Second-Order Equational Logic

(Extended Abstract)

Marcelo Fiore[1] and Chung-Kil Hur[2⋆]

[1] Computer Laboratory, University of Cambridge
[2] Laboratoire PPS, Université Paris 7

Abstract. We extend universal algebra and its equational logic from first to second order as follows.

1. We consider second-order equational presentations as specified by identities between second-order terms, with both variables and parameterised metavariables over signatures of variable-binding operators.
2. We develop an algebraic model theory for second-order equational presentations, generalising the semantics of (first-order) algebraic theories and of (untyped and simply-typed) lambda calculi.
3. We introduce a deductive system, *Second-Order Equational Logic*, for reasoning about the equality of second-order terms. Our development is novel in that this equational logic is synthesised from the model theory. Hence it is necessarily sound.
4. *Second-Order Equational Logic* is shown to be a conservative extension of Birkhoff's *(First-Order) Equational Logic*.
5. Two completeness results are established: the semantic completeness of equational derivability, and the derivability completeness of (bidirectional) *Second-Order Term Rewriting*.

1 Introduction

The notion of algebraic structure has solid mathematical foundations. In the traditional, first order, case our understanding of the subject is complete; allowing us to look at it from three different perspectives: universal algebra, equational logic, and categorical algebra. Of direct concern to us in this paper is the relationship between the first two.

Universal algebra provides a model theory for algebraic structure and equational logic a formal deductive system for reasoning about it. These are related by Birkhoff's theorem [4] establishing the semantic soundness and completeness of equational deduction (see Goguen and Meseguer [16] for the many-sorted case). The theory of computation also plays a role here: (bidirectional) term rewriting provides a sound and complete computational method for establishing equational derivability (see *e.g.* [3]).

We are interested in this paper in extending the above fundamental theory from first to second order, *i.e.* to languages with variable binding. Such formalisms arise in a wide range of subjects: category theory (*e.g.* ends), logic (*e.g.*

⋆ Supported by Digiteo/Ile-de-France project COLLODI (2009-28HD).

quantifiers), mathematics (*e.g.* integration), process calculi (*e.g.* restriction), programming languages (*e.g.* local scope), type theory (*e.g.* lambda calculi).

A central theme of our development is thus to set up the algebraic model theory and the formal deductive system that underlie higher-order equational theories. As in the first-order case, the model theory should provide general algebraic semantics from which syntactic models are to arise as free algebras. In addition, the deductive system should elucidate higher-order equational reasoning. This requirement rules out any system based on a higher-order metalanguage.

Syntactically, the passage from first to second order involves extending the language with both variable-binding operators and parameterised metavariables. These two concepts are orthogonal to each other, but it is with both of them in place that the language attains the required expressiveness. Variable-binding operators may bind a list of variables in each of their arguments and thereby lead to syntax up to alpha equivalence. Parameterised metavariables are, in effect, second-order variables for which substitution also involves instantiation. As far as we are aware, such second-order syntax was first put forward by Aczel [1]. (A variation of it incorporating abstractions features in the *CRSs* of Klop [21].)

A mathematical theory of second-order syntax was developed by Fiore [11], building on work of Fiore, Plotkin and Turi [15] and of Hamana [18]. In it, second-order syntax is abstractly characterised by free algebras of a term monad on a suitable semantic universe. This provides initial-algebra semantics, induction principles, and structural recursion. Moreover, the crucial result that term monads are strong provides a second-order substitution calculus, see [11, Part I].

Term monads for second-order syntax (being strong with respect to a biclosed action) fit into the mathematical framework of Fiore and Hur [13, 7, 19] for synthesising equational logics. This framework provides a canonical algebraic model theory for categorical equational presentations (in the form of sets of parallel pairs of Kleisli maps) together with a sound categorical equational metalogic for reasoning about them.

The gist of the work in this paper is then to instantiate our framework and apply the supporting methodology to: (*i*) derive a universal algebra for second-order equational presentations; (*ii*) synthesise a sound equational logic for reasoning about them; and (*iii*) relate the two by means of a completeness theorem.

Our work initiates thus the development of *Second-Order Universal Algebra*, which generalises the model theory of (first-order) algebraic theories and of (untyped and simply-typed) lambda calculi. The associated *Second-Order Equational Logic* is distilled from a categorical *Equational Metalogic*, but thereafter can be understood completely independent of it. Besides the rules for axioms and equivalence, it consists of just one additional rule stating that the operation of metavariable substitution in extended variable contexts is a congruence. At the level of equational derivability, the relationship between universal algebra and our second-order extension translates as a conservative-extension result. We further establish the semantic completeness of equational derivability, and the derivability completeness of (bidirectional) *Second-Order Term Rewriting*. These results firmly establish *Second-Order Universal Algebra*, *Second-Order*

Equational Logic, and *Second-Order Term Rewriting* as the model theory, proof theory, and rewriting theory of higher-order equational theories. (In particular, we provide model-theoretic foundations for *CRSs*; finally answering question (f) in [22, Section 15].)

In addition, we note that the perspective to first-order algebraic structure offered by Lawvere theories [23] has also been extended to second-order by Fiore and Mahmoud, see the companion paper [14].

Concerning related work, equational deductive systems for reasoning about algebraic structure with binding have already been considered in the literature. The systems closest to ours are the *Equational Logic for Binding Algebras* of Sun [27, 28] and the *Multi-Sorted Binding Equational Logic* of Plotkin [26]. These somehow sit in between our *Second-Order Equational Logic* and our bidirectional *Second-Order Term Rewriting.* The core of Sun's system consists of a substitution rule for variables and metavariables, and congruence rules for metavariables and operators. A major difference appears in the model theory, which Sun restricts to functional models (as in Aczel's *Frege Structures* [2]) that do not support free constructions and lead to a restricted completeness result. On the other hand, the system outlined by Plotkin shares the same syntax with ours, but it is also set up with congruence rules for metavariables and operators, and cut rules for variables and metavariables. Plotkin also considers general abstract models that axiomatise the algebraic structure of functional concrete models and are able to encompass initial syntactic models.

Another early system is the *Abstract Variable Binding Calculus* of Pigozzi and Salibra [25]. A strong point of departure between this system and ours is that in it metavariables are treated informally in the metalanguage. This results in the deductive system not being a true equational theory. More recently, there have been the *Equational Logic for Binding Terms (ELBT)* of Hamana [17], the *Nominal Algebra* of Gabbay and Mathijssen [24], and the *Nominal Equational Logic* of Clouston and Pitts [6]. The nominal systems have been shown to correspond to the *Synthetic Nominal Equational Logic (SNEL)* of Fiore and Hur [13] with metavariables that can be solely parameterised by names. The only essential difference between *SNEL* and *ELBT* is that the latter lacks a rule for atom elimination.

2 Syntactic Theory

The syntactic theory underlying *Second-Order Equational Logic* is introduced. The development comprises second-order signatures on top of which second-order terms in context are defined. For these the needed two-level substitution calculus is presented.

Signatures. A (second-order) *signature* $\Sigma = (T, O, |-|)$ is specified by a set of types T, a set of operators O, and an arity function $|-| : O \longrightarrow (T^* \times T)^* \times T$. This definition is a typed version of the binding signatures of Aczel [1] (see also [22, 15]).

Notation. We let $|\vec{\sigma}|$ be the length of a sequence $\vec{\sigma}$. For $1 \leq i \leq |\vec{\sigma}|$, we let σ_i be the i^{th} element of $\vec{\sigma}$; so that $\vec{\sigma} = \sigma_1, \ldots, \sigma_{|\vec{\sigma}|}$.

