Substitution Structures

Andrew Bacon*

January 10, 2019

Contents

1 Background **6**
1.1 Background on type theory . . . 6
1.2 Background on monoids . . . 8

2 Substitution Structures **9**
2.1 Substitution structures . . . 9
2.2 Faithfulness . . . 12
2.3 Examples of substitution structures . . . 13
2.4 Congruences . . . 16
2.5 M-logical relations . . . 17

3 Metaphysical Notions **18**
3.1 Pure elements . . . 18
3.2 Metaphysical definability . . . 21
3.3 Free objects . . . 23
3.4 Fundamentality and cofundamentality . . . 24
3.5 Fundamental languages . . . 27

4 M-Set Structures **30**
4.1 M-sets and function spaces . . . 31
4.2 The category of M-sets . . . 34
4.3 Decomposition principles . . . 36
4.4 The term model as a full M-set structure . . . 37

5 Metaphysical Theorizing **39**
5.1 Expressing substitution-theoretic concepts in the object language . . . 39
5.2 Object language principles . . . 43
5.3 A Boolean model . . . 45
5.4 Further model constructions . . . 47

6 Concluding Remarks **51**

*I owe thanks to Mike Caie, Cian Dorr, Peter Fritz, Jeremy Goodman and Harvey Lederman for many helpful conversations on this material, and two anonymous referees for their suggestions on improving the presentation of the paper. I'd like especially to thank referee 1 who caught a great number of errors and points of unclarity, and for directing me to the related work of Aczel [1].

An increasing amount of twenty-first century metaphysics is couched in explicitly hyperintensional terms: concepts such as grounding, fundamentality and metaphysical priority can draw distinctions between necessarily equivalent propositions and properties.[1] While hyperintensionality in the philosophy of language is often taken to be merely a feature of our representations of the world, a prerequisite of hyperintensional metaphysics is that reality itself be hyperintensional. At the metaphysical level, propositions, properties, operators, and other elements of the type hierarchy, must be at least more fine-grained than functions from possible worlds to extensions. In this paper I develop, in the setting of type theory, a general framework for reasoning about the granularity of propositions and properties.

Probably the two most influential accounts of propositional granularity are the *possible worlds* account, and the *structured* account. According to the possible worlds account, propositions are identified with sets of possible worlds. There are many equivalent ways to extend this position on granularity to other entities in the type hierarchy. Properties, for example, may be identified with functions from individuals to propositions, operators with functions from propositions to propositions, and so on.[2] The possible worlds theory is an instance of a more general class of theories that subscribe to *Booleanism*: the thesis that propositions satisfy the identities of a Boolean algebra.

According to the structured theory, propositions are structured entities that are built up out of simple individuals, properties, operators, etc. in a way that parallels the way that a sentence is built out of individual constants, predicates, operator expressions, and so on. The structured picture similarly extends to other entities in the type hierarchy. Note that unlike the possible worlds theory, however, we cannot simply identify properties, operators, and so on, with arbitrary functions from source to target types. An operator, for example, is a structured entity in its own right, and always increases the constituent count of the thing it is applied to (there is thus no operator corresponding to the identity function on propositions, for example).

There are several obstacles to giving a straightforward account of the structured vision. Unlike the possible worlds theory, which has an elegant and now well-understood model theory, there is no 'standard' model of the structured theory.[3] One sticky issue concerns the proper way to treat variables; an account is needed if we admit structured propositions containing quantifiers, λ-abstractions and other devices that bind variables.[4] But even if a solution to this problem is forthcoming, as an account of reality as opposed to representation, it is far from obvious that syntactic things like variables should appear in the theory at all, and it is perhaps sign that we are individuating too finely.

Another hard challenge for the structured theory, and the primary cause for departure here, is that flat-footed formulations of it are not consistent in standard systems of higher-order logic due to the infamous Russell-Myhill paradox.[5]

There are intermediate positions between the possible worlds theory and the structured

[1] See Nolan [25] for a helpful overview of this trend.

[2] Equivalently, one can also treat properties as functions from worlds to sets of individuals, or as sets of world individuals pairs. The choice of representation won't matter much in what follows, but treating them as functions from individuals has a natural interpretation in functional type theory, where the type $\sigma \to \tau$ gets interpreted by a set of functions from type σ things to type τ things.

[3] Although see Lewis [22], Cresswell [8] and Soames [29] for some examples of structured theories of propositions.

[4] And if we allow variables and λ-abstractions we must be careful: the identify function, which we just saw cannot be structural, may be defined by the λ-term $\lambda p\, p$.

[5] See Hodes [18], Dorr [9] for recent discussion of this problem. Relatedly, even outside the context of higher-order logic, there are issues surrounding the iteration of operators, such as belief, that are sensitive to structure (see, for example, Cresswell [7], Cresswell [8] chapter 10, Lapierre [20]).

theory. We have mentioned one already: one might accept the Boolean identities without identifying metaphysically necessarily equivalent propositions.[6] There are also intermediate positions that reject Booleanism, but do not go as far as the full-blown structured view. This sort of approach is characteristic of Dorr [9], and a lot of Kit Fine's recent work ([12], [13]).[7]

If we imagine the structured theory as occupying the topmost position in this hierarchy of hypotheses about propositional granularity, one way to generate intermediate positions is to take notions that are well-defined according to the structured picture — such as constituency, simplicity and definability — and theorize in terms of those notions alone, without presupposing the structured theory. The approach of this paper is to begin by noting that, were a structured picture correct, there would be a well-defined notion of *substitution*: an operation that replaces all the simple constituents of a structured proposition with (possibly) complex constituents that have the same type as the constituent they are replacing (individuals for individuals, properties for properties, and so on).[8] The notion of substitution, so understood, is subject to a number of natural laws: for example, if you make a substitution on a structured property F, and on a structured argument a, and apply the first result to the second, that should be the same as making that substitution on Fa.

Instead of assuming the structured picture, we could instead take this notion of substitution as primitive, subject to constraints like this one, and from that explain other notions like constituency, simplicity and definability in terms of it. I aim, here, to demonstrate that we can understand a large class of intermediate theories of granularity in substitutional terms. A striking advantage of this methodology is that it is possible to take definitions of structural notions in terms of substitutions, and apply them in contexts where the structured picture is not being assumed. Many structural notions, astonishingly, can even be introduced in a Boolean setting.[9] We consider some Boolean structures of interest in section 5.3.

The 'substitution-first' approach to propositional granularity also allows for the formulation of a number of new concepts that are not straightforwardly available to the structured theorist, but should be of interest to intermediate theories of granularity (see, for example, the notions of purity, constructability, quasi-functionality and cofundamentality below).

Here is an overview of the concepts explored in the paper.

> *Substitution structure.* After some background on type theory (section 1.1) and monoids (section 1.2) we introduce fundamental concept of this paper: a substitution structure (section 2.1). This is roughly a model of type theory that is equipped with a notion of substitution on the elements at each type in the structure. Some elementary examples of substitution structures are discussed in section 2.3.

[6] There might only be *two* propositions, the true and the false, as Frege arguably held. Or propositions might be more fine-grained than sets of worlds.

[7] Another example of an intermediate theory is Yablo [32], which is designed with non-metaphysical applications in mind as well.

[8] Throughout this paper I shall be regimenting questions informally formulated in terms of propositions, properties and relations in terms of quantification into sentence, predicate and relation position. *Pace* Quine, we shall understand the latter sort of higher-order quantification in its own terms, without assuming that it be reduced to singular quantification over propositions, properties and relations. For a defense of this attitude towards higher-order logic see Prior [27] , Rayo and Yablo [28] and Williamson [31]. Any higher-order theory may be 'converted' into a first-order theory of properties and relations with a primitive application function, subject to type. Thus first-order versions of the theories discussed here are available to the Quinean, but we shall not make any systematic attempt to accommodate this sort of skeptic about higher-order quantification.

[9] Although, unsurprisingly, these notions will in general behave differently in a Boolean context.

Purity. According to a hardline structural picture, everything is made up of simple constituents which may always be substituted for other things, parallel to the way that everything is made up of constants in a language. Among other things this precludes the existence of combinators: things made up entirely of variables and λs, which don't involve constants and do appear to express things with constituents. Nothing can satisfy the laws of the identity combinator ($(\lambda x\, x)a = a$ for all a), for example, since every structural operation must increase constituent count. A *pure* element is an element that is left alone by every substitution. Pure elements intuitively correspond to elements that have no constituents, and the existence of pure elements allows for there to be combinators, and other interesting devices not available to the structured theorist. This concept is examined in section 3.1.

Fundamentality. According to the structured picture the simple, or *fundamental*, elements are uniquely distinguished by the fact that they *freely generate* the elements in each type, similar to the way that a language is freely generated by its constants. This means that for every way of matching each fundamental element to a (possibly) complex element of the same type, there is a unique substitution that takes each fundamental element to its match. This gives us a purely substitution-theoretic account of fundamentality that is parallel to the fact that languages are freely generated from their constants. We investigate this in section 3.3 and 3.4.

Cofundamentality. According to the above analysis of fundamentality, it turns out that if a relation, e.g. *having less mass than*, is fundamental, then its converse, in this case *having more mass than*, cannot also be fundamental. It is somewhat surprising that reality makes choices like these; choices that seem to break obvious symmetries. Another approach is to theorize instead with a relational notion of *cofundamentality*, in which a single collection of elements can be cofundamental with *having less mass than*, and with *having more mass than*, but not both at once. The substitutional analysis gives us a very natural model of this notion. In the case of languages and models of structured propositions there is always a unique set which generates the whole structure. By contrast, in many intermediate theories of granularity, there will typically be multiple sets that generate the structure — in this case we say it has multiple *fundamental bases*, by analogy with the notion of a basis from linear algebra. In the proposed model, some things are cofundamental iff there is a fundamental basis that contains them. This is developed in section 3.4.

Metaphysical Definability. On the structural/linguistic picture we identify fundamental things with the simple elements/constants, and the non-fundamental things with the things we can define out of them. In the linguistic case, if an expression α can be defined from the constants $c_1...c_n$, than α will be moved by some substitution that replaces one or more of $c_1...c_n$. Conversely, if α can be defined from $c_1...c_n$ then α will be left alone by any substitution that fixes each of $c_1...c_n$. This leads to a definition of definability that is purely substitution theoretic, and allows for a neutral characterisation of definability that can be applied to many intermediate theories. This idea is explored in section 3.2.[10]

Constructability. Say that an element a is constructable from some other elements, $c_1...c_n$, iff there is some pure element q such that $q(c_1)...(c_n) = a$. For structural views,

[10]The terminology used here is repurposed from [ANON] [REF], where it is used in a more specific context.

in which there are no pure elements, constructability is not a interesting concept, but we shall consider an interesting class of structures (generated full M-set structures) in which constructability and metaphysical definability line up. This connection is investigated in section 4.3.

Quasi-functionality. Functionality implies that for any two properties, F and G, if $Fa = Ga$ for every a, then $F = G$. This imposes non-trivial constraints on the granularity of properties: distinct properties cannot agree about what they do to each individual. In this paper, a weakening of functionality, *quasi-functionality*, is explored (see section 2.1). Quasi-functionality only requires that F and G be equal if the result of applying iF to a and the result of applying iG to a is the same, for every a and substitution i. This turns out to identify an interesting class of structures which are examined in section 4.1.

Fundamental languages. Intuitively a fundamental language is one in which every fundamental thing is denoted by some unique non-logical constant of the language. In this paper we propose a precise definition of this notion: we require that there be a homomorphism from the substitutions on the language and the substitutions on reality, and a homomorphism from the language to reality (an interpretation) that commute in a certain way. Being a structure that has a fundamental language is a non-trivial constraint, and implies that reality behave a bit like a language in certain respects. These ideas are explored in section 3.5.

This is covered in sections 1 to 3. These sections also cover some basic constructions on substitution structures: congruences, quotients and M-logical relations. In section 4, a concrete class of quasi-functional substitution structures are introduced: M-set structures. Many facts about these structures are explored and a representation theorem showing that every quasi-functional substitution structure is isomorphic to an M-set structure is proved. In section 5 we apply the model theoretic ideas developed in the earlier parts of the paper to theories couched in a higher-order logic, and outline some object language principles of interest.[11] Many of the ideas in this paper receive a particularly simple treatment within the framework of category theory. Where relevant I draw these connections, but the reader unfamiliar with category theory may skip these remarks without much loss.

For those wishing to follow up on the concepts in this paper, basic background on type theory and applicative structures can be found in Mitchell [23] (a very compressed overview of the notions needed in this paper are found in section 1.1). For background on monoids and actions the reader should consult Kil'p et al. [19], and for more specific information on the category of M-sets, Ebrahimi and Mahmoudi [10]. Mitchell [23] also contains some information on presheaf catogories, and their interpretation of type-theory. In what follows we present these concepts from first-principles.[12]

[11] This area is still very much in its infancy, although see the companion paper [ANON] [REF] for the development of a particular theory of this sort that draws heavily on the machinery developed here.

[12] Readers interested in this project might also wish to look at Aczel [1] and subsequent literature. Although there are some important differences in the approaches, the starting point is the same: Aczel develops a theory of (non-wellfounded) structured entities by taking the idea of a replacement of constituents as primitive. A full comparison of these projects would take us too far afield here.

