▶ **Definition 37.** *Let $\vdash$ be an entailment relation for $\mathcal{L}$. The entailment relation $\vdash^f$ for $\mathcal{L}^f$ is the least entailment relation that includes $\vdash$ and it is closed under the following proof rules:*

1. (CB) $\dfrac{\Phi \vdash^f \forall var(t) \cdot \psi(t) \text{ for all } t \in T_{\Sigma^c}(Y)}{\Phi \vdash^f \forall x \cdot \psi(x)}$, *where (a) x is a variable of constrained sort, (b) Y is a S^l-sorted block of variables such that* $\mathsf{card}(Y_s) = \omega$ *for all loose sorts $s \in S^l$, and (c) $var(t)$ is the set of all variables occuring in t.*
2. (FN) $\dfrac{\Phi \cup \{\Gamma_{\overline{n}}\} \vdash^f \psi \text{ for all } \overline{n} \in \omega^\alpha}{\Phi \vdash^f \psi}$, *where*
   - **a.** $\alpha \coloneqq \mathsf{card}(S^f)$ *and $\{s_i\}_{i<\alpha}$ is an enumeration of S^f,*
   - **b.** $\Gamma_{\overline{n}}$ *denotes the set of sentences $\{\gamma_{s_i,\overline{n}(i)} \mid i < \alpha\}$ for all tuples $\overline{n} \in \omega^\alpha$, where*
     - $\overline{n}(i)$ *is the natural number in position i from $\overline{n}$, for all tuples $\overline{n} \in \omega^\alpha$ and all $i < \alpha$;*
     - $\gamma_{s,n}$ *denotes the sentence $\forall x_1, \dots, x_{n+1} \cdot \bigvee_{i \neq j} x_i = x_j$, for all natural numbers $n < \omega$, all sorts $s \in S^f$ and all variables $x_1, \dots, x_{n+1}$ of sort s.*

   Note that $\gamma_{s,n}$ states that the sort s has at most n elements. If $n = 0$ then $\gamma_{0,s} = \forall x_1 \cdot \bigvee \emptyset$, which is satisfied only by transition algebras whose carrier set for the sort s is empty.

(CB) asserts that to prove the property ψ for an arbitrary element x, it suffices to establish ψ for all constructor terms t in $T_{(S,F^c)}(Y)$. (CB) should be disregarded if the base logic $\mathcal{L}$ is not closed under first-order quantification. (FN) states that the number of elements of any sort in S^f is finite. It is straightforward to show that $\vdash^f$ is sound for $\mathcal{L}^f$ provided that $\vdash$ is sound for $\mathcal{L}$. Establishing completeness is generally more challenging, but it can be achieved with the help of the OTP.

▶ **Theorem 38** (Completeness)**.** *The entailment relation $\vdash^f$ is complete for $\mathcal{L}^f$ if $\vdash$ is complete for $\mathcal{L}$ and $\mathcal{L}$ has OTP.*

10 Conclusions

In this contribution, we focused on two main aspects of TA. First, we examined its model-theoretic properties, including the Löwenheim-Skolem and Omitting Types properties. Due to its increased expressivity, TA does not possess the Löwenheim-Skolem property. We then developed a semantic forcing property based on recent developments from [19], which enabled us to prove the Löwenheim-Skolem Theorem and the Omitting Types Theorem. Given that the full expressivity of TA is not required in many practical case studies, our study was conducted on a fragment of TA that restricts the syntax while preserving the semantics.

Secondly, we explored applications of the Omitting Types Theorem to formal methods. Software engineers often seek a restricted class of models for a given theory, such as constructor-based and/or finite models. We extended the system of proof rules proposed in [19] with new rules that are sound for these models of interest, ensuring that completeness is maintained. This approach provides sound and complete proof rules for fragments of TA obtained by restricting the syntax and/or the semantics.

An interesting future research direction involves developing specification and verification methodologies based on the proof rules we have defined for the fragments of TA. Additionally, we aim to propose a notion of Horn clauses for TA, which could make specifications defined in TA executable through rewriting or narrowing.

References

1 Michel Bidoit and Rolf Hennicker. Constructor-based observational logic. *J. Log. Algebraic Methods Program.*, 67(1-2):3–51, 2006. doi:10.1016/J.JLAP.2005.09.002.

2 Michel Bidoit, Rolf Hennicker, and Alexander Kurz. Observational logic, constructor-based logic, and their duality. *Theor. Comput. Sci.*, 3(298):471–510, 2003. doi:10.1016/S0304-3975(02)00865-4.

3 C. C. Chang and H. J. Keisler. *Model Theory*, volume 73 of *Studies in Logic and the Foundations of Mathematics*. North-Holland Publishing Co., Amsterdam, third edition, 1990.

4 Manuel Clavel, Francisco Durán, Steven Eker, Patrick Lincoln, Narciso Martí-Oliet, José Meseguer, and Carolyn L. Talcott, editors. *All About Maude – A High-Performance Logical Framework, How to Specify, Program and Verify Systems in Rewriting Logic*, volume 4350 of *LNCS*. Springer Berlin, Heidelberg, 2007. doi:10.1007/978-3-540-71999-1.

5 Paul J. Cohen. The independence of the continuum hypothesis. *Proceedings of the National Academy of Sciences of the United States of America*, 50(6):1143–1148, December 1963. doi:10.1073/pnas.50.6.1143.