For an operator $\mathsf{o} \in O$, we typically write $\mathsf{o} : (\vec{\sigma_1})\tau_1, \ldots, (\vec{\sigma_n})\tau_n \longrightarrow \tau$ whenever $|\mathsf{o}| = \big((\vec{\sigma_1}, \tau_1) \ldots (\vec{\sigma_n}, \tau_n), \tau\big)$. The intended meaning here is that o is an operator of type τ taking n arguments each of which binds $n_i = |\vec{\sigma_i}|$ variables of types $\sigma_{i,1}, \ldots, \sigma_{i,n_i}$ in a term of type τ_i.

The second-order signature of the λ-calculus [1] is given below. Further examples already spelled out in the literature are the primitive recursion operator [1], the quantifiers [2], the fixpoint operator [22], and the list iterator [29]. In fact, any language with variable binding fits the formalism.

Example 1. The signature of the *typed λ-calculus* over a set of base types B has set of types $B^{\Rightarrow}$ given by

$$\frac{\beta \in B}{\beta \in B^{\Rightarrow}} \qquad\qquad \frac{\sigma, \tau \in B^{\Rightarrow}}{\sigma \Rightarrow \tau \in B^{\Rightarrow}}$$

and, for $\sigma, \tau \in B^{\Rightarrow}$, operators $\mathsf{abs}^{\sigma,\tau} : (\sigma)\tau \longrightarrow \sigma \Rightarrow \tau$ and $\mathsf{app}^{\sigma,\tau} : \sigma \Rightarrow \tau, \sigma \longrightarrow \tau$.

The signature of the *untyped λ-calculus* is as above when only one type, say D, is available. Hence, it has operators $\mathsf{abs} : (\mathsf{D})\mathsf{D} \longrightarrow \mathsf{D}$ and $\mathsf{app} : \mathsf{D}, \mathsf{D} \longrightarrow \mathsf{D}$.

Contexts. We will consider terms in typing contexts. Typing contexts have two zones, each respectively typing variables and metavariables. Variable typings are types. Metavariable typings are parameterised types: a metavariable of type $[\sigma_1, \ldots, \sigma_n]\tau$, when parameterised by terms of type $\sigma_1, \ldots, \sigma_n$, will yield a term of type τ. In accordance, we use the following representation for typing contexts: $\mathrm{M}_1 : [\vec{\sigma_1}]\tau_1, \ldots, \mathrm{M}_k : [\vec{\sigma_k}]\tau_k \rhd x_1 : \sigma'_1, \ldots, x_\ell : \sigma'_\ell$, where all metavariables and all variables are assumed distinct.

Terms. Signatures give rise to terms. These are built up by means of operators from both variables and metavariables, and hence referred to as second-order.

Terms are considered up the α-equivalence relation induced by stipulating that, for every operator o, in the term $\mathsf{o}\big(\ldots, (\vec{x_i})t_i, \ldots\big)$ the $\vec{x_i}$ are bound in t_i. This may be formalised in a variety of ways, but it is not necessary for us to do so here.

The judgement for *terms* in context $\Theta \rhd \Gamma \vdash - : \tau$ is defined by the rules below. This definition is a typed version of the second-order syntax of Aczel [1].

(Variables) For $(x : \tau) \in \Gamma$,

$$\frac{}{\Theta \rhd \Gamma \vdash x : \tau} \tag{1}$$

(Metavariables) For $(\mathrm{M} : [\tau_1, \ldots, \tau_n]\tau) \in \Theta$,

$$\frac{\Theta \rhd \Gamma \vdash t_i : \tau_i \ (1 \leq i \leq n)}{\Theta \rhd \Gamma \vdash \mathrm{M}[t_1, \ldots, t_n] : \tau} \tag{2}$$

(Operators) For $\mathsf{o} : (\vec{\sigma_1})\tau_1, \ldots, (\vec{\sigma_n})\tau_n \longrightarrow \tau$,

$$\frac{\Theta \rhd \Gamma, \vec{x_i} : \vec{\sigma_i} \vdash t_i : \tau_i \ (1 \leq i \leq n)}{\Theta \rhd \Gamma \vdash \mathsf{o}\big((\vec{x_1})\, t_1, \ldots, (\vec{x_n})\, t_n\big) : \tau} \tag{3}$$

where $\vec{x} : \vec{\sigma}$ stands for $x_1 : \sigma_1, \ldots, x_k : \sigma_k$.

Example 2. Two sample terms for the signature of the typed λ-calculus follow:

$$\mathrm{M} : [\sigma]\tau, \mathrm{N} : \sigma \rhd \cdot \vdash \mathsf{app}\big(\mathsf{abs}\big((x)\mathrm{M}[x]\big), \mathrm{N}[\,]\big) : \tau \enspace ,$$
$$\mathrm{M} : [\sigma]\tau, \mathrm{N} : \sigma \rhd \cdot \vdash \mathrm{M}[\mathrm{N}[\,]] : \tau \enspace .$$

Substitution calculus. The second-order nature of the syntax requires a two-level substitution calculus [1, 22, 29, 11]. Each level respectively accounts for the substitution of variables and metavariables, with the latter operation depending on the former.

The operation of *capture-avoiding simultaneous substitution* of terms for variables maps

$$\Theta \rhd x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash t : \tau \quad \text{and} \quad \Theta \rhd \Gamma \vdash t_i : \sigma_i \;\; (1 \le i \le n)$$

to

$$\Theta \rhd \Gamma \vdash t[{}^{t_i}\!/\!_{x_i}]_{1 \le i \le n} : \tau$$

according to the following definition:

- $x_j[{}^{t_i}\!/\!_{x_i}]_{1 \le i \le n} = t_j$
- $\big(\mathrm{M}[\ldots, s, \ldots]\big)[{}^{t_i}\!/\!_{x_i}]_{1 \le i \le n} = \mathrm{M}\big[\ldots, s[{}^{t_i}\!/\!_{x_i}]_{1 \le i \le n}, \ldots\big]$
- $\big(\mathsf{o}(\ldots, (y_1, \ldots, y_k)s, \ldots)\big)[{}^{t_i}\!/\!_{x_i}]_{1 \le i \le n}$
  $= \mathsf{o}\big(\ldots, (z_1, \ldots, z_k)s[{}^{t_i}\!/\!_{x_i}, {}^{z_j}\!/\!_{y_j}]_{1 \le i \le n, 1 \le j \le k}, \ldots\big)$
  with $z_j \notin \mathrm{dom}(\Gamma)$ for all $1 \le j \le k$

The operation of *metasubstitution* of abstracted terms for metavariables maps

$$\mathrm{M}_1 : [\vec{\sigma_1}]\tau_1, \ldots, \mathrm{M}_k : [\vec{\sigma_k}]\tau_k \rhd \Gamma \vdash t : \tau \;\; \text{and} \;\; \Theta \rhd \Gamma, \vec{x_i} : \vec{\sigma_i} \vdash t_i : \tau_i \;\; (1 \le i \le k)$$

to

$$\Theta \rhd \Gamma \vdash t\{\mathrm{M}_i := (\vec{x_i})t_i\}_{1 \le i \le k} : \tau$$

according to the following definition:

- $x\{\mathrm{M}_i := (\vec{x_i})t_i\}_{1 \le i \le k} = x$
- $\big(\mathrm{M}_\ell[s_1, \ldots, s_m]\big)\{\mathrm{M}_i := (\vec{x_i})t_i\}_{1 \le i \le k} \; = t_\ell[{}^{s'_j}\!/\!_{x_{i,j}}]_{1 \le j \le m}$
  where, for $1 \le j \le m$, $s'_j = s_j\{\mathrm{M}_i := (\vec{x_i})t_i\}_{1 \le i \le k}$
- $\big(\mathsf{o}(\ldots, (\vec{x})s, \ldots)\big)\{\mathrm{M}_i := (\vec{x_i})t_i\}_{1 \le i \le k} \; = \; \mathsf{o}\big(\ldots, (\vec{x})s\{\mathrm{M}_i := (\vec{x_i})t_i\}_{1 \le i \le k}, \ldots\big)$

The syntactic theory can be completely justified on model-theoretic grounds, see [11]. We turn to this next.