1 Background

In this section we introduce some definitions and facts about monoids and type theory that we shall rely on in the rest of the paper.

1.1 Background on type theory

The type heirarchy is freely generated from two basic types, e and t, and a binary type constructor $\to$. e is the type of *individuals*, and t, the type of *truth evaluable* entities (propositions). If σ and τ are types then $(\sigma \to \tau)$ is also a type: it is a *functional type* whose members have arguments of type σ and values of type τ.

We write $\mathcal{L}(\Sigma)$ to denote the typed language over signature Σ. The signature determines a type indexed collection of (possibly empty) sets, Σ^σ, containing the logical and non-logical constants of type σ. For each σ write Var^σ to denote an infinite set of variables of type σ, and write Var for the set of all variables belonging to any Var^σ. The terms of this language are defined recursively as follows:

- If $\alpha \in \Sigma^\sigma \cup Var^\sigma$ then α is a term of type σ.
- If α is a term of type $\sigma \to \tau$ and β a term of type σ then $(\alpha\beta)$ is a term of type τ
- If α is a term of type τ and $x \in Var^\sigma$ $\lambda x\, \alpha$ is a term of type $\sigma \to \tau$

The notion of a free variable is defined in the usual way. We shall occasionally encounter product types: if σ and τ are types, $\sigma \times \tau$ is the product of types σ and τ. In this case we additionally have the term formation rules:

- If α has type σ and β has type τ then (α, β) is a term of type $\sigma \times \tau$
- If α is a term of type $\sigma \times \tau$ then $\pi_1\alpha$ is a term of type σ and $\pi_2\alpha$ a term of type τ.

In general we shall write $\mathcal{L}^\sigma(\Sigma)$ to denote the set of all closed terms of type σ in $\mathcal{L}(\Sigma)$ and, when no ambiguity arises, $\mathcal{L}(\Sigma)$ for the set of all closed terms.

Throughout this paper we shall work with a very general class of structures for interpreting type theory:

Definition 1 (Applicative structure.)**.** *An applicative structure A consists of a type-indexed collection of domains, A^σ, and application functions, $App^{\sigma,\tau}$ such that:*

- $App^{\sigma\tau} : (A^{\sigma\to\tau} \times A^\sigma) \to A^\tau$

Any element $f \in A^{\sigma\to\tau}$ therefore determines a function $a \mapsto App^{\sigma\tau}(f, a) : A^\sigma \to A^\tau$.[13] *In general several distinct elements in $A^{\sigma\to\tau}$ may determine the same functional behaviour, and there may be functions $A^\sigma \to A^\tau$ that are not determined by any element of $A^{\sigma\to\tau}$.*

Let us write $Curry(App^{\sigma\tau})$ for the map $f \mapsto (a \mapsto App^{\sigma\tau}(f, a)) : A^{\sigma\to\tau} \to (A^\sigma \to A^\tau)$. An applicative structure is:

- Functional *iff $Curry(App^{\sigma\tau})$ is injective for every σ and τ*
- Full *iff $Curry(App^{\sigma\tau})$ is surjective for every σ and τ*

[13] Throughout we use $a \mapsto \ldots a \ldots$ to denote the function that maps a to $\ldots a \ldots$ (it is thus the metalanguage analogue of the λ abstractor, except it denotes a set-theoretic function).

- *A functional applicative structure* has combinators *iff for every type* σ, τ *and* υ *there are elements* $k \in A^{\sigma\to\tau\to\sigma}$ *and* $s \in A^{(\sigma\to\tau\to\upsilon)\to(\sigma\to\tau)\to\sigma\to\upsilon}$ *such that:*

  - $kxy = x$ *whenever* $x \in A^{\sigma}$ *and* $y \in A^{\tau}$
  - $sxyz = xz(yz)$ *whenever* $x \in A^{\sigma\to\tau\to\upsilon}$, $y \in A^{\sigma\to\tau}$ *and* $z \in A^{\sigma}$

  Note that if the applicative structure is not functional, there may be multiple elements s *and* k *satisfying these equations.*

If $f \in A^{\sigma\to\tau}$ and $a \in A^{\sigma}$ we shall often write fa as shorthand for $App^{\sigma\tau}(f, a)$. We moreover will always associate brackets on terms to the left, so that fab is short for $(fa)b$, and associate brackets on types to the right, so that $\sigma \to \rho \to \tau$ is short for $(\sigma \to (\rho \to \tau))$. These conventions should be adopted when reading the conditions for the s and k combinators above. It should be noted that a full applicative structure always has combinators. An applicative structure is functional iff, for any $f, g \in A^{\sigma\to\tau}$, $f = g$ whenever $fa = ga$ for all $a \in A^{\sigma}$. Thus elements of $A^{\sigma\to\tau}$ can, for all intents and purposes, be identified with functions from A^{σ} to A^{τ}, as they are completely determined by their applicative behavior. Indeed, every functional applicative structure is isomorphic to an applicative structure where $A^{\sigma\to\tau}$ consists of a set of functions from A^{σ} to A^{τ}, and $App^{\sigma\tau}$ is interpreted as function application.

An interpretation function $[\![\cdot]\!]$ for a language $\mathcal{L}(\Sigma)$ in a functional applicative structure A is a type indexed collection of functions $[\![\cdot]\!]^{\sigma} : \Sigma^{\sigma} \to A^{\sigma}$, assigning elements of the appropriate type to each term. We omit the type superscript when no confusion arises. A variable assignment, g, is similarly a type indexed collection of mappings from Var^{σ} to A^{σ}. If A is functional and has combinators then every term α and variable assignment g determine an interpretation, $[\![\alpha]\!]^{g} \in A^{\sigma}$:

- $[\![c]\!]^{g} = [\![c]\!]$ for $c \in \Sigma$
- $[\![x]\!]^{g} = g(x)$ for $x \in Var$.
- $[\![\alpha\beta]\!]^{g} = App^{\sigma\tau}([\![\alpha]\!]^{g}, [\![\beta]\!]^{g})$
- $[\![\lambda x\alpha]\!]^{g}$ is the unique element $f \in A^{\sigma\to\tau}$ such that $App^{\sigma\tau}(f, a) = [\![\alpha]\!]^{g[a/x]}$ for any $a \in A^{\sigma}$.

Here $g[a/x]$ is the function that like g except it maps x to a. The uniqueness of f in the last condition is guaranteed by functionality, and the existence by the fact that A has combinators (see, for example, Mitchell [23]).

A combinator expression is a closed expression that involve no constants (only variables and λs). An example would be the C combinator: $C := \lambda R\lambda x\lambda y\, Ryx$, where R has type $(\sigma \to \tau \to \rho)$ (e.g. a relation of type $e \to e \to t$) and x and y have type σ and τ respectively. Intuitively, C maps a relation to its converse. Two more important examples of combinator expressions are $K^{\sigma\tau} := \lambda xy\, y$ where $x : \sigma$ and $y : \tau$, and $S^{\sigma\tau\rho} := \lambda xyz\, xz(yz)$ where $x : \sigma \to \tau \to \rho$, $y : \sigma \to \tau$ and $z : \sigma$. In a functional applicative structure with combinators these expressions are guaranteed to denote the (unique) elements s and k satisfying the two equations from definition 1. It will also be convenient to have a word for elements of a functional applicative structure, such as s and k, that are denoted by combinator expressions. In a harmless abuse of notation, we shall often also refer to combinator expressions as combinators.

Terms of the form $(\lambda x\, \alpha)\beta$ and $\alpha[\beta/x]$ where β contains no free variables that would get bound upon substitution, are called *simple* β-equivalents. When x is not free in ϕ, $\lambda x\, \phi x$ and ϕ are simple η-equivalents. Two terms are $\eta\beta$-equivalent if one can be obtained from the other by any number of substitutions of simple η-equivalent or simple β-equivalent terms.[14] The theory of $\eta\beta$ equivalence on a language $\mathcal{L}(\Sigma)$ is the equational theory consisting of all identities $\phi = \psi$ consisting of $\eta\beta$ equivalent terms.

1.2 Background on monoids

Definition 2 (Monoid). *A monoid* $(M, \circ, 1)$ *is a set* M*, an element* $1 \in M$ *and a binary operation* $\circ : M \times M \to M$ *such that:*

1. $1 \circ i = i = i \circ 1$ *for all* $i \in M$
2. $(i \circ j) \circ k = i \circ (j \circ k)$ *for all* $i, j, k \in M$

We shall adopt of the convention of using the letter M interchangeably to denote a monoid or the underlying set of that monoid.

Definition 3 (M-set). *Given a monoid* M*, an* M*-set is a pair* (A, μ)*, where* A *is a set and* $\mu : M \times A \to A$ *a function such that:*

1. $\mu(1, a) = a$ *for all* $a \in A$
2. $\mu(i, \mu(j, a)) = \mu(i \circ j, a)$ *for all* $i, j \in M$ *and* $a \in A$*.*

We shall often follow a convention of simply writing ia or $i(a)$ instead of $\mu(i, a)$, and ij instead of $i \circ j$. Note that ija disambiguates to $(i \circ j)a$ and $i(j(a))$, but due to the second clause of definition 3 it does not matter. Similarly, we shall often adopt the convention of omitting reference to μ.

In what follows we shall call the elements of M *substitutions*. If M acts on a set A and $a \in A$, we shall call *ia the result of applying the substitution i to a*. With this gloss in mind, it's easy to see why the above definitions are motivated. Substitutions can be composed, in the sense that you can perform one after the other, and composition is associative and has a unit (the substitution of everything for itself). If you apply the composition of two substitutions, $i \circ j$ to some a, that's the same as performing the substitution j to a, and then i to the result; whence the definition of an action.

Definition 4. *Given two* M*-sets,* (A, μ) *and* (B, ν) *a function* $f : A \to B$ *is called* equivariant *iff for any* $i \in M, a \in A$*,* $f(\mu(i, a)) = \nu(i, f(a))$*.*

We summarize all this with a familiar linguistic example.

Example 1 (Substitution monoids). *Let* $\mathcal{L}(\Sigma)$ *be a typed language in signature* Σ*. A substitution on* $\mathcal{L}(\Sigma)$ *can be identified with a collection of functions,* $i^\sigma : \Sigma^\sigma \to \mathcal{L}^\sigma(\Sigma)$ *taking each constant of type* σ *to an arbitrary closed term of type* σ*. A substitution may be extended to a function* $i^+ : \mathcal{L}(\Sigma) \to \mathcal{L}(\Sigma)$*, which can be defined recursively by setting* $i^+c = ic$ *for each constant* c*,* $i^+(\alpha\beta) = (i^+\alpha)(i^+\beta)$*, and* $i^+(\lambda x\alpha) = \lambda x\, i^+(\alpha)$*. A substitution can be*

[14] Note two terms that differ only in the labeling of bound variables (known as α-equivalent terms) are $\eta\beta$ equivalent. One shows this by induction on terms complexity; the key observation is that if x and y are distinct variables with y not appearing in α then $\lambda x\alpha$ is a simple η-equivalent of $\lambda y\, (\lambda x\alpha)y$ which is a simple β-equivalent of $\lambda y\, \alpha[y/x]$.

equivalently defined as a type-preserving function from $\mathcal{L}(\Sigma) \to \mathcal{L}(\Sigma)$ that commutes with application and lambda abstraction: every such function will be of the form i^+ for some $i : \Sigma \to \mathcal{L}(\Sigma)$.

The set of all substitutions, understood as functions from $\mathcal{L}(\Sigma)$ to $\mathcal{L}(\Sigma)$, form a monoid M under function composition. $\mathcal{L}(\Sigma)$ is an M-set, whose action $\mu(i, \alpha)$ is given by applying the substitution i to α. Restricted classes of substitutions also form monoids, and act on the language in a similar way:

- *We might identify a subset $\Lambda \subseteq \Sigma$ of* logical constants*. The set of all substitutions that fix logical constants forms a monoid.*
- *Above we have defined substitutions that only take closed terms as values. There is also a wider class of open substitutions that take constants to arbitrary terms, although we will not have much use for them. (Note that our notion of substitution moves constants but leaves variables alone.)*
- *The set of substitutions that fix all but finitely many members of Σ is a monoid.*
- *Assuming that $|\Sigma^\sigma| = |\mathcal{L}^\sigma(\Sigma)|$ whenever $\Sigma^\sigma \neq \emptyset$, then there is at least one substitution that is surjective on each type. In which case, the set of surjective substitutions forms a monoid.*

2 Substitution Structures

In this section the notion of a substitution structure is defined. A number of examples are discussed, and several important concepts — faithfulness, congruences and M-logical relations — are introduced.