6 Paul J. Cohen. The independence of the continuum hypothesis, II. *Proceedings of the National Academy of Sciences of the United States of America*, 51(1):105–110, January 1964. doi:10.1073/pnas.51.1.105.

7 Răzvan Diaconescu. *Institution-Independent Model Theory*. Studies in Universal Logic. Birkhäuser, 2008.

8 Răzvan Diaconescu and Kokichi Futatsugi. Logical foundations of CafeOBJ. *Theoretical Computer Science*, 285(2):289–318, 2002. doi:10.1016/S0304-3975(01)00361-9.

9 Ronald Fagin. Generalized First-Order Spectra and Polynomial-Time Recognizable Sets. In Richard M. Karp, editor, *Complexity of Computation*, volume 7 of *SIAM-AMS Proceedings*, pages 43–73. American Mathematical Society, 1974. doi:10.1090/psapm/007.

10 Ronald Fagin. Finite-model theory - a personal perspective. *Theoretical Computer Science*, 116:3–31, 1993. doi:10.1016/0304-3975(93)90218-I.

11 Joseph A. Goguen and Rod M. Burstall. Institutions: Abstract Model Theory for Specification and Programming. *J. ACM*, 39(1):95–146, 1992. doi:10.1145/147508.147524.

12 Erich Grädel, Phokion Kolaitis, Libkin G., Marx Leonid, Spencer Maarten, Vardi Joel, Y. Moshe, Yde Venema, and Scott Weinstein. *Finite Model Theory and its Applications*. Springer, 2007. doi:10.1007/3-540-68804-8.

13 Daniel Găină. Interpolation in logics with constructors. *Theor. Comput. Sci.*, 474:46–59, 2013. doi:10.1016/J.TCS.2012.12.002.

14 Daniel Găină. Forcing, Downward Löwenheim-Skolem and Omitting Types Theorems, Institutionally. *Logica Universalis*, 8(3-4):469–498, 2014. doi:10.1007/S11787-013-0090-0.

15 Daniel Găină. Forcing and Calculi for Hybrid Logics. *Journal of the Association for Computing Machinery*, 67(4):1–55, 2020. doi:10.1145/3400294.

16 Daniel Găină, Guillermo Badia, and Tomasz Kowalski. Omitting types theorem in hybrid dynamic first-order logic with rigid symbols. *Annals of Pure and Applied Logic*, 174(3):103212, 2023. doi:10.1016/J.APAL.2022.103212.

17 Daniel Găină, Kokichi Futatsugi, and Kazuhiro Ogata. Constructor-based Logics. *Journal of Universal Computer Science*, 18(16):2204–2233, August 2012. doi:10.3217/JUCS-018-16-2204.

18 Daniel Găină and Marius Petria. Completeness by forcing. *Journal of Logic and Computation*, 20(6):1165–1186, 2010. doi:10.1093/LOGCOM/EXQ012.

19 Go Hashimoto, Daniel Găină, and Ionut Ţuţu. Forcing, Transition Algebras, and Calculi. In Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson, editors, *51st International Colloquium on Automata, Languages, and Programming, ICALP 2024, July 8-12, 2024, Tallinn, Estonia*, volume 297 of *LIPIcs*, pages 143:1–143:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.ICALP.2024.143.

20 Go Hashimoto and Daniel Găină. Model-theoretic forcing in transition algebra, 2025. arXiv:2506.22828.

21 Wilfrid Hodges. *A Shorter Model Theory.* Cambridge University Press, Cambridge, UK, 1997.

22 H. Jerome Keisler and Arnold W. Miller. Categoricity without equality. *Fundamenta Mathematiae*, 170:87—-106, 2001.

23 H.J. Keisler. Forcing and the omitting types theorem. In M.D. Morley, editor, *Studies in Model Theory, MAA Studies in Mathematics*, volume 8, pages 96–133, 1973.

24 Leonid Libkin. *Elements of Finite Model Theory.* Texts in Theoretical Computer Science. Springer, Berlin, Heidelberg, 2004. doi:10.1007/978-3-662-07003-1.

25 José Meseguer. Conditional rewriting logic: Deduction, models and concurrency. In Stéphane Kaplan and Mitsuhiro Okada, editors, *Conditional and Typed Rewriting Systems, 2nd International CTRS Workshop, Montreal, Canada, June 11-14, 1990, Proceedings*, volume 516 of *LNCS*, pages 64–91. Springer, 1990. doi:10.1007/3-540-54317-1_81.

26 Marius Petria. An Institutional Version of Gödel's Completeness Theorem. In Till Mossakowski, Ugo Montanari, and Magne Haveraaen, editors, *Algebra and Coalgebra in Computer Science, Second International Conference, CALCO 2007, Bergen, Norway, August 20-24, 2007, Proceedings*, volume 4624 of *LNCS*, pages 409–424. Springer, 2007. doi:10.1007/978-3-540-73859-6_28.

27 Adrián Riesco, Kazuhiro Ogata, Masaki Nakamura, Daniel Găină, Duong Dinh Tran, and Kokichi Futatsugi. Proof scores: A survey. *ACM Comput. Surv.*, April 2025. doi:10.1145/3729166.

28 Abraham Robinson. Forcing in model theory. *Symposia Mathematica*, 5:69–82, 1971.

29 Donald Sannella and Andrzej Tarlecki. *Foundations of Algebraic Specification and Formal Software Development.* Monographs in Theoretical Computer Science. An EATCS Series. Springer, 2012. doi:10.1007/978-3-642-17336-3.