3 Abstract Syntactic Theory

Having developed the second-order syntactic theory, our purpose in this section is to show how it arises from a model theory. This is important for several reasons: it provides an abstract characterisation of syntax by free constructions and thereby supports initial-algebra semantics and definitions by structural recursion; it encompasses and guarantees all the necessary properties of the substitution calculus; and it opens up the development of an algebraic model theory for second-order equational presentations together with an associated equational logic.

We now introduce the semantic universe and, within it, constructions for modelling substitution and algebraic structure. These lead to a canonical notion of model for second-order signatures [15, 10]. The syntactic nature of free models is then explained.

Notation. For a sequence $\vec{\sigma}$, we let $[\vec{\sigma}] = \{ 1, \ldots, |\vec{\sigma}| \}$.

Semantic universe. For a set T, we write $\mathbb{F}[T]$ for the free cocartesian category on T. Explicitly, it has set of objects T^* and morphisms $\vec{\sigma} \longrightarrow \vec{\tau}$ given by functions $\rho : [\vec{\sigma}] \longrightarrow [\vec{\tau}]$ such that $\sigma_i = \tau_{\rho i}$ for all $i \in [\vec{\sigma}]$.

For a set of types T, we will work within and over the semantic universe $\big(\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]}\big)^T$ of T-sorted sets in T-typed contexts [8]. We write $\boldsymbol{y}$ for the Yoneda embedding $\mathbb{F}[T]^{\mathrm{op}} \hookrightarrow \boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]}$.

Substitution. We recall the *substitution monoidal structure* in semantic universes [9]. It has tensor unit and tensor product respectively given by $\mathrm{V}_\tau = \boldsymbol{y}(\tau)$ and $(X \bullet Y)_\tau = X_\tau \circledbullet Y$ where $P \circledbullet Y = \int^{\vec{\sigma} \in \mathbb{F}[T]} P(\vec{\sigma}) \times \prod_{i \in [\vec{\sigma}]} Y_{\sigma_i}$.

A monoid $\mathrm{V} \longrightarrow A \longleftarrow A \bullet A$ for the substitution monoidal structure equips A with substitution structure. In particular, the components $\boldsymbol{y}(\gamma) \longrightarrow A_\gamma$ of the unit induce the embedding

$$\big(A_\tau{}^{\boldsymbol{y}(\vec{\sigma})} \times \textstyle\prod_{i \in [\vec{\sigma}]} A_{\sigma_i}\big)(\vec{\gamma}) \longrightarrow A_\tau(\vec{\gamma}, \vec{\sigma}) \times \prod_{j \in [\vec{\gamma}]} A_{\gamma_j}(\vec{\gamma}) \times \prod_{i \in [\vec{\sigma}]} A_{\sigma_i}(\vec{\gamma}) \longrightarrow \big(A_\tau \circledbullet A\big)(\vec{\gamma})$$

which together with the component $A_\tau \circledbullet A \longrightarrow A_\tau$ of the multiplication yield a *substitution operation*

$$\varsigma_{\vec{\sigma},\tau} : A_\tau{}^{\boldsymbol{y}(\vec{\sigma})} \times \textstyle\prod_{i \in [\vec{\sigma}]} A_{\sigma_i} \longrightarrow A_\tau \ .$$

These substitution operations provide the interpretation of metavariables.

The category of monoids for the substitution tensor product is isomorphic to that of T-sorted Lawvere theories and maps. The Lawvere theory $\mathbb{L}_A$ associated to A has objects T^* and hom-sets $\mathbb{L}_A(\vec{\sigma}, \vec{\tau}) = \prod_{i \in [\vec{\tau}]} A_{\tau_i}(\vec{\sigma})$, with identities and composition provided by the monoid structure. On the other hand, for every cartesian category $\mathscr{C}$ and assignment $C \in \mathscr{C}^T$ consider the functor

$$\langle C, - \rangle : \mathscr{C}^T \longrightarrow \big(\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]}\big)^T \tag{4}$$

defined as $\langle C, D\rangle_\tau = \langle\!\langle C, D_\tau \rangle\!\rangle$ with $\langle\!\langle C, d \rangle\!\rangle(\vec{\sigma}) = \mathscr{C}\big(\prod_{1 \le i \le |\vec{\sigma}|} C_{\sigma_i}, d\big)$. Then, $\langle C, C\rangle$ has a canonical monoid structure given by projections and composition.

Algebras. Every signature Σ over a set of types T induces a *signature endofunctor* on $(\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]})^T$ given by $(\underline{\Sigma} X)_\tau = \coprod_{\mathsf{o} : (\vec{\sigma_1})\tau_1, \ldots, (\vec{\sigma_n})\tau_n \to \tau} \prod_{1 \le i \le n} X_{\tau_i}{}^{\boldsymbol{y}(\vec{\sigma_i})}$. $\underline{\Sigma}$-algebras provide interpretations for the operators of Σ.

We note that there are canonical natural isomorphisms

$$\textstyle\coprod_{i \in I} (X_i \bullet Y) \cong \big(\coprod_{i \in I} X_i\big) \bullet Y$$

$$\big(\textstyle\prod_{1 \le i \le n} X_i\big) \bullet Y \cong \prod_{1 \le i \le n} (X_i \bullet Y)$$

and, for all points $\nu : \mathrm{V} \longrightarrow Y$, a natural extension map

$$\nu^\# : P^{\boldsymbol{y}(\vec{\sigma})} \circledbullet Y \longrightarrow (P \circledbullet Y)^{\boldsymbol{y}(\vec{\sigma})} \ .$$

These constructions equip every signature endofunctor with a *pointed strength*

$$\varpi_{X,\mathrm{V}\to Y} : \underline{\Sigma}(X) \bullet Y \longrightarrow \underline{\Sigma}(X \bullet Y) \enspace .$$

(See [11] for details.)

Models. The models that we are interested in (referred to as Σ-*monoids* in [15, 11]) are algebras equipped with a compatible substitution structure.

For a signature Σ over a set of types T, we let Σ-Mod be the category of Σ-*models* with objects $A \in (\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]})^T$ equipped with a $\underline{\Sigma}$-algebra structure $\underline{\Sigma}A \longrightarrow A$ and a monoid structure $\mathrm{V} \longrightarrow A \longleftarrow A \bullet A$ that are compatible in the sense that the diagram

$$\begin{array}{ccccc} \underline{\Sigma}(A) \bullet A & \xrightarrow{\varpi_{A,\mathrm{V}\to A}} & \underline{\Sigma}(A \bullet A) & \longrightarrow & \underline{\Sigma}(A) \\ \downarrow & & & & \downarrow \\ A \bullet A & & \longrightarrow & & A \end{array}$$

commutes. Morphisms are maps that are both $\underline{\Sigma}$-algebra and monoid homomorphims.

Term monad. The forgetful functor Σ-$\mathsf{Mod} \longrightarrow \big(\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]}\big)^T$ is monadic [11, 12]. Hence, writing $\mathcal{M}$ for the induced monad and $\mathcal{M}$-Alg for its category of Eilenberg-Moore algebras, we have a canonical isomorphism Σ-$\mathsf{Mod} \cong \mathcal{M}$-$\mathsf{Alg}$.