2.1 Substitution structures

Definition 5 (Substitution structure)**.** *A substitution structure $(M, A, Sub^{\cdot})$ consists of an applicative structure, $A = (A^\sigma, App^{\sigma\tau})$, a monoid $M = (|M|, \circ)$, and a typed collection of actions $Sub^\sigma : M \times A^\sigma \to A^\sigma$ of the monoid on the domain A^σ, subject to the following constraints.*

1. $Sub^\sigma(1, a) = a$
2. $Sub^\sigma(i, Sub^\sigma(j, a)) = Sub^\sigma(i \circ j, a)$
3. $Sub^\tau(i, App^{\sigma\tau}(f, a)) = App^{\sigma\tau}(Sub^{\sigma\to\tau}(i, f), Sub^\sigma(i, a))$

A substitution structure $(M, A, Sub^{\cdot})$ is functional *iff the applicative structure A is functional.*

The first two conditions are simply the conditions for being an action. Thus (A^σ, Sub^σ) is an M-set for each σ. The final condition states that the action distributes over application.

Shorthand: when it is not ambiguous we shall shorten $App^{\sigma\tau}(f, a)$ to fa, and shorten $Sub^\sigma(i, a)$ to ia. More generally, following our conventions concerning M-sets, we suppress reference to the action Sub (provided no ambiguity arises), writing e.g. (M, A) to refer to the substitution structure (M, A, Sub).

Given these shorthands, clause (3), for example, becomes the much more reader friendly: $i(fa) = (if)(ia)$. The clauses in this definition can be straightforwardly motivated by reflecting on the linguistic case (example 1). For instance, if you perform some substitution on a term $\alpha\beta$ then that is the same as performing that substitution to α and β separately, and applying the result of the first substitution to the second. This gives us clause 3.

Notice that 2 and 3 can be compressed into a single axiom, 4:

4. $Sub^{\tau}(i, App^{\sigma\tau}(Sub^{\sigma\to\tau}(j, f), a)) = App^{\sigma\tau}(Sub^{\sigma\to\tau}(i \circ j, f), Sub^{\sigma}(i, a))$

Or, in our slightly more readable notation: $i((jf)a) = ((i \circ j)f)(ia)$.

Firstly, we note that in certain substitution structures, the behaviour of the substitutions is completely determined by their behaviour on the base types. Recall that a group is a monoid in which every element has an inverse: for every i, there is a j such that $i \circ j = 1$. This inverse is always unique and we write it i^{-1}.[15]

Definition 6 (Conjugation structures)**.** *Suppose that M is a group, A a functional applicative structure, and that M acts on A^t and A^e, with actions Sub^e and Sub^t respectively. If A is rich enough, one can form a substitution structure from A and M by inductively defining $Sub^{\sigma\to\tau}$ as:*

$$Sub^{\sigma\to\tau}(i, f) = a \mapsto Sub^{\tau}(i, App(f, Sub^{\sigma}(i^{-1}, a))) \text{ for each } f \in A^{\sigma\to\tau}, i \in M$$

Or, applying notational conventions: $i_{\sigma\to\tau}f = i_{\tau} \circ f \circ i_{\sigma}^{-1}$. A is 'rich enough' precisely if $i \circ f \circ i^{-1} \in A^{\sigma\to\tau}$ whenever $f \in A^{\sigma\to\tau}$. We shall call this the conjugation structure *generated by the applicative structure A, and the chosen actions on A^t and A^e.*

It's easy to see why conjugations structures are substitution structures. The only non-trivial thing to show is that the action commutes with application: $(if)(ia) = (i \circ f \circ i^{-1})ia = i(fi^{-1}ia) = i(fa)$.

Indeed, one can prove a representation theorem for functional substitution structures based on groups. Whenever M is a group, and (M, A) a functional substitution structure, then (M, A) is a conjugation structure. Thus when M is a group and A functional, the behavior of substitutions is uniquely pinned down by the behaviour of the action on base types.

Proposition 1 (Representation theorem)**.** *Let (M, A) be a substitution structure where M is a group and A is functional. Then (M, A) is a conjugation structure.*

Proof. Since A is functional it suffices to show that, for any $f \in A^{\sigma\to\tau}$, $(if)a = (i \circ f \circ i^{-1})a$ for all $a \in A^{\sigma}$. Since G is a group, i has an inverse i^{-1} with $a = 1a = i(i^{-1}a)$. Then $(if)(a) = (if)(i(i^{-1}a))) = i(f(i^{-1}a) = (i \circ f \circ i^{-1})a$

□

In general the notion of a conjugation structure does not make sense when A is not functional, but one can nonetheless prove the analogous constraint that $|if|$ must be identical to $i \circ |f| \circ i^{-1}$, where $|h|$ is the function determined by h: $a \mapsto App(h, a)$.[16]

---

[15]If j is the inverse of i, it follows that $ji = 1$ (so it follows that an element's inverse is both its left and right inverse). Let k be the inverse of j so that $jk = 1$. $ij = 1$, so $j(ij)k = jk = 1$, so $(ji)(jk) = 1$ (by associativity), so $ji = 1$ (since $jk = 1$). Moreover, if $ij = ik = 1$ then $j = k$, so everything has a unique inverse. Since $ik = 1$, $j(ik) = j$, so $(ji)k = j$. Since j is a left inverse of i as well, $k = 1k = (ji)k = j$ as required.

[16]Suppose, moreover, that A contains the composition combinator (i.e. B), and that for each $i \in M$, there is an element $g_i^{\sigma} \in A^{\sigma\to\sigma}$ such that $ia = App(g_i^{\sigma}, a)$ for all $a \in A^{\sigma}$. Then setting if to be $B(Bg_i f)g_{i^{-1}}$ generates a conjugation structure of this sort.

Proposition 1 thus gives a complete characterization of functional substitution structures when M is a group. In section 4.1 we shall see that this representation theorem is an instance of a more general representation theorem that provides a characterization of substitution structures based on arbitrary monoids. Crucial to this more general representation theorem is the idea that elements of the functional types can have more structure than the functions they determine between their argument and target types. Such structures are thus not functional, but they do possess a feature that is closely related to functionality:

Definition 7 (Quasi-functionality)**.** *A substitution structure* (M, A) *is* quasi-functional *iff*

- *For* $f, g \in A^{\sigma\to\tau}$, $f = g$ *if and only if* $App(if, a) = App(ig, a)$ *for every* $a \in A^{\sigma}$, $i \in M$.

This weakening of functionality imposes a more demanding condition for when properties are identified, which can be illustrated as follows. Consider the property *has three vertices* and the property *has three edges*. The principle of functionality would identify these properties given the working assumption that the proposition that a has three vertices is the same as the proposition that a has three edges for any individual a (in virtue of both propositions saying that a is a triangle).

But quasi-functionality does not require the identification of these properties, even given this assumption. For, even if we are unsure about whether to identify *having three vertices* with *having three edges*, *being a vertex* is certainly not the same property as *being an edge* (they are not even coextensive). So intuitively there could be a substitution that mapped *being a vertex* to *being a face*, but mapped *being an edge* to itself. This substitution would intuitively map *having three vertices* to *having three faces*, and *having three edges* to itself. But clearly you can have three edges without having three faces, so the resulting two properties are different. Since applying a substitution to identical inputs should yield identical results, it follows that *having three edges* is distinct from *having three vertices*. This is all consistent with quasi-functionality: what quasi-functionality *does* require, in order to identify these properties, is that for any substitution i, and any individual a, *a has i(three vertices)* is identical to *a has i(three edges)*. But we have just seen that this condition does not hold since when i is the substitution described above, and a an individual with three edges but not three faces (i.e. a triangle), we get that *a has i(three vertices)* has a different truth value to *a has i(three edges)*, and so these two propositions are thus different.

Note that the distinction between functionality and quasi-functionality evaporates if M is a group:

Proposition 2. *Suppose that* M *is a group. A substitution structure of the form* (M, A) *is functional if and only if it is quasi-functional.*

Proof. Clearly any functional structure is quasi-functional. To show quasi-functionality implies functionality, suppose that $f, g \in A^{\sigma\to\tau}$ and $App(f, x) = App(g, x)$ for every $x \in A^{\sigma}$. Let $i \in M$ and $a \in A^{\sigma}$. By supposition $App(f, i^{-1}a) = App(g, i^{-1}a)$, so $iApp(f, i^{-1}a) = iApp(g, i^{-1}a)$, so $App(if, ii^{-1}a) = App(ig, ii^{-1}a)$, so $App(if, a) = App(ig, a)$. Since i and a were arbitrary, we may conclude that $f = g$, by quasi-functionality. □

Indeed, we can replace the assumption that M is a group in the above argument with the weaker assumption that, for each $i \in M$, the function i determines from A^{σ} to A^{σ} is surjective for each σ (just replace $i^{-1}a$ in the above with any b such that $ib = a$). Thus quasi-functional substitution structures with *surjective* actions are always functional.

It's worth noting that in a non-functional applicative structure the equations in definition 1 for s and k do not necessarily pin down an element of the structure uniquely: for example, there might be distinct elements k and k' such that $k'xy = x = kxy$ for every x and y. Nonetheless, there is a natural analogue of *having combinators* for quasi-functional substitution structures:

Definition 8. *A quasi-functional substitution structure* (M, A) has combinators *iff for every* σ, τ *and* ρ *there are elements* $k \in A^{\sigma\to\tau\to\sigma}$ *and* $s \in A^{(\sigma\to\tau\to\rho)\to(\sigma\to\tau)\to\sigma\to\rho}$*, as in definition 1, such that*

– *For all* $i \in M$*,* $x \in A^{\sigma}, y \in A^{\tau}$ $(ik)xy = x$

– *For all* $i \in M$*,* $x \in A^{\sigma\to\tau\to\rho}$*,* $y \in A^{\sigma\to\tau}$ *and* $z \in A^{\sigma}$*,* $(is)xyz = (xz)(yz)$*.*

We can also impose a similarly more stringent constraint on interpretations of λ terms. Let (M, A) be a quasi-functional substitution structure with combinators. Given an interpretation function $[\![\cdot]\!]$ in A defined only on a signature Σ, let us write $[\![\cdot]\!]_i$ for the variant interpretation generated by setting $[\![c]\!]_i = i([\![c]\!])$ (that is, $Sub^{\sigma}(i, [\![c]\!])$) for each constant $c \in \Sigma^{\sigma}$. We may then simultaneously extend these interpretation functions to arbitrary terms. This proceeds exactly as in section 1.1, except we modify the clause for λ-abstraction:

- $[\![c]\!]^g = [\![c]\!]$ for $c \in \Sigma$
- $[\![x]\!]^g = g(x)$ for $x \in Var$.
- $[\![\alpha\beta]\!]^g = App^{\sigma\tau}([\![\alpha]\!]^g, [\![\beta]\!]^g)$
- $[\![\lambda x\, \alpha]\!]^g$ is the unique f in $A^{\sigma\to\tau}$ such that $(if)a = [\![\alpha]\!]_i^{(i\circ g)[x\mapsto a]}$ for all $i \in M$ and $a \in A^{\sigma}$

The existence of f follows from the existence of combinators in the more stringent sense, and the uniqueness follows from quasi-functionality.[17]

To summarize the main points of this section, we have given a complete characterization of substitution structures that are functional and based on groups. We have also identified a more general class of substitution structures that are quasi-functional and based on arbitrary monoids, and have suggested that a characterization of these structures might be available with a generalization of the notion of a conjugation structure (a suggestion we make good on in section 4.1).

2.2 Faithfulness

Note that $A^{\sigma A^{\sigma}}$ is itself a monoid under function composition.

Definition 9. *An* M*-set* (A, μ) *is* faithful *iff the monoid homomorphism* $\phi : M \to A^A$ *given by* $i \mapsto \mu(i, \cdot)$ *is an embedding. That is: if* $ia = ja$ *for every* $a \in A$*, then* $i = j$*.*

It is strongly faithful *iff if for any* a*,* $ia = ja$*, then* $i = j$*.*[18]

In what follows we shall pay special attention to substitution structures that are at least faithful on type t: that is, substitution structures where (A^t, Sub^t) is a faithful M-set. For these are structures that correspond closely to fine-grained accounts of reality: different substitutions differ at least in their action on at least one proposition. We do not

[17] Note: although we defined $[\![\cdot]\!]$ in terms of $[\![\cdot]\!]_i$, our definition is properly inductive, since we are simultaneously defining $[\![\cdot]\!]_i$ *for all* i at once (where $[\![\cdot]\!] = [\![\cdot]\!]_1$).

[18] See Kilp et al. [19] chapter 1, §4.

make the same requirement for type e. While substitutions of individuals had better move individuals, we shall not think of individuals as composed of properties and relations, and so substitutions that only move propositions or relations, for example, need not be faithful on type e. (Sometimes we will only impose the weaker condition that $(A^e \cup A^t, Sub^e \cup Sub^t)$ is faithful.[19])

Here is a useful proposition about faithful substitutions:

Proposition 3. *Let (M, A) be a substitution structure, where A is a quasi-functional structure with combinators such that A^σ is nonempty for all σ. Then the action of M is faithful on type $\sigma \to \tau$ if it is faithful on type τ.*

Proof. Assume, for induction, that Ms action is faithful on A^τ. Now suppose $i(f) = j(f)$ for all $f \in A^{\sigma\to\tau}$. In particular, $i(ka) = j(ka)$ for all $a \in A^\tau$, so $i(ka) = ik(ia) = k(ia)$ since k is a sink (that is, $ik = k$ — see next section), and similarly for j. So $k(ia) = k(ja)$, and thus $ia = ja$ for all $a \in A^\tau$ (by applying $k(ia)$ to any element of A^σ). By inductive hypothesis it follows that $i = j$, as required. □

Let me record two important consequences of this fact. It follows that (i) if M is faithful at type t it is faithful at all relational types[20], and (ii) if M is faithful on the base types (as in example 2, but not example 4 below) it is faithful on all types.