Carriers of free models can be explicitly described as initial algebras:

$$\mathcal{M}(X) = \mu Z.\, \mathrm{V} + X \bullet Z + \underline{\Sigma} Z \enspace .$$

Free models on objects arising from metavariable contexts have a syntactic description [18, 11]. Indeed, for $\Theta = \big(\mathrm{M}_1 : [\vec{\sigma_1}]\tau_1, \ldots, \mathrm{M}_k : [\vec{\sigma_k}]\tau_k\big)$ a metavariable context, let $\underline{\Theta} = \coprod_{1 \leq i \leq k} \boldsymbol{y}(\vec{\sigma_i})_{@\tau_i}$ in $\big(\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]}\big)^T$, where $(P_{@\tau})_\alpha$ is P for $\alpha = \tau$ and 0 otherwise. Then, using that

$$\underline{\Theta} \bullet Z \;\cong\; \textstyle\coprod_{1 \leq i \leq k} \big(\prod_{1 \leq j \leq |\vec{\sigma_i}|} Z_{\sigma_{i,j}}\big)_{@\tau_i} \enspace ,$$

the initial algebra structure

$$\mathrm{V} + \underline{\Theta} \bullet \mathcal{M}\underline{\Theta} + \underline{\Sigma}\mathcal{M}\underline{\Theta} \xrightarrow{\cong} \mathcal{M}\underline{\Theta}$$

corresponds to the rules in (1–3), and we have the following syntactic characterisation

$$(\mathcal{M}\underline{\Theta})_\tau(\vec{\sigma}) \cong \{\, t \mid \Theta \rhd \vec{x} : \vec{\sigma} \vdash t : \tau \,\} \enspace . \tag{5}$$

We thus refer to $\mathcal{M}$ as the *term monad*.

The two-level substitution calculus arises as follows. The monoid multiplication $\mathcal{M}(\underline{\Theta}) \bullet \mathcal{M}(\underline{\Theta}) \to \mathcal{M}(\underline{\Theta})$, for Θ a metavariable context, amounts to the operation of capture-avoiding simultaneous substitution. On the other hand, the term monad comes equipped with a strength $\mathcal{M}(X) \otimes P \longrightarrow \mathcal{M}(X \otimes P)$ where $(X \otimes P)_\tau = X_\tau \times P$. Thereby, every model $\mathcal{M}A \longrightarrow A$ admits an interpretation map $\mathcal{M}(X) \otimes [X, A] \longrightarrow \mathcal{M}\big(X \otimes [X, A]\big) \longrightarrow \mathcal{M}(A) \longrightarrow A$ where $[X, Y] = \prod_{\tau \in T} {Y_\tau}^{X_\tau}$. In particular, for metavariable contexts Θ and Ξ, the interpretation map $\mathcal{M}(\underline{\Theta}) \otimes [\underline{\Theta}, \mathcal{M}(\underline{\Xi})] \longrightarrow \mathcal{M}(\underline{\Xi})$ amounts to the operation of metasubstitution. (See [11] for details.)

(Axiom)

$$\frac{(f, g : X \longrightarrow TY) \in E}{E \vdash f \equiv g : X \longrightarrow TY}$$

(Equivalence)

$$\frac{f : X \longrightarrow TY}{E \vdash f \equiv f : X \longrightarrow TY} \qquad \frac{E \vdash f \equiv g : X \longrightarrow TY}{E \vdash g \equiv f : X \longrightarrow TY} \qquad \frac{\begin{array}{c}E \vdash f \equiv g : X \longrightarrow TY \\ E \vdash g \equiv h : X \longrightarrow TY\end{array}}{E \vdash f \equiv h : X \longrightarrow TY}$$

(Composition)

$$\frac{E \vdash f_1 \equiv g_1 : X \longrightarrow TY \qquad E \vdash f_2 \equiv g_2 : Y \longrightarrow TZ}{E \vdash f_1[f_2] \equiv g_1[g_2] : X \longrightarrow TZ}$$

where, for $f : X \longrightarrow TY$ and $g : Y \longrightarrow TZ$, $f[g]$ is the Kleisli composite $X \xrightarrow{f} TY \xrightarrow{Tg} TTY \longrightarrow TY$

(Parameterisation)

$$\frac{E \vdash f \equiv g : X \longrightarrow TY}{E \vdash f\langle P \rangle \equiv g\langle P \rangle : X \otimes P \longrightarrow T(Y \otimes P)}$$

where, for $h : X \longrightarrow TY$, $h\langle P \rangle = \big(X \otimes P \xrightarrow{h \otimes \mathrm{id}} T(Y) \otimes P \longrightarrow T(Y \otimes P) \big)$

(Local character)

$$\frac{E \vdash f\, e_i \equiv g\, e_i : X_i \longrightarrow TY \quad (i \in I)}{E \vdash f \equiv g : X \longrightarrow TY} \ \big(\{e_i : X_i \longrightarrow X\}_{i \in I} \text{ jointly epi} \big)$$

Fig. 1. *Equational Metalogic.*

4 Equational Metalogic

Our aim now is to use the above monadic model theory for second-order syntax to synthesise a *Second-Order Equational Logic*. This development, which is presented in Section 5, depends on a general theory and methodology of the authors [13, 7, 19]. For the sake of completeness, an outline of the framework follows.

To every strong monad T with respect to a biclosed action [20] we associate an *Equational Metalogic*. This is a deductive system for reasoning about the equality of the interpretation of Kleisli maps $X \longrightarrow TY$ in Eilenberg-Moore algebras $TA \longrightarrow A$ as captured by the following satisfaction relation:

$$A \models f \equiv g : X \longrightarrow TY \text{ iff } [\![f]\!] = [\![g]\!] : X \otimes [Y, A] \longrightarrow A$$

where

$$[\![h]\!] = \big(X \otimes [Y, A] \xrightarrow{h \otimes \mathrm{id}} T(Y) \otimes [Y, A] \to A \big) \ . \tag{6}$$

Equational metalogic is parameterised by a set of axioms E given by parallel pairs of Kleisli maps. The rules assert the derivability of judgements of the form $E \vdash f \equiv g : X \longrightarrow TY$ and are given in Figure 1.

Remark. In the presence of coproducts, the rule

(Local parameterised composition)

$$\frac{\begin{array}{c} E \vdash f \equiv g : X \longrightarrow T\big(\coprod_{i\in I} Y_i\big) \\ E \vdash f_i \equiv g_i : Y_i \otimes P \longrightarrow TZ \quad (i \in I) \end{array}}{E \vdash f\langle P\rangle\big[\,[f_i]_{i\in I}\,\big] \equiv g\langle P\rangle\big[\,[g_i]_{i\in I}\,\big] : X \otimes P \longrightarrow T(Z)}$$

is derivable, and may be used instead of the (Composition) and (Parameterisation) rules.

The category (T, E)-Alg of (T, E)*-algebras* is defined as the full subcategory of the category T-Alg of Eilenberg-Moore algebras that satisfy the axioms E. We have the following two important results.

(Soundness) If $E \vdash f \equiv g$ then $A \models f \equiv g$ for all (T, E)-algebras A.

(Internal completeness) If every object Z admits a free (T, E)-algebra $T_E(Z)$, then we have a quotient map $q : T \longrightarrow T_E$ and the following are equivalent:

(1) $A \models f \equiv g : X \longrightarrow TY$ for all (T, E)-algebras A

(2) $T_E(Y) \models f \equiv g : X \longrightarrow TY$

(3) $q_Y\, f = q_Y\, g : X \longrightarrow T_E(Y)$

5 Second-Order Equational Logic

Second-Order Equational Logic is now synthesised from *Equational Metalogic*. This is done by: (i) considering the term monad $\mathcal{M}$; (ii) restricting attention to Kleisli maps of the form $\boldsymbol{y}(\vec{\sigma})_{@\tau} \longrightarrow \mathcal{M}(\underline{\Theta})$, which by (5) amount to second-order terms of type τ in context $\Theta \rhd \vec{x} : \vec{\sigma}$; and (iii) rendering the rules in syntactic form.

Presentations. An *equational presentation* is a set of axioms each of which is a pair of terms in context.

Example 3. The equational presentation of the typed λ-calculus follows.

$$(\beta)\ \mathrm{M} : [\sigma]\tau, \mathrm{N} : [\,]\sigma \rhd \cdot \vdash \mathsf{app}\big(\mathsf{abs}(\,(x)\mathrm{M}[x]\,), \mathrm{N}[\,]\big) \equiv \mathrm{M}\big[\mathrm{N}[\,]\big] : \tau$$

$$(\eta)\ \mathrm{F} : [\,](\sigma \Rightarrow \tau) \rhd \cdot \vdash \mathsf{abs}\big(\,(x)\mathsf{app}(\mathrm{F}[\,], x)\,\big) \equiv \mathrm{F}[\,] : \sigma \Rightarrow \tau$$

On top of the second-order equational presentation of the typed λ-calculus, one can then formalise any higher-order equational theory (like, for instance, equational axiomatisations of Church's *Simple Theory of Types* [5]) as an extended second-order equational presentation. We emphasise, however, that the expressiveness of our formalism does not rely on that of lambda calculi. For instance, one can directly axiomatise primitive recursion [1], predicate logic [26], and integration [25] as second-order equational presentations.

Logic. The rules of *Second-Order Equational Logic* are given in Figure 2. The (Extended metasubstitution) rule is a syntactic rendering of the (Local parameterised composition) rule. The syntactic counterpart of the (Local character) rule is derivable and hence omitted.

We illustrate the expressive power of the system by giving two sample derivable rules.

(Axiom)

$$\frac{(\Theta \rhd \Gamma \vdash s \equiv t : \tau) \in E}{\Theta \rhd \Gamma \vdash s \equiv t : \tau}$$

(Equivalence)

$$\frac{\Theta \rhd \Gamma \vdash t : \tau}{\Theta \rhd \Gamma \vdash t \equiv t : \tau} \qquad \frac{\Theta \rhd \Gamma \vdash s \equiv t : \tau}{\Theta \rhd \Gamma \vdash t \equiv s : \tau} \qquad \frac{\Theta \rhd \Gamma \vdash s \equiv t : \tau \quad \Theta \rhd \Gamma \vdash t \equiv u : \tau}{\Theta \rhd \Gamma \vdash s \equiv u : \tau}$$

(Extended metasubstitution)

$$\frac{\begin{array}{c}\mathrm{M}_1 : [\vec{\sigma_1}]\tau_1, \ldots, \mathrm{M}_k : [\vec{\sigma_k}]\tau_k \rhd \Gamma \vdash s \equiv t : \tau \\ \Theta \rhd \Delta, \vec{x_i} : \vec{\sigma_i} \vdash s_i \equiv t_i : \tau_i \quad (1 \leq i \leq k)\end{array}}{\Theta \rhd \Gamma, \Delta \vdash s\{\mathrm{M}_i := (\vec{x_i})s_i\}_{1 \leq i \leq k} \equiv t\{\mathrm{M}_i := (\vec{x_i})t_i\}_{1 \leq i \leq k} : \tau}$$

Fig. 2. *Second-Order Equational Logic.*

(Substitution)

$$\frac{\Theta \rhd x_1 : \sigma_1, \ldots, x_n : \sigma_n \vdash s \equiv t : \tau \qquad \Theta \rhd \Gamma \vdash s_i \equiv t_i : \sigma_i \quad (1 \leq i \leq n)}{\Theta \rhd \Gamma \vdash s[{}^{s_i}/_{x_i}]_{1 \leq i \leq n} \equiv t[{}^{t_i}/_{x_i}]_{1 \leq i \leq n} : \tau}$$

(Extension)

$$\frac{\mathrm{M}_1 : [\vec{\sigma_1}]\tau_1, \ldots, \mathrm{M}_k : [\vec{\sigma_k}]\tau_k \rhd \Gamma \vdash s \equiv t : \tau}{\mathrm{M}_1 : [\vec{\sigma_1}, \vec{\sigma}]\tau_1, \ldots, \mathrm{M}_k : [\vec{\sigma_k}, \vec{\sigma}]\tau_k \rhd \Gamma, \vec{x} : \vec{\sigma} \vdash s^{\#} \equiv t^{\#} : \tau}$$

where $u^{\#} = u\{\mathrm{M}_i := (\vec{x_i})\mathrm{M}_i[\vec{x_i}, \vec{x}]\}_{1 \leq i \leq k}$

Parameterisation. Every term $\Theta \rhd \Gamma \vdash t : \tau$ can be *parameterised* to yield a term $\Theta, \widehat{\Gamma} \rhd \cdot \vdash \widehat{t} : \tau$ where, for $\Gamma = (x_1 : \tau_1, \ldots, x_n : \tau_n)$,

$$\widehat{\Gamma} = (\mathrm{X}_1 : [\,]\tau_1, \ldots, \mathrm{X}_n : [\,]\tau_n) \quad \text{and} \quad \widehat{t} = t[{}^{\mathrm{X}_1[\,]}/_{x_1}, \ldots, {}^{\mathrm{X}_n[\,]}/_{x_n}] \ .$$

Performing the operation on a set of equations E to obtain a set of parameterised equations $\widehat{E}$, we have that the following are equivalent:

$$\begin{array}{ll}\Theta \rhd \Gamma \vdash_E s \equiv t : \tau \ , & \Theta, \widehat{\Gamma} \rhd \cdot \vdash_E \widehat{s} \equiv \widehat{t} : \tau \ , \\ \Theta \rhd \Gamma \vdash_{\widehat{E}} s \equiv t : \tau \ , & \Theta, \widehat{\Gamma} \rhd \cdot \vdash_{\widehat{E}} \widehat{s} \equiv \widehat{t} : \tau \ .\end{array}$$

Thus, without loss of generality, one may restrict to axioms containing an empty variable context as in the *CRSs* of Klop [22]. However, there is no need for us to do so here.

6 Model Theory

The model theory of *Second-Order Equational Logic* is presented and exemplified. The soundness of deduction is a by-product of our methodology. In the next section, the model theory is used to establish a conservative-extension result.

Semantics. The interpretation of a term $\Theta \rhd \vec{x} : \vec{\sigma} \vdash t : \tau$ in a model A is that of its associated Kleisli map $\boldsymbol{y}(\vec{\sigma})_{@\tau} \longrightarrow \mathcal{M}(\underline{\Theta})$ $\big($see (5)$\big)$ according to the general definition (6). Explicitly, for $\Theta = (\mathrm{M}_1 : [\vec{\alpha_1}]\beta_1, \dots, \mathrm{M}_k : [\vec{\alpha_k}]\beta_k)$ and $\Gamma = (\vec{x} : \vec{\sigma})$, we have that the interpretation

$$[\![\Theta \rhd \Gamma \vdash t : \tau]\!]_A : [\![\Theta \rhd \Gamma]\!]_A \longrightarrow A_\tau \quad ,$$

where $[\![\Theta \rhd \Gamma]\!]_A = \prod_{1 \le i \le k} {A_{\beta_i}}^{\boldsymbol{y}(\vec{\alpha_i})} \times \boldsymbol{y}(\vec{\sigma})$, is given inductively on the structure of terms as follows.