2.3 Examples of substitution structures

Note that we usually understand substitutions as things that can be applied to linguistic entities. One might have thought that the idea of a substitution makes no sense when applied to a coarse-grained theory of propositions, such as one in which they are just sets of possible worlds (or more generally, any theory in which propositions satisfy the Boolean identities). In fact it does make sense as the following two examples demonstrate. Let's begin by trying to make the notion of substituting one individual for another in a (set of worlds) proposition precise.[21]

Example 2 (Finean Structures)**.** *Let A be the full, functional applicative structure given by letting A^e be any set, and $A^t = P(W)$ for some set of worlds W.*

Suppose M is a group of permutations of A^e, the individuals. Suppose that we also have an action of M on W. For each $i \in M$, and possible world $w \in W$, we will informally interpret iw as the unique world that is qualitatively identical to w but in which the individuals occupy permuted qualitative roles, as dictated by i.[22] Each such permutation lifts to an automorphism of $P(W)$, giving us natural actions on the base domains:

$$Sub^t(i, p) = ip := \{iw \mid w \in p\}.$$

$$Sub^e(i, a) = ia$$

[19] Note that this does not require either (A^t, Sub) or (A^e, Sub) to be faithful individually.

[20] The relational types are defined inductively as follows: e and t are relational types, and $\sigma \to \tau$ is a relational type whenever σ and τ are relational types, and σ is not e.

[21] This substitution structure is used in Bacon [4] for modeling the distinction between qualitative and haecceitistic propositions; further discussion of the application can be found there. The technology is originally due to Fine [11] (see also Fritz and Goodman [15] for further developments).

[22] Note that formally speaking, every choice of action of M on W generates a separate example. But on the intended application we are suppose to think of W as the set of metaphysically possible worlds, and the action of a permutation of individuals on a world determined by the informal description above.

This determines a unique conjugation structure, where the action on type $\sigma \to \tau$ is given by conjugation (that is, $if = i \circ f \circ i^{-1}$).

In order for an action on worlds to be suitable we shall require that the action of M on W be faithful (equivalently, we may require that the derived action of M on $P(W)$ be faithful).

We may intuitively think of $i(p)$ as the result of substituting the individuals that occur in p with their i correspondents. We can lift substitutions to other types by conjugation (i.e. $if = i \circ f \circ i^{-1}$).

If our structure is faithful at type t then that means that every permutation of individuals generates a distinct permutation of worlds. In particular, the idea of permuting the roles of arbitrary individuals within a world (to get a different world) makes sense for any permutation of individuals. Indeed, one might argue that the intuitive idea motivates the constraint that the structure should be strongly faithful at type t: distinct permutations of individuals should always generate distinct worlds when applied to a given world w.

Both of these thoughts rely on some substantive metaphysics. The latter idea, for example, requires that we reject the possibility of worlds containing qualitative symmetries. For example, Max Black argued that there could be two qualitatively identical spheres in otherwise empty space (see Black [5]). Call this possibility w. If w was indeed possible it would violate the strong faithfulness condition, since the identity permutation, and a permutation that swaps the two spheres, would both map w to itself. So these two permutations are distinct even though agree on what they do to the proposition $\{w\}$.

Even the plain faithfulness condition encodes some substantive metaphysics. Suppose that M is the group of all permutations of individuals. Then, for every substitution, i, it's possible that things occupy qualitative roles that have been switched according to i: iw will be a world where each individual $a \in A^e$ occupies the qualitative role that $i(a)$ occupies in w. In particular, if i is a permutation that switches me with a boiled egg (call it Egg), and leaves everything else alone, it follows that there must be some world w such that $iw \neq w$ (for otherwise i would agree with the identity substitution on every world, contradicting the faithfulness constraint). Thus we have a pair of qualitatively alike possibilities in which I have switched roles with Egg.[23]

The thought that I could have occupied the qualitative role of a boiled egg sounds a little wild when combined with the idea that the possibilities in question are metaphysical possibilities. But there is nothing forcing that identification: the idea that propositions form a complete atomic Boolean algebra is entailed by the view that propositions are sets of possible worlds, but is not equivalent to it. Indeed it is possible to consistently develop a theory in which there is a broadest necessity that is not identical to metaphysical necessity (see Bacon [3]). Another premise in this argument was the assumption that M was the group of all permutations of individuals. According to a variant construction the individuals are partitioned *kinds*, and M consists of only those permutations that preserve the kind of each individual.

To illustrate the scope of applications consider the following example from classical mechanics. (For more details on this example the reader may consult Arnold [2].)

Example 3 (Phase space)**.** *Let $(\mathcal{M}, \omega)$ be a symplectic manifold and $H : \mathcal{M} \to \mathbb{R}$ a smooth function. Together these determine an abelian group $G(\mathcal{M}, \omega, H) = \{\phi_t \mid t \in \mathbb{R}\}$*

[23] Faithfulness was important for this result: if $iw = w$ for every w, then there is no good sense in which I can be said to have switched roles with a boiled egg, since the witness of the alleged switched possibility is just the world where I occupy the role I in fact occupy.

of diffeomorphisms on $\mathcal{M}$, parameterized by $\mathbb{R}$, which is obtained from the time flow of the Hamiltonian vector field generated by H.

To generate a full functional conjugation structure we just need to specify sets A^e and A^t and to specify the action of $G(\mathcal{M}, \omega, H)$ on them. As before we may take A^t to be $P(\mathcal{M})$ and endow it with the induced action $ip = \{iw \mid w \in p\}$. We shall let A^e be any set with the trivial action $\phi_t a = a$ for any $a \in A^e$ and $t \in \mathbb{R}$.

Informally we may think of the points of $\mathcal{M}$ as states of a physical system with H assigning an energy to each state. ϕ_t takes a state w and returns the state that w will evolve to in t units of time (or has evolved from if t is negative). As one would expect from this informal interpretation, the group is subject to the law: $\phi_{t+s} = \phi_t \circ \phi_s$.

Here is another example that illustrates the idea of making substitutions in the context of a Boolean view of propositions. Here we make sense of the idea of substitution propositions in propositions, rather than individuals in propositions.

Example 4 (Lindenbaum Tarski structures)**.** *Let A^t be the Lindenbaum-Tarski algebra of a propositional language L, and A^e any fixed set. Now consider the full and functional applicative structure based on A^e and A^t.*

Any permutation of propositional letters induces a permutation on A^t. For a sentence, ϕ, we may understand $i[\phi]$ as the class of formulas logically equivalent to the result of performing an i substitution on ϕ.[24] *We now stipulate that i fixes each element of A^e, and that i raises to higher types by conjugation (this is possible because M forms a group).*

This example is faithful at type t but not strongly faithful. If $i[\phi] = j[\phi]$ for every ϕ, they must agree on their action on each equivalence class $[p]$ of a propositional letter, so i and j must be the same permutation of letters. However, for a given ϕ, i and j might agree about $[\phi]$, but be distinct because i and j do different things to letters not appearing in ϕ.

This example considers permutations of propositional letters, but there's a larger class of of substitutions that are well defined on A^t determined simply by arbitrary functions from propositional letters to sentences of L. The problem is we do not have a good way of extending these arbitrary substitutions up the type-theoretic heirarchy to higher types, because conjugation relies essentially on the existence of inverses.

We return to this issue in more generality in section 4.1. In the meantime, here is an example where the restriction to permutations is lifted.

Example 5 (Term Models)**.** *Let $\mathcal{L}(\Sigma)$ a typed language, and consider an equational theory S. We assume that provable identity, written $\vdash_S \alpha = \beta$, is an equivalence relation, and than S is closed under under the rule of substitution: if $\vdash_S \alpha = \beta$ then $\vdash_S i\alpha = i\beta$ for any substitution of the language i. (Examples that will interest us in what follows include the pure theory of $\eta\beta$ equivalence, higher-order logic, and higher-order logic with the rule of equivalence.*[25]*)*

Let T be the closed term model of $\mathcal{L}(\Sigma)$:

- *$T^\sigma = \{[\alpha] \mid \alpha \in \mathcal{L}^\sigma(\Sigma)\}$, where $[\alpha]$ is the equivalence class of α under the equivalence relation of provable identity in S (i.e. $\vdash_S \alpha = \beta$).*

---

[24] This is well-defined since the rule of substitution is admissible in the propositional calculus.

[25] As an anonymous referee points out, the restrictions we have placed on S preclude identities between distinct constants, but do not preclude S containing identities between distinct variables. The restrictions we have placed suffice for the construction of a well-behaved term model, but the theories of any interest to us are also closed under the rule of substitution for variables.

- $App^{\sigma\tau}([\alpha])([\beta]) = [\alpha\beta]$

Let M be the monoid of substitutions from example 1: type preserving mappings $\Sigma \to \mathcal{L}(\Sigma)$. Since S admits the rule of uniform substitution, this monoid generates an action on the elements of T as in example 4. A variant of this construction is the open term model for S in which α in the first bullet point is allowed to be open.

Eventually we will be interested in looking at substitution structures that are also models of higher-order logic. This is achieved by providing a valuation on the elements of type t that tells us whether each proposition is true or false. It's worth noting that while it's possible to turn term models of the sort described above into models of higher-order logic, they are often not *Leibnizian*: distinct elements of the model might end up counting as identical according to the model. Thus term models do not in general provide faithful models of the idea that propositions are as fine-grained as terms up to equivalence in some theory, unless one can show that they can be equipped with a Leibnizian valuation.

2.4 Congruences

Definition 10 (Congruence)**.** *A congruence, $\sim$, on an M-set A is an equivalence relation on A with the property that:*

- *If $a \sim b$ then $ia \sim ib$ for all $i \in M$*

Probably the most important examples of congruences in algebra are kernels of homomorphisms. If $f : A \to B$ is an equivariant mapping between M-sets A and B, then $a\,ker_f\,b$ iff $f(a) = f(b)$. There are versions of classical isomorphism theorems (see Kil'p [19]).

Definition 11 (Quotient)**.** *If $\sim$ is a congruence on A then $A/\sim$ is the M-set whose elements are equivalence classes of elements of A and whose action, μ, is given by:*

- $\mu(i, [a]) = [ia]$

The following notion is useful (see chapter 1 of Kil'p et al. [19]).

Definition 12 (The annihilator congruence)**.** *Consider M as an M-set with the action of composition on the left. If A is an M-set and $a \in A$, then the annihilator congruence on M is:*

- *$i \sim_a j$ if and only if $ia = ja$.*

$\sim_a$ is in fact the kernel of the equivariant map $\lambda_a : M \to A$ with $\lambda_a(i) = ia$

It is easy to check that if $i \sim_a j$ then $ki \sim_a kj$. We can think of the monoid $M/\sim_a$ as the set of substitutions that can be performed on a, ignoring differences between what substitutions do to other elements of A. Of particular interest is the case where the target M-set is of the form $A^{\sigma_1} \times ... \times A^{\sigma_n}$, and $\overline{a} = (a_1...a_n)$. In this $i \sim_{a_1...a_n} j$ iff $ia_m = ja_m$ for $m = 1, ..., n$. More generally we shall write $i \sim_Z j$ iff $iz = jz$ for all $z \in Z$, when Z is a set of elements from $\bigcup_\sigma A^\sigma$. As before, $M/\sim_Z$ can be thought of the set of substitutions that can be performed on elements of Z (the behavior of a substitution on elements not appearing in Z is 'washed out' by the equivalence relation).

In section 2.2 we introduced the notion of a faithful substitution structure. Non-faithful substitution structures can contain distinct substitutions, i and j, that agree on what they do to every element of the substitution structure. Using the annihilator congruence we can collapse every non-faithful substitution structure to a faithful one:

Definition 13 (Faithful collapse of a substitution structure)**.** *Suppose that (M, A) is a substitution structure. The* faithful collapse *of (M, A) is the substitution structure $(M/\sim_{\bigcup_\sigma A^\sigma}, A)$, where the action of $M/\sim_{\bigcup_\sigma A^\sigma}$ on A is the obvious one:*

- $Sub^\sigma([i]_\sim, a) = ia$ *for every $a \in A^\sigma$*

This clause is well-defined, since if $[i] = [j]$, so that $i \sim_{\bigcup_\sigma A^\sigma} j$, then $ia = ja$.

As was shown in proposition 3, if a monoid is faithful on $A^e \cup A^t$ and A is quasi-functional and has combinators, it is faithful at all types. Thus in many cases we only need to quotient by $A^e \cup A^t$, as opposed to the whole $\bigcup_\sigma A^\sigma$.