- $[\![\Theta \rhd \Gamma \vdash x_j : \sigma_j]\!]_A$ is the composite $[\![\Theta \rhd \Gamma]\!]_A \xrightarrow{\pi_2} \boldsymbol{y}(\vec{\sigma}) \longrightarrow \boldsymbol{y}(\sigma_j) \longrightarrow A_{\sigma_j}$.
- $[\![\Theta \rhd \Gamma \vdash \mathrm{M}_i[t_1, \dots, t_{m_i}] : \beta_i]\!]_A$ is the composite

$$[\![\Theta \rhd \Gamma]\!]_A \xrightarrow{\langle \pi_i\, \pi_1, f \rangle} {A_{\beta_i}}^{\boldsymbol{y}(\vec{\alpha_i})} \times \prod_{1 \le j \le m_i} A_{\alpha_{i,j}} \xrightarrow{\varsigma} A_{\beta_i}$$

  where $f = \big\langle [\![\Theta \rhd \Gamma \vdash t_j : \alpha_{i,j}]\!]_A \big\rangle_{1 \le j \le m_i}$.
- For $\mathsf{o} : (\vec{\gamma_1})\tau_1, \dots, (\vec{\gamma_n})\tau_n \longrightarrow \tau$,

$$[\![\Theta \rhd \Gamma \vdash \mathsf{o}\big((\vec{y_1})t_1, \dots, (\vec{y_n})t_n\big) : \tau]\!]$$

  is the composite $[\![\Theta \rhd \Gamma]\!]_A \xrightarrow{\langle f_j \rangle_{1 \le j \le n}} \prod_{1 \le j \le n} {A_{\tau_j}}^{\boldsymbol{y}(\vec{\gamma_j})} \longrightarrow A_\tau$ where f_j is the exponential transpose of

$$\begin{aligned} &\textstyle\prod_{1 \le i \le k} {A_{\beta_i}}^{\boldsymbol{y}(\vec{\alpha_i})} \times \boldsymbol{y}(\vec{\sigma}) \times \boldsymbol{y}(\vec{\gamma_j}) \\ &\quad \cong \textstyle\prod_{1 \le i \le k} {A_{\beta_i}}^{\boldsymbol{y}(\vec{\alpha_i})} \times \boldsymbol{y}(\vec{\sigma}, \vec{\gamma_j}) \xrightarrow{[\![\Theta \rhd \Gamma, \vec{y_j} : \vec{\gamma_j} \vdash t_j : \tau_j]\!]_A} A_{\tau_j} \quad . \end{aligned}$$

Models. A model A *satisfies* $\Theta \rhd \Gamma \vdash s \equiv t : \tau$, written $A \models (\Theta \rhd \Gamma \vdash s \equiv t : \tau)$, iff $[\![\Theta \rhd \Gamma \vdash s : \tau]\!]_A = [\![\Theta \rhd \Gamma \vdash t : \tau]\!]_A$.

For an equational presentation E over a signature Σ, we write (Σ, E)-Mod for the full subcategory of Σ-Mod consisting of the Σ-models A that satisfy the axioms E. Thus, (Σ, E)-$\mathsf{Mod} \cong (\mathcal{M}, \underline{E})$-$\mathsf{Alg}$ where $\underline{E}$ is the set of parallel pairs of Kleisli maps corresponding to the pairs of terms in E.

Example 4. For the signature of the typed λ-calculus over a set of base types B (Example 1), a model

$${A_\tau}^{\boldsymbol{y}(\sigma)} \xrightarrow{\mathsf{abs}} A_{\sigma \Rightarrow \tau} \quad , \quad A_{\sigma \Rightarrow \tau} \times A_\sigma \xrightarrow{\mathsf{app}} A_\tau$$

$$\boldsymbol{y}(\tau) \xrightarrow{\nu} A_\tau \longleftarrow A_\tau \circledcirc A$$

satisfies the (β) and (η) axioms (Example 3) iff the following diagrams commute.

$$\begin{array}{ccc} {A_\tau}^{\boldsymbol{y}(\sigma)} \times A_\sigma & \xrightarrow{\varsigma} & A_\tau \\ \downarrow{\scriptstyle \mathsf{abs} \times \mathrm{id}} & (\beta) & \| \\ A_{\sigma \Rightarrow \tau} \times A_\sigma & \xrightarrow{\mathsf{app}} & A_\tau \end{array} \qquad \begin{array}{ccc} A_{\sigma \Rightarrow \tau} & \xrightarrow{\mathrm{id}} & A_{\sigma \Rightarrow \tau} \\ \downarrow{\scriptstyle \lambda(\mathsf{app}\,(\mathrm{id} \times \nu))} & (\eta) & \| \\ {A_\tau}^{\boldsymbol{y}(\sigma)} & \xrightarrow{\mathsf{abs}} & A_{\sigma \Rightarrow \tau} \end{array}$$

The Lawvere theory $\mathbb{L}$ associated to such a model is a $B^{\Rightarrow}$-sorted Lawvere theory equipped with cartesian closed structure $\mathbb{L}(\vec{\gamma} \cdot \sigma \,,\, \tau) \cong \mathbb{L}(\vec{\gamma} \,,\, \sigma \Rightarrow \tau)$.

On the other hand, for every cartesian closed category $\mathscr{C}$, an assignment $C : B \longrightarrow \mathscr{C}$ extends to an assignment $C^{\#} : B^{\Rightarrow} \longrightarrow \mathscr{C}$ $\big($with $C^{\#}(\beta) = C_\beta$ for $\beta \in B$, and $C^{\#}(\sigma \Rightarrow \tau) = C^{\#}(\sigma) \Rightarrow C^{\#}(\tau)\big)$ and canonically gives rise to a model on $\langle C^{\#}, C^{\#}\rangle$ that satisfies the (β) and (η) axioms.

Soundness. The soundness of *Second-Order Equational Logic* follows as a direct consequence of that of *Equational Metalogic*.

(Soundness) For an equational presentation E over a signature Σ, if the judgement $\Theta \rhd \Gamma \vdash s \equiv t : \tau$ is derivable from E then $A \models (\Theta \rhd \Gamma \vdash s \equiv t : \tau)$ for all (Σ, E)-models A.

7 Conservativity

Every first-order signature can be regarded as a second-order signature, and every first-order term $\Gamma \vdash t : \tau$ as the second-order term $\cdot \rhd \Gamma \vdash t : \tau$. It follows that, for a set of first-order equations,

$$\begin{array}{l} \text{if } \Gamma \vdash s \equiv t : \tau \text{ is derivable in (first-order) equational logic, then} \\ \cdot \rhd \Gamma \vdash s \equiv t : \tau \text{ is derivable in second-order equational logic.} \end{array} \tag{7}$$

We now proceed to establish the converse.

Let Ω be a first-order signature over a set of types T. For $\mathscr{C}$ cartesian and $C \in \mathscr{C}^T$, since $\langle C, - \rangle : \mathscr{C}^T \longrightarrow (\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]})^T$ preserves limits, it follows that an Ω-algebra structure

$$\textstyle\prod_{1 \leq i \leq n} C_{\tau_i} \longrightarrow C_\tau \qquad \big(\tau_1, \ldots, \tau_n \to \tau \text{ in } \Omega\big)$$

on C yields the Ω-algebra structure

$$\textstyle\prod_{1 \leq i \leq n} \langle\!\langle C, C_{\tau_i} \rangle\!\rangle \cong \big\langle\!\big\langle C, \prod_{1 \leq i \leq n} C_{\tau_i} \big\rangle\!\big\rangle \longrightarrow \langle\!\langle C, C_\tau \rangle\!\rangle \qquad \big(\tau_1, \ldots, \tau_n \to \tau \text{ in } \Omega\big)$$

on $\langle C, C \rangle$. This Ω-algebra structure is compatible with the canonical monoid structure, and we thus obtain an $\underline{\Omega}$-model on $\langle C, C \rangle$.