Example 6. *Consider the term substitution structure of example 5, where elements of A^σ are equivalence classes of $\eta\beta$-equivalent terms on a language $\mathcal{L}(\Sigma)$, and M is the monoid of substitutions on $\mathcal{L}(\Sigma)$. Suppose $i(c)$ is $\eta\beta$ equivalent to $j(c)$ for every $c \in \Sigma$ but for some c, $i(c)$ is distinct from $j(c)$. Then $i[\alpha] = j[\alpha]$ for every term α, and thus (M, A) is not faithful.*

On the other hand we may form a congruence $\sim$ on M, where $i \sim j$ iff $i(c)$ is $\eta\beta$ equivalent to $j(c)$ for every $c \in \Sigma$. $(M/\sim, A)$ is faithful.

2.5 M-logical relations

Here we introduce a generalization of the notion of a logical relation to substitution structures.[26]

Definition 15 (M-logical relation (for substitution structures))**.** *Given two substitution structure (M, A) and (M, B), an M-logical relation between (M, A) and (M, B) is a type indexed collections of relations $R^\sigma \subseteq A^\sigma \times B^\sigma$ subject to the following constraints:*

- *If $R^\sigma ab$ then $R^\sigma ia\, ib$ for any $i \in M$ and σ a base type.*
- *$R^{\sigma \to \tau} fg$ if and only if, whenever $a, b \in A^\sigma$, $i \in M$ and $R^\sigma ab$, $R^\tau (if)a\,(ig)b$*

The generalization to n-ary relations is the obvious one, although we shall not have need for it here.

It is easy to see the behaviour of an M-logical relation in this sense is determined by its behaviour on base types, and that one can prove that for *any* type σ, if $R^\sigma ab$ then $R^\sigma ia\, ib$ for any $i \in M$. For M-logical relations, we get a version of the usual fundamental theorem of logical relations (see Mitchell [23] chapter 8):

[26] See Mitchell [23] chapter 8 for an introduction to ordinary logical relations for applicative structures and Mitchell and Moggi [24] for Kripke logical relations on Kripke λ-structures. Our definition is a special case of a more general notion. Let C be an arbitrary category and consider the presheaf category Set^C (defined in section 4.1).

Definition 14 (Kripke logical relation)**.** *A binary Kripke logical relation on a presheaf category, Set^C, consists in a collection of relations $R^w_F \subset F(w) \times F(w)$ indexed by objects w in C and objects F in Set^C, subject to the following constraints:*

- *If $R^w_F ab$ and $i : w \to v$ then $R^v_F F(i)a\, F(i)b$*
- *$R^w_{F \to G} fg$ iff for all $i : w \to v$, $a, b \in F(v)$ if $R^v_F ab$ then $R^v_G (F(i)(f)a)\,(F(i)(g)b)$*

where $F \to G$ is the exponential object $Hom(y - \times F, G)$. This has Mitchell and Moggi's [24] notion of a Kripke logical relation as a special case where C is a preorder, and our notion as a special case in which C is a monoid considered as a category with one object.

Theorem 4. *Suppose that (M, A) and (M, B) are substitution structures with combinators, and R is a logical relation between (M, A) and (M, B). Suppose moreover that $[\![\cdot]\!]^A$ and $[\![\cdot]\!]^B$ are interpretations of a λ-language $\mathcal{L}(\Sigma)$ in A and B respectively such that:*

- $R[\![c]\!]^A[\![c]\!]^B$ *for every constant* $c \in \Sigma$

Then it follows that

- $R[\![\alpha]\!]^A\,[\![\alpha]\!]^B$ *for every closed term* $\alpha \in \mathcal{L}(\Sigma)$

I omit the proof, as the argument is standard (see Mitchell [23] chapter 8). Note that our theorem is restricted to substitution structures based on the same monoid M (for otherwise there are no M-logical relations between them).

A partial equivalence relation on a set A is a transitive symmetric relation.[27] Given a substitution structure (M, A), one can always generate a quasi-functional substitution structure, $(M, A)/\sim$, by quotienting by a partial equivalence relation. For consider the logical relation $\sim$, between (M, A) and itself, generated by the identity relation on base types ($p \sim^t q$ iff $p = q$, $a \sim^e b$ iff $a = b$). By induction one can see that $\sim^\sigma$ is a partial equivalence relation on A^σ for arbitrary σ, and moreover, $\sim^\sigma$ is a congruence of M-sets (recall definition 10). The substitution structure $A/\sim$ may be defined as follows:

- $A^\sigma/\sim = \{[a]_\sim \mid a \in A^\sigma\}$
- $i[a]_\sim = [ia]_\sim$
- $App([f]_\sim, [a]_\sim) = [App(f, a)]_\sim$

Since $\sim$ is a congruence on the action, and a congruence with respect to application, the second two stipulations are consistent. Now suppose that $App(i[f], [a]) = App(i[g], [a])$ for all $i \in M, a \in A^\sigma$. In our original structure A, this means that whenever $a \sim b$, $(if)a \sim (ig)b$. This is exactly the condition for $f \sim g$, thus $[f] = [g]$ as required.

(This theorem is an analogue of the functional collapse theorem for applicative structures with respect to ordinary logical relations (see Gandy [16]). If you quotient an applicative structure by the ordinary logical relation generated by identity at the base types, you get a functional structure. Notice that we merely get a quasi-functional structure. See Mitchell corollary 8.2.25 for more details.)

3 Metaphysical Notions

In this section a number of metaphysically significant concepts are informally introduced and given substitution theoretic analyses: the notion of a *pure* property, proposition, relation, etc., the notion of a fundamental basis of properties, relations, etc. and the notion of a fundamental language.

3.1 Pure elements

If we are interested in principles governing fineness of grain, the following is an important concept.

[27] Thus, it can be considered an equivalence relation on a subset of A.

Definition 16 (Purity)**.** *An element $a \in A$ of an M-set is a* pure element *(or a* sink*) iff $ia = a$ for all $i \in M$.*

Intuitively a pure element is an element that is left alone by every substitution. Thus, intuitively, a pure element is an entity that has no constituents. Since on a strictly structured picture of propositions, properties, relations, and so on, everything is constructed from simple constituents, one might characterize the strict structured vision by the thesis that there are no pure elements. After all, since it is always possible to substitute simple elements for other elements, the only way something could be left alone by every substitution is if it did not contain any simple constituents.

But there are other positions about propositional fineness of grain that are less stringent than the structured vision. One might, for example, entertain a quasi-structured picture that allowed for *combinators*.[28] Such a view is inconsistent with a strict structured theory: if a of type σ is simple, and f is of type $\sigma \to \sigma$, then fa is structured, and thus an identity of the form $fa = a$ can never hold since a structured thing cannot be identical to a simple thing. Thus a structured proposition theorist must reject the identity combinator $\lambda x\, x$, insofar as it is governed by the identity $(\lambda x\, x)a = a$. A quasi-structured view might relax the idea that applying one thing to another always introduces structure, and an especially attractive version of this idea maintains that combinators are pure elements. Indeed, for any i, $i(\lambda x\, x)ia = i((\lambda x\, x)a) = ia$, and so $i(\lambda x\, x)$ maps every element of the form ia to itself. This condition is clearly satisfied if $\lambda x\, x$ is pure, for then $i(\lambda x\, x)$ just *is* the identity function. (The term structure of example 5 under the theory of $\beta\eta$-equivalence is a natural model of the sort of quasi-structured picture described here.)

Indeed, it is not hard to show that in a quasi-functional structure, the identity combinator — indeed *any* combinator — is a pure element. This is the fundamental property of pure elements:

Proposition 5. *Suppose that (M, A) is a substitution structure, and that A is a quasi-functional applicative structure with combinators (thus every λ-term has an interpretation). Then every combinator is a pure element.*

This can be proved by a simple induction on term structure. For any interpretation $[\![\cdot]\!]$ of $\mathcal{L}(\emptyset)$ in substitution structure (M, A), variable assigment g and $i \in M$: $i([\![\alpha]\!]^g) = [\![\alpha]\!]^{i \circ g}$.[29] (The special case of this result for functional structures follows from the fact that the relation Rab iff $ia = b$ is a logical relation, for any $i \in M$ (see Mitchell [23] chapter 8).)

Beyond the combinators, there are other pictures of granularity that suggest further things should be counted as pure. Here are two examples:

Example 7 (The logical connectives)**.** *Considering the monoid of substitutions on propositional letters in example 4. The Boolean connectives are all pure: for example $i \circ \neg \circ i^{-1}([p])$ is the result of performing an i substitution on $[\neg i^{-1} p]$, which is just $[p]$, so every i fixes negation. Similarly, since we don't consider substitutions that move individuals, all individuals are pure elements.*

[28] Recall that we are using *combinator* to mean anything denoted by a closed λ-term constructed only out of variables and λ.

[29] The only non-trivial case is the case for λ abstraction. For arbitrary $j \in M$ and $a \in A^\sigma$, recall that $(ji([\![\lambda x \alpha]\!]^g))a = [\![\alpha]\!]^{(j i \circ g)[x \mapsto a]}$ by the clause for λ-abstraction in a quasi-functional substitution structure (see section 2.1). Similarly, $j([\![\lambda x \alpha]\!]^{i \circ g})a = [\![\alpha]\!]^{(j i \circ g)[x \mapsto a]}$. Thus we've shown that $(ji([\![\lambda x \alpha]\!]^g))a = (j([\![\lambda x \alpha]\!]^{i \circ g}))a$ so by quasi-functionality $i([\![\lambda x \alpha]\!]^g) = ([\![\lambda x \alpha]\!]^{i \circ g})$. More generally, for arbitrary terms one can show $i([\![\alpha]\!]^g) = [\![\alpha]\!]_i^{i \circ g}$.

Example 8 (Qualitativeness as purity)**.** *When M consists only of substitutions of individuals, as in example 2, then intuitively the qualitative entities are pure. The idea here is simple: if an entity is qualitative — it doesn't have individuals as constituents — then it will not be moved by any substitution that only moves individuals.*

Thus, in example 2, we may simply define *the qualitative elements of a substitution structure to be the pure elements.*

Example 9 (Conserved properties)**.** *Consider the substitution structure of example 3. Here the pure elements intuitively correspond to conserved properties.*

For example, a pure proposition — represented as a set of states in phase space — is a proposition that is either always true or always false, such as, for example, a proposition about the energy of the system.

It is also straightforward to see that if you apply one pure element, f, to another, a, the result is also pure: for any $i \in M$, since $if = f$ and $ia = a$ it follows that $i(fa) = (if)(ia) = fa$, so fa is fixed by every substitution. Thus, one can prove the following more general proposition:

Proposition 6. *If (M, A) is a quasi-functional substitution structure with combinators, then anything definable from pure elements by combinators is also a pure element.*

The following example illustrates the point, made earlier, that there are no pure elements according to the flatfooted structured view:

Example 10 (Structured propositions)**.** *A natural way to model structured propositions as tuples proceeds as follows. For each type σ, let S^σ denote a set, which we will think of as representing the* simple *(or* fundamental*) entities of type σ. Let A be the smallest typed collection of sets such that*

- *$S^\sigma \subseteq A^\sigma$ for each type σ*
- *Whenever $f \in A^{\sigma\to\tau}$ and $a \in A^\sigma$, the ordered pair $(f, a) \in A^\tau$.*

Thus, for example, if $\neg, \Box \in S^{t\to t}$ and $p \in S^t$, then $(\neg, (\Box, p)) \in A^t$.

Let M consist of the collection of typed functions, i^σ, that take $S^\sigma \to A^\sigma$, where $i \circ j$ is the function that maps $a \in S^\sigma$ to the result of applying an i substitution to ja. The action of i on arbitrary elements of A^σ is defined inductively by $i(f, a) = (if, ia)$.

There are no pure elements in this structure.

It should be of no surprise that this applicative structure does not have combinators: there is no element of $A^{t\to t}$, for example, whose applicative behavior matches that of the identity combinator (application in this structure always increases complexity).

One moral we might draw from this is that, since an important component of the structured picture is the thesis that there are no pure elements, a structured vision is incompatible with the λ-calculus. After all, one can construct constituentless terms in the λ-calculus using only λ and bound variables (combinators) such as for example $\lambda X \lambda y\, Xy$. This suggests, however, that there is a spectrum of quasi-structured views that admit a well-defined notion of substitution, but that are consistent with the λ-calculus in allowing for constituentless entities denoted by combinators.

Here is an intermediate example which admits the combinators as pure, but nothing else:

Example 11. *Let (M, A) be the closed term model of $\eta\beta$-equivalent closed terms from example 5, and suppose that $|\Sigma^\sigma| \neq 1$ for every σ. Then an element of A is pure if and only if it is a combinator. That is: every pure element is of the form $[\alpha]$ for some closed term α not containing constants.*

Here we rely on the standard result that every term is $\eta\beta$ equivalent to a unique term in '$\eta\beta$ normal form', and that substituting constants for constants in a term in $\eta\beta$ normal form will produce another term in $\eta\beta$ normal form.[30] Suppose $[\alpha]$ is not a combinator, and α is in $\eta\beta$ normal form, then α contains a constant $c \in \Sigma^\sigma$. Since there is another constant $c' \in \Sigma^\sigma$, $\alpha[c'/c]$ is also in $\eta\beta$ normal form and thus not $\eta\beta$ equivalent to α, by the uniqueness of $\eta\beta$ normal forms. Thus $[\alpha]$ is moved by at least one substitution and is not pure. This means that the only pure elements are combinators.[31]

Remark 1. *Suppose that (M, A) is a substitution structure, that A is functional and that there are elements and $\in A^{t\to t\to t}$ and not $\in A^{t\to t}$ corresponding to conjunction and negation. Suppose, moreover, that the Boolean identities are satisfied at type t. Then one might have thought that all Boolean operations are pure elements. However this is not always true: for example the mapping $i : [\phi] \mapsto [\phi^*]$ in example 4 (the Lindenbaum Tarski algebra) that takes a formula to its dual (e.g. flips disjunction and conjunction, $\top$ for $\bot$ etc) generates a mapping that could added to the substitutions in that example. This corresponds to a well-defined substitution on equivalence classes since if two sentences are equivalent in the propositional calculus, so are their duals. In this case the top element of the Boolean algebra, $[\top]$, would not be a pure element.*

3.2 Metaphysical definability

This framework can also be used to introduce some notions useful for metaphysical theorizing. The following terminology is repurposed from Bacon [4].[32]

Definition 17 (Metaphysical definability)**.** *Suppose that (M, A) is a substitution structure. Let $X \subseteq \bigcup_\tau A^\tau$ and let a be some element of A^σ. a is* metaphysically definable *from X if and only if, every $i \in M$ that fixes X fixes a.*

(Say that i fixes X iff $ix = x$ for all $x \in X$, and fixes a if $ia = a$.)

Here is the intuition behind the definition. The thought is that substitutions that leave the constituents of a alone must leave a alone. Thus if the constituents of a are contained in a set Z, then any substitution that fixes Z fixes a. According to a natural analysis of 'constituenthood', the converse should hold as well: when any substitution that leaves Z alone leaves a alone, the constituents of a must be contained in Z.

Remark 2. *Note that something is definable from the empty set iff it is pure.*

[30] The former fact is a consequence of the confluence and strong normalization theorems. The latter fact is immediate from the definition of normal form. See, e.g., Mitchell [23] §4.4.2.

[31] This structure appears to offer a very fine-grained picture of reality, and has a minimal number of pure elements. However, Cian Dorr has shown that it is not possible to turn it into a model of higher-order logic in which distinct elements of the model are distinct from the perspective of the model itself: $x = y$ can come out true in the model relative to an assignment that assigns x and y distinct elements. Thus it appears as though this structure doesn't guarantee the coherence of a metaphysical picture in which entities are individuated as finely $\eta\beta$-equivalence.

[32] In that paper a particular substitution structure — discussed in example 2 — was being supposed. The following is a general definition suitable for an arbitrary substitution structure.

Remark 3. *For $a \in A^\sigma, f \in A^{\sigma \to \tau}$, fa is metaphysically definable from $\{f, a\}$ (and thus any super set). More generally, anything that results from combining elements of X together using application will be metaphysically definable from X.*

Remark 4. *If $f : \sigma_1 \to ...\sigma_n \to \tau$ is pure, and $f(a_1)...(a_n) = d$ then d is metaphysically definable from $a_1...a_n$. Since f is pure, i fixes f for any i. If i moreover fixes $a_1...a_n$, then $i(f(a_1)...(a_n)) = (if)(ia_1)...(ia_n) = f(a_1)...(a_n)$.*

It follows that if (M, A) is a functional applicative structure with combinators, and $[\![\cdot]\!]$ is some interpretation of a λ-language in A, then: $[\![\alpha]\!]$ is metaphysically definable from $[\![c_1]\!]...[\![c_n]\!]$ where $c_1...c_n$ are the constants appearing in α. (This follows because you can 'λ out' the constants appearing α to get a combinator expression which takes you from $c_1...c_n$ to α. By proposition 5, every combinator expression denotes a pure element.)

Example 12. *Suppose that $|\Sigma^\sigma| \neq 1$ for any σ, and consider the closed term model based on $\eta\beta$-equivalent closed terms of $\mathcal{L}(\Sigma)$ (example 5). We have already shown that an entity is metaphysically definable from the empty set iff it's a combinator (example 11).*

More generally we have the following. Suppose γ is in $\eta\beta$ normal form, then $[\gamma]$ is metaphysically definable from a finite set Γ iff γ is definable in the λ calculus using only constants that appear in the $\eta\beta$-normal forms from the equivalence classes appearing in Γ.[33]

Note that metaphysical definability does not always correspond to definability in the λ-calculus. Consider Churches δ function, of type $\sigma \to \sigma \to \tau \to \tau \to \tau$, where $\delta abcd$ outputs c if $a = b$ and d otherwise. It can be shown that Church's δ function is fixed by every substitution in any conjugation structure, so δ is metaphysically definable from the emptyset, but it is not λ definable.[34]

Example 13 (Structured propositions)**.** *Consider the substitution structure of example 10, where every element is a tuple of simple elements of the structure. A tuple f of type σ is metaphysically definable from exactly those sets that contain fs simple constituents. For example $(\Box, (\neg, p))$ is metaphysically definable from any set containing $\Box, \neg$ and p.*

Note, on the other hand, that $(\Box, (\neg, p))$ is also metaphysically definable from sets that do not contain all of its simple constituents. For example, it is metaphysically definable from $\{\neg, (\Box, p)\}$ and $\{(\neg, (\Box, p))\}$.

This latter example demonstrates that metaphysical definability can behave strangely when the basis set contains non-simple entities (such as $(\Box, p)$), so we shall often impose the assumption that X consists only of simple entities when we assert that something is metaphysically definable from X. (Note that we have not yet given an account of simplicity.)

The notion of metaphysical definability shares some affinities with certain hyperintensional notions appealed to in recent metaphysics, (e.g. ground, metaphysical priority etc). However, it can hold between things of arbitrary type (see Dorr's notion of priority [9]). One might have thought such a notion would be well-founded; however whether metaphysical definability is well-founded, in a given substitution structure, is closely connected to how closely it adheres to the structured vision of propositions. For instance, in example 13, no two simple things can be defined from one another. However, in substitution structures with pure elements, you can have failures of antisymmetry

[33] The argument here is similar to that of example 11. If γ contains a constant c not appearing in the normal forms of Γ, then a substitution that maps c to a distinct constant c' moves $[\gamma]$ but leaves the elements of Γ alone. On the other hand, if i fixes all elements of Γ, it must fix all the constants appearing in the normal forms of Γs elements, so if you can define γ from the constants appearing in the normals forms of Γs elements, i must fix $[\gamma]$ too.

[34] For the latter result, see Plotkin [26] p12.

Example 14. *Suppose (M, A) is a substitution structure, and that A is functional and has combinators. Then $R : e \to e \to t$ is metaphysically definable from the singleton of its converse, and* conversely*. Let C be the converse combinator, then any i that fixes R, also fixes CR since $i(CR) = i(C)i(R) = CR$ since C is a pure element.*

More generally, if $a \in A^\sigma$ and $b \in A^\tau$ are distinct, $f \in A^{\tau\to\sigma}, g \in A^{\sigma\to\tau}$ two pure elements, and $a = fb$ and $b = ga$, then a is metaphysically definable from the singleton of b and conversely.

3.3 Free objects

According to a naïve structured theory of propositions there is an distinction between *simple* properties — properties without any proper constituents — and *complex* properties. It is natural to ask to what features of simple entities extend to other structures that aren't straightforwardly structured. A natural way to approach this question is to look at features that simple entities have in structured applicative structures (as in, e.g., example 10, and the linguistic model in example 1) and see if we can characterize them by their substitution theoretic features.

Here is one feature: languages are *freely generated* by the syntactically simple terms (the constants). Consider the set of closed terms in some language $\mathcal{L}(\Sigma)$. The constants Σ of that language have a very special feature. Let $\mathcal{L}(\Sigma')$ be any other language. Then any type preserving mapping $f : \Sigma \to \mathcal{L}(\Sigma')$, can be uniquely extended to a (type preserving) translation $f^+ : \mathcal{L}(\Sigma) \to \mathcal{L}(\Sigma')$ (a translation, in this case, is a mapping from closed terms to closed terms that commutes with application and λ-abstraction). Indeed, the constants are the unique set of terms in $\mathcal{L}(\Sigma)$ that have this feature.

It's worth emphasizing that the notion of a freely generated language is always relative to a class of languages. Above we have observed that $\mathcal{L}(\Sigma)$ is freely generated by Σ relative to the class of all languages. In example 1, in section 1.2, we defined a substitution as a function $i : \Sigma \to \mathcal{L}(\Sigma)$ and noted that any such function has a unique extension to a function $i^+ : \pm \to \mathcal{L}(\Sigma)$. Thus $\mathcal{L}(\Sigma)$ is also generated by Σ relative to the singleton class of $\mathcal{L}(\Sigma)$ alone.

Two more examples are illustrative. In the Lindembaum-Tarski algebra (example 4) the fundamental propositions are intuitively the equivalence classes, $[p]$, of sentence letters. These have a structurally similar property. For any other Boolean algebra, B, and function f from the equivalence classes of propositional letters to B, there is a unique Boolean homomorphism from A (the LT algebra) to B that commutes with the function. Finally a non-linguistic example: a vector spaces V is freely generated by any basis, B, for that vector space, relative to the class of all vector spaces. Since for any vector space U, any function $f : B \to U$ can be uniquely extended to a linear mapping $f^+ : V \to U$.

Definition 18 (Free object)**.** *Suppose $\mathcal{C}$ is a class of algebras, with an associated class of homomorphisms between any two algebras in the class. If $A \in \mathcal{C}$ we say that $X \subseteq A$ freely generates A relative to $\mathcal{C}$ iff, for any other algebra $B \in \mathcal{C}$, and function $f : X \to B$, there is a unique homomorphism $h : A \to B$ such that $h\restriction_X = f$.*

Thus, X freely generates A iff every function from X to an algebra B can be extended to a unique homomorphism from A to B.

It's worth noting that in the linguistic example, the set of constants is the only set X with the property described above. However in the Lindenbaum-Tarski algebra, the 'freeness role' does not pin the set X down uniquely: e.g. the classes of the form $[\neg p]$ where p is a propositional letter also freely generates the Lindenbaum-Tarski algebra. Nonetheless, freeness seems like an important constraint on being a set of fundamental entities.

3.4 Fundamentality and cofundamentality

Here we formulate this feature of fundamental entities using notions internal to a substitution structure. In section 3.5 we'll give an alternative characterization in terms of fundamental languages.

Definition 19 (Basis)**.** *Let A be an M-set and X a non-empty subset of A. X (freely) generates A iff, for every function $f : X \to A$, there is a unique $i \in M$ such that $ix = fx$ for all $x \in X$. In this case we call X a fundamental basis of A (or just a basis). An M-set is generated iff it has a basis.*

A has a unique generator iff there is a unique X that generates A.

The notion of a basis can be extended to a substitution structure. Let $A = \bigcup_\sigma A^\sigma$, then M acts on A in the obvious way. A basis is a subset X of A such that every *type preserving* function from $X \to A$ corresponds to a unique element of M. If X is a basis of A then we say that A is (freely) generated by X.

In what follows, if X is a basis of (M, A), $X \cap A^\sigma$ can be thought of as a candidate set of fundamental entities of type σ. Here the guiding intuition draws on the parallel between (quasi-)structured theories of propositions and languages: fundamental entities stand in the same relation to a structured reality as constants stand to a language. For that reason it is desirable to consider substitution structures that have a basis. One may think of the above as a way of characterizing the notion of a fundamental entity in substitution theoretic terms.[35]

Example 15 (Substitutions)**.** *Consider the monoid of closed substitutions acting on the closed terms of a language $\mathcal{L}(\Sigma)$ considered as a substitution structure (see example 1 – recall that a closed substitution only takes closed terms as values). Suppose, moreover, that $\Sigma^\sigma \neq \emptyset$ for every σ. The the set of constants, Σ, is a unique basis for $\mathcal{L}(\Sigma)$. By definition every function from constants to closed terms of the same type defines a substitution, and no two distinct substitutions can agree on their behaviour on constants. Moreover, Σ is the unique basis. If X is a basis then X must contain only constants. For if a complex closed term α belonged to X, then functions $X \to \mathcal{L}(\Sigma)$ that map α to a closed term that is not a substitution instance of α do not correspond to a substitution.[36] (By contrast, every term of type σ is a substitution instance of a constant of type σ.) On the other hand, if X were a proper subset of Σ, then a given function from $X \to \mathcal{L}(\Sigma)$ may correspond to multiple substitutions that differ on what they do to $\Sigma \setminus X$.*

Note that when we move from the language $\mathcal{L}(\Sigma)$ to the $\eta\beta$ term structure based on $\mathcal{L}(\Sigma)$, the equivalence classes of constants in Σ still generate the term structure, but it is not in general the unique set to do so. For example, if $R \in \Sigma^{e\to e\to t}$ then $(\Sigma \setminus \{[R]\}) \cup \{[CR]\}$, where CR is the converse of R, also generates the $\eta\beta$ closed term structure of $\mathcal{L}(\Sigma)$. (We saw this phenomena illustrated with the Lindenbaum-Tarski algebra in the previous section.)