The interpretations of first-order terms are related as follows:

$$\begin{array}{ccc} \prod_{1 \leq i \leq n} \langle\!\langle C, C_{\tau_i} \rangle\!\rangle & \xrightarrow{\quad\cong\quad} & \big\langle\!\big\langle C, \prod_{1 \leq i \leq n} C_{\tau_i} \big\rangle\!\big\rangle \\ {\scriptstyle [\![\widehat{\Gamma} \rhd \cdot \vdash \widehat{t} : \tau]\!]_{\langle C, C \rangle}} \searrow & & \swarrow {\scriptstyle \langle\!\langle C, [\![\Gamma \vdash t : \tau]\!]_C \rangle\!\rangle} \\ & \langle\!\langle C, C_\tau \rangle\!\rangle & \end{array}$$

and we have that $C \models (\Gamma \vdash s \equiv t : \tau)$ iff $\langle C, C \rangle \models (\widehat{\Gamma} \rhd \cdot \vdash \widehat{s} \equiv \widehat{t} : \tau)$. Consequently, if $A \models (\widehat{\Gamma} \rhd \cdot \vdash \widehat{s} \equiv \widehat{t} : \tau)$ for all $(\underline{\Omega}, \widehat{E})$-models $A \in \big(\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]}\big)^T$ then $C \models (\Gamma \vdash s \equiv t : \tau)$ for all (Ω, E)-algebras $C \in \mathscr{C}^T$. Thus, the converse of (7) holds, and we have established the following result.

(Conservativity) *Second-Order Equational Logic* is a conservative extension of *(First-Order) Equational Logic.*

8 Completeness

We finally outline the semantic completeness of equational derivability and the derivability completeness of (bidirectional) *Second-Order Term Rewriting.*

$$\frac{\begin{array}{c}(\mathrm{M}_1 : [\vec{\sigma_1}]\tau_1, \ldots, \mathrm{M}_k : [\vec{\sigma_k}]\tau_k \rhd \vec{x} : \vec{\alpha} \vdash l \equiv r : \tau) \in E \\ \rho : \vec{\alpha} \longrightarrow \vec{\beta}\ ,\quad \Theta \rhd \vec{y} : \vec{\beta}, \vec{x_j} : \vec{\sigma_j} \vdash t_j : \tau_j \ \ (1 \le j \le k)\end{array}}{\Theta \rhd \vec{y} : \vec{\beta} \vdash l[{}^{y_{\rho i}}\!/x_i]_{1 \le i \le |\vec{\alpha}|}\{\mathrm{M}_j := (\vec{x_j})t_j\}_{1 \le j \le k} \approx r[{}^{y_{\rho i}}\!/x_i]_{1 \le i \le |\vec{\alpha}|}\{\mathrm{M}_j := (\vec{x_j})t_j\}_{1 \le j \le k} : \tau}$$

$$\frac{\Theta \rhd \vec{x} : \vec{\sigma} \vdash s \approx t : \tau \qquad \Theta \rhd \Gamma \vdash s_i \approx t_i : \sigma_i \ (1 \le i \le |\vec{\sigma}|)}{\Theta \rhd \Gamma \vdash s[{}^{s_i}\!/x_i]_{1 \le i \le |\vec{\sigma}|} \approx t[{}^{t_i}\!/x_i]_{1 \le i \le |\vec{\sigma}|} : \tau}$$

$$\frac{\Theta \rhd \Gamma, \vec{x_i} : \vec{\sigma_i} \vdash s_i \approx t_i : \tau_i \quad (1 \le i \le k)}{\Theta \rhd \Gamma \vdash \mathsf{o}\big((\vec{x_1})s_1, \ldots, (\vec{x_k})s_k\big) \approx \mathsf{o}\big((\vec{x_1})t_1, \ldots, (\vec{x_k})t_k\big) : \tau}\ \big(\mathsf{o} : (\vec{\sigma_1})\tau_1, \ldots, (\vec{\sigma_k})\tau_k \longrightarrow \tau\big)$$

Fig. 3. Rules of $\approx$ (omitting those of equivalence).

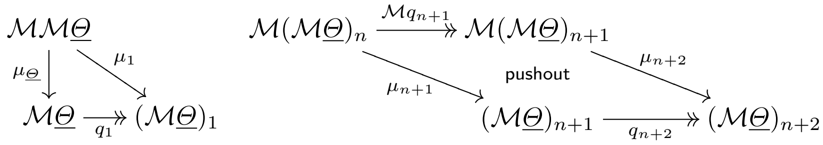
Free algebras. It follows from our theory of free constructions [12, 13] that the forgetful functor (Σ, E)-$\mathsf{Mod} \longrightarrow (\boldsymbol{\mathcal{S}et}^{\mathbb{F}[T]})^T$ is monadic. Hence, writing $\mathcal{M}_E$ for the induced monad, we have a quotient map $q : \mathcal{M} \longrightarrow \mathcal{M}_E$.

In fact, since $\mathcal{M}$ is finitary and preserves epimorphisms, we have the following construction of free algebras, see [12, 13].

(A) For $E = \{\Theta_i \rhd \Gamma_i \vdash l_i \equiv r_i : \tau_i\}_{i \in I}$ we take the joint coequaliser

$$\big(\llbracket \Theta_i \rhd \Gamma_i \rrbracket_{\mathcal{M}\underline{\Theta}}\big)_{@\tau_i} \underset{\llbracket \Theta_i \rhd \Gamma_i \vdash r_i : \tau_i \rrbracket_{@\tau_i}}{\overset{\llbracket \Theta_i \rhd \Gamma_i \vdash l_i : \tau_i \rrbracket_{@\tau_i}}{\rightrightarrows}} \mathcal{M}\underline{\Theta} \xrightarrow[\mathsf{coeq}]{q_1} (\mathcal{M}\underline{\Theta})_1 \qquad (i \in I)$$

(B) We perform the following inductive construction setting $(\mathcal{M}\underline{\Theta})_0 = \mathcal{M}\underline{\Theta}$.



(C) We obtain $\mathcal{M}_E\underline{\Theta}$ as the colimit of

$$\mathcal{M}\underline{\Theta} \overset{q_1}{\twoheadrightarrow} (\mathcal{M}\underline{\Theta})_1 \twoheadrightarrow \cdots \twoheadrightarrow (\mathcal{M}\underline{\Theta})_n \overset{q_n}{\twoheadrightarrow} \cdots$$

Hence we have an epimorphic quotient map

$$q_{\underline{\Theta}} : \mathcal{M}\underline{\Theta} \twoheadrightarrow \mathcal{M}_E\underline{\Theta}\ . \tag{8}$$

In the light of the (Internal completeness) result, our method for establishing completeness is to show that, for $\Theta \rhd \vec{x} : \vec{\sigma} \vdash s : \tau$ and $\Theta \rhd \vec{x} : \vec{\sigma} \vdash t : \tau$, if $q_{\underline{\Theta},\tau,\vec{\sigma}}(s) = q_{\underline{\Theta},\tau,\vec{\sigma}}(t)$ then $\Theta \rhd \vec{x} : \vec{\sigma} \vdash s \equiv t : \tau$ is derivable from E.

Syntactic model. A concrete analysis of the constructions (A–C) yields a characterisation of the quotient (8) as induced by the equivalence relation $\approx$ on $\mathcal{M}\underline{\Theta}$ given by the rules in Figure 3. Consequently, since derivability in this deductive system can be mimicked in *Second-Order Equational Logic*, we have:

$$\frac{\begin{array}{c}(\mathrm{M}_1 : [\vec{\sigma_1}]\tau_1, \ldots, \mathrm{M}_k : [\vec{\sigma_k}]\tau_k \rhd \vec{x} : \vec{\alpha} \vdash l \equiv r : \tau) \in E \\ \rho : \vec{\alpha} \longrightarrow \vec{\beta} \quad , \qquad \Theta \rhd \vec{y} : \vec{\beta}, \vec{x_j} : \vec{\sigma_j} \vdash t_j : \tau_j \quad (1 \leq j \leq k) \\ \Theta \rhd \Gamma \vdash s_\ell : \beta_\ell \quad (1 \leq \ell \leq |\vec{\beta}|)\end{array}}{\Theta \rhd \Gamma \vdash l' \to r' : \tau}$$

where $u' = u[{}^{y_{\rho i}}/_{x_i}]_{1 \leq i \leq |\vec{\alpha}|}\{\mathrm{M}_j := (\vec{x_j})t_j\}_{1 \leq j \leq k}[{}^{s_\ell}/_{y_\ell}]_{1 \leq \ell \leq |\vec{\beta}|}$

$$\frac{\Theta \rhd \Gamma \vdash s_i \to t_i : \sigma_i}{\Theta \rhd \Gamma \vdash \mathrm{M}[\ldots, s_i, \ldots] \to \mathrm{M}[\ldots, t_i, \ldots] : \tau} \big(1 \leq i \leq n \ , \ \ (\mathrm{M} : [\sigma_1, \ldots, \sigma_n]\tau) \in \Theta\big)$$

$$\frac{\Theta \rhd \Gamma, \vec{x_i} : \vec{\sigma_i} \vdash s_i \to t_i : \tau_i}{\Theta \rhd \Gamma \vdash \mathsf{o}\big(\ldots, (\vec{x_i})s_i, \ldots\big) \to \mathsf{o}\big(\ldots, (\vec{x_i})t_i, \ldots\big) : \tau} \left(\begin{array}{c}\mathsf{o} : (\vec{\sigma_1})\tau_1, \ldots, (\vec{\sigma_k})\tau_k \to \tau \ , \\ 1 \leq i \leq k\end{array}\right)$$

Fig. 4. *Second-Order Term Rewriting.*

(Completeness) For an equational presentation E over a signature Σ, if $A \models (\Theta \rhd \Gamma \vdash s \equiv t : \tau)$ for all (Σ, E)-models A then $\Theta \rhd \Gamma \vdash s \equiv t : \tau$ is derivable from E.

Moreover, since $\approx$ can be characterised as the equivalence relation generated by the *Second-Order Term Rewriting* relation $\to$ given in Figure 4, we also have:

(Completeness of *Second-Order Term Rewriting*) For every equational presentation, $\Theta \rhd \Gamma \vdash s \equiv t : \tau$ iff $\Theta \rhd \Gamma \vdash s \overset{*}{\longleftrightarrow} t : \tau$.

9 Conclusion

We have introduced *Second-Order Equational Logic*: a logical framework for specifying and reasoning about simply-typed equational theories over algebraic signatures with variable-binding operators. The conceptual part of our development consisted in the synthesis of the equational deductive system from a canonical algebraic model theory that forms the basis of *Second-Order Universal Algebra*; the technical part established the soundness and completeness of the logic. We have also provided logical and semantic foundations for higher-order term rewriting, specifically *CRSs*, by exhibiting *Second-Order Term Rewriting* as a sound and complete computational method for establishing equality.

Acknowledgement. We are most grateful to Pierre-Louis Curien for detailed comments on a preliminary version of the paper.

References

[1] P. Aczel. A general Church-Rosser theorem. Typescript, 1978.
[2] P. Aczel. Frege structures and the notion of proposition, truth and set. In *The Kleene Symposium*, pages 31–59, 1980.
[3] F. Baader and T. Nipkow. *Term Rewriting and All That.* CUP, 1998.

[4] G. Birkhoff. On the structure of abstract algebras. *P. Camb. Philos. Soc.*, 31:433–454, 1935.
[5] A. Church. A formulation of the simple theory of types. *J. Symbolic Logic*, 5:56–68, 1940.
[6] R. Clouston and A. Pitts. Nominal equational logic. *ENTCS*, 172:223–257, 2007.
[7] M. Fiore. Algebraic meta-theories and synthesis of equational logics. Research Programme, 2009.
[8] M. Fiore. Semantic analysis of normalisation by evaluation for typed lambda calculus. In *PPDP 2002*, pages 26–37, 2002.
[9] M. Fiore. Mathematical models of computational and combinatorial structures. In *FOSSACS 2005*, volume 3441 of *LNCS*, pages 25–46, 2005.
[10] M. Fiore. A mathematical theory of substitution and its applications to syntax and semantics. Invited tutorial for the *Workshop on Mathematical Theories of Abstraction, Substitution and Naming in Computer Science*, ICMS, 2007.
[11] M. Fiore. Second-order and dependently-sorted abstract syntax. In *LICS'08*, pages 57–68, 2008.
[12] M. Fiore and C.-K. Hur. On the construction of free algebras for equational systems. *Theor. Comput. Sci.*, 410:1704–1729, 2008.
[13] M. Fiore and C.-K. Hur. Term equational systems and logics. In *MFPS XXIV*, volume 218 of *ENTCS*, pages 171–192, 2008.
[14] M. Fiore and O. Mahmoud. Second-order algebraic theories. To appear in *MFCS 2010*, 2010.
[15] M. Fiore, G. Plotkin, and D. Turi. Abstract syntax and variable binding. In *LICS'99*, pages 193–202, 1999.
[16] J. Goguen and J. Meseguer. Completeness of many-sorted equational logic. *Houston J. Math.*, 11:307–334, 1985.
[17] M. Hamana. Term rewriting with variable binding: An initial algebra approach. In *PPDP 2003*, pages 148–159, 2003.
[18] M. Hamana. Free Σ-monoids: A higher-order syntax with metavariables. In *APLAS 2004*, volume 3202 of *LNCS*, pages 348–363, 2005.
[19] C.-K. Hur. *Categorical Equational Systems: Algebraic Models and Equational Reasoning*. PhD thesis, Computer Laboratory, University of Cambridge, 2010.
[20] G. Janelidze and G. Kelly. A note on actions of a monoidal category. *TAC*, 9(4):61–91, 2001.
[21] J. Klop. *Combinatory Reduction Systems*. PhD thesis, Mathematical Centre Tracts 127, CWI, Amsterdam, 1980.
[22] J. Klop, V. van Oostrom, and F. van Raamsdonk. Combinatory reduction systems: introduction and survey. *Theor. Comput. Sci.*, 121:279–308, 1993.
[23] F. Lawvere. Functorial semantics of algebraic theories. Republished in: Reprints in *TAC*, No. 5, pp. 1–121, 2004.
[24] M.J. Gabbay and A. Mathijssen. Nominal (universal) algebra: Equational logic with names and binding. *J. Logic Computation*, 19(6):1455–1508, 2009.
[25] D. Pigozzi and A. Salibra. The abstract variable-binding calculus. *Studia Logica*, 55:129–179, 1995.
[26] G. Plotkin. Binding algebras: A step from universal algebra to type theory. Invited talk at *RTA-98*, 1998.
[27] Y. Sun. *A Framework for Binding Operators*. PhD thesis, LFCS, The University of Edinburgh, 1992.
[28] Y. Sun. An algebraic generalization of Frege structures — binding algebras. *Theor. Comput. Sci.*, 211:189–232, 1999.
[29] F. van Raamsdonk. Higher-order rewriting. In *Term Rewriting Systems*, volume 55 of *Cambridge Tracts in Theoretical Computer Science*, pages 588–667, 2003.