There are two attitudes one might have towards this phenomenon. The first is to simply take the notion of fundamentality as *primitive*, and understand the free generation condition

[35] Note that Kilp et al. [19] use the word 'generator' slightly differently.

[36] The constraint $\Sigma^\sigma \neq \emptyset$ ensures there there is always a term that is not a substitution instance of α, namely a constant of the same type.

not as a *definition* of fundamentality in substitution-theoretic terms, but merely a constraint. On this picture, one might have two relations that are converses of one another: from a purely substitution theoretic perspective they are on a par, but given the primitivist conception of fundamentality at most one of them can be fundamental.

The other attitude, which is more in the spirit of the 'substitution-first' methodology, is to reject the monadic conception of fundamentality altogether. On this picture, the relevant primitive is a notion of *cofundamentality*. Roughly $a_1 ... a_n$ are *cofundamental* iff there is a basis, X, which they all belong to.

Definition 20 (Cofundamentality)**.** *Let* (M, A) *be a substitution structure. A subset* $Y \subseteq \bigcup_\sigma A^\sigma$ *is cofundamental iff* $Y \subseteq X$ *for some basis* X*.*

Similarly, we say that an element $a \in A^\sigma$ *is cofundamental with* $b \in A^\tau$ *iff they both belong to some basis.*

Suppose that R is a non-symmetric relation, and $Z \cup \{R\}$ is a basis with $R \notin Z$, so that $Z \cup \{CR\}$ is also a basis. Then any element $z \in Z$ is cofundamental with R, and is also cofundamental with CR. But any collection of elements including both R and CR are never cofundamental with one another.

A basis is a maximal collection of cofundamental elements; it's natural to call such a set a *fundamental basis*. The terminology is appropriated from linear algebra, where a basis for a vector space generates the space, but is not in general unique.[37] Under this analogy a (possibly non-maximal) set of cofundamental elements corresponds to a set of linearly independent vectors.[38]

Let us take the analogy between linear algebra further.

Definition 21. *Let* (M, A) *be a substitution structure. A subset* $X \subseteq \bigcup_\sigma A^\sigma$ *is*

1. Fundamentally independent *iff every type-indexed collection of functions* $f^\sigma : X^\sigma \to A^\sigma$ *extends to at least one substitution* $i \in M$*.*

2. Fundamentally complete *iff every type-indexed collection of functions* $f^\sigma : X^\sigma \to A^\sigma$ *extends to at most one substitution* $i \in M$*.*

It is relatively easy to see that a set is fundamentally independent iff the elements of the set are cofundamental. A subset of a vector space is linearly independent if no element of the set can be defined from the others via linear combinations. This is equivalent to the claim that any function from a linearly independent set to the rest of the space can be extended to at least one linear map. A subset of a vector space *spans* the vector space iff every element of the space can be defined as a linear combination of the elements of the set; or equivalently, every function from a spanning set to the rest of the space extends to at most one linear mapping. The corresponding conditions on substitution structures are similar: fundamental independence captures the idea that the elements of the set can't be defined

[37] Except in the 0-dimensional case.

[38] A set of vectors is linearly independent if no vector in the set can be defined as a linear combination from the remaining elements of the set. This can be shown to be equivalent to being a subset of a basis. To pursue the analogy further, one might hope that a similar equivalence can be proven for cofundamental things: that a set Z is a subset of a fundamental basis iff no element in Z can be metaphysically defined from other elements in Z. This involves showing, among other things, that if Z is 'independent' in this sense — nothing in Z can be defined in terms of the other elements of Z — then it can be extended to a maximal set of independent elements (which corresponds to a basic theorem in linear algebra). It would take us too far afield to explore this further.

from one another (via pure elements), whereas fundamental completeness ensures there are enough 'fundamental' things to define everything (via pure elements). This connection is laid out more explicitly in [ANON][REF].

There are some analogies to be drawn here with the notion of supervenience: just as there may be multiple fundamental bases, there also be multiple supervenience bases for a given class of propositions or properties. However, there are also some crucial differences and the notion of a fundamental basis seems in general to be a more informative notion. For example, our notion of a fundamental basis is a structural notion, not a modal notion. Moreover, not all substitution structures are generated, and so not all substitution structure have fundamental bases; by contrast there is always a supervenience base for a class of propositions or properties (the class itself). Relatedly, every proposition or property belongs to some supervenience base or other. By contrast, many elements do not belong to any fundamental basis (for example, a pure element is never cofundamental with itself unless it is the only thing inhabiting its type).[39]

If (M, A) is a substitution structure with a basis X, then an alternative characterization of metaphysical definability is available in terms of the annihilator congruence. Recall that, for a set Z, we defined $i \sim_Z j$ iff $iz = jz$ for all $z \in Z$.

Lemma 7 (The Fixing Lemma)**.** *If A is an M-set with basis X, and $iy = jy$ for all y in some subset Y of X, then there is a $k \in M$ such k fixes the elements of Y ($ky = y$ for all $y \in Y$), and $i \circ k = j \circ k$.*

Proof. Choose a function $f : X \to A$ which fixes the elements of Y, and maps the elements of $X \setminus Y$ to elements of Y. Since X is a basis there is a (unique) $k \in M$ whose action agrees with f on X. Thus k fixes Y, and since i and j agree on the elements of Y, $ikx = jkx$ for all $x \in X$. Thus $i \circ k = j \circ k$ since X is a basis. □

Proposition 8. *Suppose that (M, A) is a substitution structure with a basis $X \subseteq \bigcup_\sigma A^\sigma$. Let $Z \subseteq X$. Then the following are equivalent:*

- *a is metaphysically definable from Z*
- *For any $i, j \in M$, if $i \sim_Z j$, then $ia = ja$.*

Proof. Suppose that a is metaphysically definable from Z, and suppose that $iz = jz$ for all $z \in Z$. Then, by the fixing lemma, there is a k which fixes Z and is such that $i \circ k = j \circ k$. Since k fixes Z it follows that $ka = a$, and so $ia = i(ka) = j(ka) = ja$) as required.

Conversely, suppose that $i \sim_Z j$ implies that $ia = ja$. Suppose, moreover, that $kz = z$ for all $z \in Z$. Then $k \sim_Z 1$, and so $ka = 1a$, i.e. $ka = a$. Thus any k that fixes Z also fixes a. □

[39] Note, however, that there are also some substantial analogies. In Stalnaker [30] connections between supervenience, definability in infinitary languages, and invariance under certain permutations are drawn. These results suggest that one might expect minimal supervenience bases to line up with fundamental bases in conjugation structures based on Boolean algebras — see especially example 2 (Stalnaker theorizes in terms of isomorphisms between worlds that leave properties in the supervenience base alone, but this is equivalent to our notion of metaphysical definability in terms of permutations if we assume a constant domain modal model theory). However, insofar as the present framework generalizes conjugation structures, the notion of a fundamental basis is more general. A full exploration of the connection between supervenience and fundamental bases will have to wait until future work.

The intuition behind the second characterization of metaphysical definability should be clear: if all the constituents of a appear in a set of fundamental entities Z, and i and j do the same things to Z, then i and j must agree about what they do to a. Less obviously, but still plausibly: if the action of some substitutions on Z is sufficient to determine what they do to a, then a's constituents must be included in Z.

3.5 Fundamental languages

Sometimes metaphysicians will theorize in terms of the notion of a 'fundamental language'. Informally, a language is a fundamental language if every non-logical constant denotes a fundamental proposition, property, relation, etc., and no two constants co-denote. Sometimes it is useful to talk about a complete fundamental language in which every fundamental proposition, relation, etc. is denoted by some constant. Given a substitution structure, (M, A), we have just seen that the behaviour of the substitutions on A, at least partly, encodes the structure of fundamental entities in A.[40] So it is natural to look for a substitution theoretic characterization of what it means for a language $\mathcal{L}(\Sigma)$ to be a fundamental language relative to an interpretation, $[\![\cdot]\!]$, in that structure. This will give us an equivalent characterization of what it means for a structure to be generated, but also will give us a more fine-grained relation between a structure and language making more precise the idea that reality is built out of the fundamental entities via logical (or pure) operations in a way that parallels the way that a fundamental language is built out of its non-logical constants via the logical operations.

If $\mathcal{L}(\Sigma)$ is a language, write $Sub(\mathcal{L}(\Sigma))$ for the monoid of substitutions on $\mathcal{L}(\Sigma)$: substitutions generated by mapping non-logical constants of $\mathcal{L}(\Sigma)$ to arbitrary closed terms of the same type. If $\mathcal{L}(\Sigma)$ is a fundamental language for a substitution structure (M, A), we suppose that every substitution on the language transfers to a substitution on reality: each $i \in Sub(\mathcal{L}(\Sigma))$ determines a substitution of the substitution in M on the applicative structure.

Definition 22 (Fundamental language)**.** *Suppose (M, A) is a substitution structure and $[\![\cdot]\!]$ an interpretation on (M, A). $\mathcal{L}(\Sigma)$ is a fundamental language for $(M, A, [\![\cdot]\!])$ iff*

1. *There is a monoid homomorphism $\rho : Sub(\mathcal{L}(\Sigma)) \to M$.*

2. *$\rho(i)[\![\alpha]\!] = [\![i\alpha]\!]$ for every type σ, and closed term $\alpha \in \mathcal{L}^\sigma(\Sigma)$ and $i \in Sub(\mathcal{L}(\Sigma))$.*

We say that an $\mathcal{L}(\Sigma)$ with these properties is full *for $(M, A, [\![\cdot]\!])$ iff*

1. *ρ is additionally a* surjective *monoid homomorphism.*

2. *$[\![\cdot]\!]$ is surjective: every element of A is denoted by a closed term in $\mathcal{L}(\Sigma)$.*

Finally, say that $\mathcal{L}(\Sigma)$ is a (full) fundamental language for (M, A) iff $\mathcal{L}(\Sigma)$ is a (full) fundamental language for $(M, A, [\![\cdot]\!])$ relative to some interpretation function $[\![\cdot]\!]$.

If $\mathcal{L}(\Sigma)$ is a full fundamental language for (M, A) then there is a derived action of $Sub(\mathcal{L}(\Sigma))$ on each A^σ: the action of $i \in Sub(\mathcal{L}(\Sigma))$ on $a \in A^\sigma$ is defined by $\rho(i)a$. If

[40] For those who take fundamentality to be primitive, the condition that fundamental entities freely generate the structure is to be merely a constraint on fundamentality. Such a person will not accept the idea that the behaviour of substitutions completely determines the fundamental entities, unless there's a unique fundamental basis. But they may still find the following definitions to be of interest.

$i \in Sub(\mathcal{L}(\Sigma))$ and $a \in A^\sigma$, we may thus write ia to unambiguously denote the result of using this action to apply i to a; using this convention condition (2), for example, can be written $i[\![\alpha]\!] = [\![i\alpha]\!]$. Indeed, if (M, A) is a faithful substitution structure, then M is isomorphic to $Sub(\mathcal{L}(\Sigma))/(\sim_{A^e \cup A^t})$ (this is an easy application of the first isomorphism theorem described in Kilp et. al [19]).[41]

If a substitution structure admits a fundamental language that means the structure can be thought of, in a sense, as a sort of coarse-grained language. All the linguistic operations — specifically substitutions — that can be applied to the fundamental language can be also applied to the structure. Thus the structure represents a picture in which reality is structured a bit like a language. Of course, the structure might be more coarse-grained than the language in the sense that linguistically distinct expressions denote the same element of the structure.

I shall leave it open ended which constants of $\mathcal{L}(\Sigma)$ count as non-logical vocabulary. In much of the following we shall treat the connectives and quantifiers as logical, and fixed by every substitution.[42] Since we haven't stipulated that every pure element have an expression in a fundamental language, a substitution structure might contain undenoted pure elements, and so not every fundamental language is full for a give structure and interpretation. The constants in a full fundamental language for an interpretation must denote distinct elements, but distinct complex terms may codenote. For this reason if a language is fundamental and full for a substitution structure it does not in general mean that the substitution structure is isomorphic to a language or a term substitution structure. When $\mathcal{L}(\Sigma)$ is a (full) fundamental language for $(M, A, [\![\cdot]\!])$, $[\![\cdot]\!]$ determines a (surjective) homomorphism from the term substitution structure to A (where a homomorphism between substitution structures on M is a functional M-logical relation: a type preserving mapping that preserves application and the action of M).

(Aside: an interpretation subject to the above constraint is an interpretation that is internal to the category of $Sub(\mathcal{L}(\Sigma))$-sets (the definition of the category of M-sets, for a given M, is given in section 4.2). This is because for a given type σ, $[\![\cdot]\!] : T^\sigma \to A^\sigma$ is an arrow (an equivariant function) from the closed term structure T^σ acted on by $Sub(\mathcal{L}(\Sigma))$ to A^σ with the derived action described earlier. (Indeed, if $[\![\cdot]\!]$ is surjective, then $[\![\cdot]\!]$ is an M-logical relation between substitution structures in the sense of section 2.5. If we set $B^\sigma = [\![T^\sigma]\!] \subseteq A^\sigma$, the image of T^σ under $[\![\cdot]\!]$, then the resulting structure B is closed under $Sub(\mathcal{L}(\Sigma))$-substitutions, and $[\![\cdot]\!]$ is surjective.))

It's worth noting what happens if $\mathcal{L}(\Sigma)$ contains no non-logical constants. Then M is the trivial monoid, and any applicative structure can be trivially endowed with an M-action, making it into a substitution structure (M, A). $\mathcal{L}(\Sigma)$ will automatically be a fundamental language for (M, A). This is because (M, A) contains no fundamental entities in the substitution theoretic sense: all of its elements will be pure — i.e. constituentless entities. More generally, a language can be a fundamental language for many substitution structures that differ over what pure elements there are.

It should be emphasized that it's far from obvious that every substitution structure

[41]By the first isomorphism theorem, $im(\rho) \cong M/ker(\rho)$. Since ρ is surjective, $im(\rho) = M$. Since the action of M on $A^e \cup A^t$ is, by assumption, faithful, $ker(\rho) = \sim_A$: $i\, ker(\rho)\, j$ iff $\rho(i) = \rho(j)$ iff $\rho(i)a = \rho(j)a$ for every $a \in A^e \cup A^e$ iff $i \sim_{A^e \cup A^t} j$ (according to the derived action of $Sub(\mathcal{L}(\Sigma))$ on $A^e \cup A^t$).

[42]Note that I do not require that M be faithful on A, and in general we will have cases where distinct substitutions on $\mathcal{L}(\Sigma)$ correspond to the same substitution on A. For example, if A is a Boolean structure, the the substitution mapping every propositional constant to its double negation, will have the same action on A^t as the identity substitution, even though they are distinct. One may always quotient M by the annihilator congruence $\sim_{A^e \cup A^t}$ to get rid of these redundancies.

should have a full fundamental language: intuitively, something has a full fundamental language if every element of the structure can be built out of pure operations and fundamental things (the denotations of constants). Being a substitution structure with a full fundamental language is thus a constraint on substitution structures.

Here is the most obvious example of a fundamental language for a substitution structure. Let M be the monoid of substitutions on $\mathcal{L}(\Sigma)$ and (M, T) the term substitution structure for $\mathcal{L}(\Sigma)$ under $\eta\beta$ equivalence. Then $\mathcal{L}(\Sigma)$ is the fundamental language for (M, T). One might hope for a less syntactic example.[43] We will look for further examples of fundamental languages in section 5.3 and 5.4.

We shall now see that having a fundamental language is equivalent, with certain side conditions, to having a basis. Firstly we show that if a faithful structure admits a full fundamental language, then the structure has a basis.

Proposition 9. *If (M, A) is a faithful substitution structure that admits a full fundamental language $\mathcal{L}(\Sigma)$ relative to the interpretation $[\![\cdot]\!]$ then (M, A) is generated, and the interpretations of the constants Σ are a basis for (M, A).*

Proof. Let $X^\sigma := \{[\![c]\!] \mid c \in \Sigma^\sigma \text{ a constant of type } \sigma\}$.

Suppose that f is a typed collection of functions, f^σ, that map X^σ to A^σ. Let $k \in Sub(\mathcal{L}(\Sigma))$ be any substitution of $\mathcal{L}(\Sigma)$ such that $[\![\cdot]\!] \circ k = f \circ [\![\cdot]\!]$ (i.e. a function that maps each constant c to any term α such that $[\![\alpha]\!] = f([\![c]\!])$. At least one such term always exists by fullness.)

Since $k \in Sub(\mathcal{L}(\Sigma))$, $\rho(k)$ (as in definition 22) is a substitution i in M. Suppose j is another substitution in M such that $ja = f(a)$ for all $a \in X$ — i.e. $j[\![c]\!] = f([\![c]\!])$ for every constant c. Since ρ is surjective, there is an $k' \in Sub(\mathcal{L}(\Sigma))$ such that $\rho(k') = j$. Note, in particular, that $[\![k'c]\!] = [\![kc]\!]$ for every constant c, since $\rho(k')([\![c]\!]) = f([\![c]\!]) = [\![kc]\!]$ for every constant c. It follows that $[\![k'\alpha]\!] = [\![k\alpha]\!]$ for any term α.

Now since (M, A) is faithful on A, it suffices to show that $ia = ja$ for every a in every A^σ. Since $\mathcal{L}(\Sigma)$ is a full fundamental language for (M, A) relative to $[\![\cdot]\!]$, there is some term α such that $a = [\![\alpha]\!]$. $\rho(k')[\![\alpha]\!] = [\![k'\alpha]\!] = [\![k\alpha]\!] = \rho(k)[\![\alpha]\!]$. Since $i = \rho(k), j = \rho(k')$ and $a = [\![\alpha]\!]$ it follows that $ia = ja$ as required.

□

We can also get two partial converses to theorem 9. Firstly, we can show that every generated substitution structure has a fundamental language.

Proposition 10. *Suppose that (M, A) is a generated substitution structure with a basis X. Then there is a language $\mathcal{L}(\Sigma)$ and interpretation $[\![\cdot]\!]$ such that $\mathcal{L}(\Sigma)$ is a fundamental language for (M, A) relative to $[\![\cdot]\!]$, and $[\![\Sigma]\!] = X$ (i.e. X is the set of interpretations of the constants Σ).*

Proof. Let X be a basis of (M, A), and X^σ the elements of X of type σ. One can construct a higher-order language $\mathcal{L}(\Sigma)$ by letting the set of non-logical constants of type σ be X^σ, and an interpretation function $[\![\cdot]\!]$ defined by mapping each constant to itself.

Every substitution $i \in Sub(\mathcal{L}(\Sigma))$ defines a function $f : X^\sigma \to A^\sigma$ defined by letting $f(c) = [\![ic]\!]$. Since X is a basis of (M, A), this function extends to a unique substitution in M, which we shall call $\rho(i)$, such that $\rho(i)c = f(c) = [\![ic]\!]$ for all $c \in X$. Thus we have

[43] By theorem 20, which we encounter later, we know that term structures are always isomorphic to certain M-set structures, which are a class of non-syntactically defined substitution structure.

shown that $\rho(i)[\![c]\!] = [\![ic]\!]$ for each constant. It is easily seen that this extends to arbitrary terms: $\rho(i)[\![\alpha]\!] = [\![i\alpha]\!]$.

It remains to show that ρ is indeed a monoid homomorphism: i.e. that $\rho(1) = 1$ (trivial), and that $\rho(i \circ j) = \rho(i) \circ \rho(j)$. For the latter, note $\rho(i \circ j)$ is the unique substitition in M that extends the function $f(c) = [\![ijc]\!]$, $f : X \to A$. We have already argued that for every $c \in X$, $\rho(j)c = [\![jc]\!]$, and also that $\rho(i)[\![jc]\!] = [\![ijc]\!]$. So $\rho(i \circ j)c = \rho(i) \circ \rho(j)c$ for every $c \in X$. Since X is a basis it follows from the uniqueness property that $\rho(i \circ j) = \rho(i) \circ \rho(j)$. □

Summarizing these two theorems we have:

1. Every faithful substitution structure with a full fundamental language has a basis.

2. Every substitution structure with a basis has a fundamental language.

One might also hope for a full converse to proposition 9, telling us that every faithful substitution structure with a basis has a *full* fundamental language. It seems unlikely that such a proposition could be proved. However, when (M, A) has a finite basis X, and is moreover a special sort of substitution structure — a full M-set structure — then we can prove that (M, A) has a full fundamental language. We shall not meet the notion of a full M-set structure until section 4.1. However, the following proof relies only on the following elementary fact about them: an element is definable by a pure element and elements of a basis if it is metaphysically definable from those basis elements. We shall prove this fact in section 4.3 — see theorem 18 (the stipulation that X be finite below is so that theorem 18 can be applied).

Proposition 11. *Suppose that (M, A) is a finitely generated full M-set structure. Then there is a language $\mathcal{L}(\Sigma)$ and interpretation $[\![\cdot]\!]$ such that $\mathcal{L}(\Sigma)$ is a full fundamental language for (M, A) relative to $[\![\cdot]\!]$.*

Proof. As in the last argument, let $X = \{c_1, ..., c_n\}$ be a finite basis of (M, A). Let S be the set of pure elements in A and S^σ the pure elements of type σ. One can construct a higher-order language $\mathcal{L}(\Sigma)$ by letting the non-logical constants of type σ be X^σ, the logical constants of type σ as S^σ, and an interpretation function $[\![\cdot]\!]$ defined by mapping each constant to itself.

ρ is defined as in the proof of proposition 10. It remains to show ρ and $[\![\cdot]\!]$ are surjective. Firstly we show that every element $a \in A$ is the denotation of some term. It is sufficient (and indeed necessary) to show that there is a pure element q such that $q(c_1)...(c_n) = a$. Moreover, by theorem 18, it suffices to show that a is metaphysically definable from $c_1...c_n$. But this is trivial: if $ic_m = c_m$ for each m, then $i = 1$, since $c_1...c_n$ form a basis, and so there can be only one substitution that agrees with 1 on $c_1...c_n$, namely 1 itself. Thus $ia = 1a = a$, and a is metaphysically definable from $c_1...c_n$.

Now for the surjectivity of ρ. Suppose that $i \in M$. For each $c \in X^\sigma$, $ic = [\![\alpha_{i,c}]\!]$ for some term $\alpha_{i,c}$, by our previous argument. So let $kc = \alpha_{i,c}$ for each $c \in X$. This defines a substitution on $\mathcal{L}(\Sigma)$, and moreover $\rho(k)c = [\![ic]\!]$ for each c. □

4 M-Set Structures

In this section a class of concrete quasi-functional substitution structures is investigated. It is shown that every quasi-functional substitution structure is isomorphic to a member of this class. The concreteness assists in getting a better picture of how quasi-functional substitution structures behave, and we investigate the interpretation of λ-terms in these structures in detail. They also bring to salience the notion of a quasi-functional substitution structure being *full* in a sense that's analogous to the fullness of a Henkin model; it is shown that a term structure is a full M-set structure when M is the monoid of substitutions of the relevant language.

4.1 M-sets and function spaces

In a substitution structure every type gets associated with a set, by the applicative structure, and is equipped with an action of the monoid on that set. There is an approach to type theory in which types are not associated with a set, but with a more general notion of a object from a category. Since it is natural to want to identify the semantic values of types themselves with sets equipped with an action (M-sets) it's natural to consider the category whose objects are M-sets and whose arrows are equivariant functions between M-sets. Luckily this category has some nice properties: in particular, it is cartesian closed, meaning that it determines, for any choice of objects for base types, an applicative structure with combinators. Unlike our previous examples, these constructions allow for a non-syntactic construction of the functional domains without the restriction that M is group.

The M-set formalism in a natural sense generalizes the conjugation structures we considered in example 2 to allow for substitutions that aren't necessarily bijective. The fact that we can't simply lift a permutation from lower domains to higher domains by conjugation (since we don't have inverses) means that the function space between A and B has to have more structure than merely a collection of functions from A to B. It needs extra information to encode how a substitution lifts from A and B to an element of the function space: indeed, we shall identify function spaces with certain well-behaved functions from $M \times A \to B$. Since elements of the function space have more structure than mere functions between the argument types, the resulting applicative structure is not functional. However, it is quasi-functional in the sense of definition 7.

Indeed, these M-set structures give rise to a representation theorem for quasi-functional structures (theorem 15): every quasi-functional substitution structure is isomorphic to a structure where the function space is given by a collection of equivariant functions $M \times A \to B$.

Finally, it is worth noting that the category of M-sets is part of a wider class of models of type theory that generalize the ordinary set-theoretic models. These are functor categories whose objects are functors from a category C to sets, and whose arrows are natural transformations; we write this Set^C. In this case, they are functor categories Set^M where M is category with a single object (i.e. a monoid).

Given two M-sets A and B we can form the product of A and B, $A \times B$, which simply consists of the cartesian product of A and B, with the action defined by setting $i(a, b)$ to (ia, ib). We can similarly form a function space B^A as follows:

Definition 23 (Function space for M-sets)**.** *If A and B are M-sets, the* full M-set function space *B^A is defined as follows*

- *The underlying set is $B^A := \{f : M \times A \to B \mid f(i \circ j, ix) = i(f(j, x))$ for all $x \in A$ and $i, j \in M\}$*
- *The action on this set is: $if := (j, x) \mapsto f(j \circ i, x)$ for each $f : M \times A \to B$.*

The condition for f to belong to B^A is just that f be an equivariant function in $M \times A \to B$, where we consider M as an M-set itself with the action $\mu : M \times M \to M$ of precomposition: $\mu(i, j) = j \circ i$. So understood, $M \times A$ is just the product of two M-sets with the componentwise action $i(a, b) := (ia, ib)$. [44]

To get an intuition for our definition of the function space, suppose that f interprets some property F, which possibly has constituents. $f(i, a)$ may then be thought of as the result of taking an i substitution of F and applying it to a: take all of the constituents of F and substitute them with their i-counterparts in F, then apply the result to a. By this reasoning, then, we can see that $f(1, a)$ corresponds to Fa. More generally, $f(i, ia)$ corresponds to $i(Fa)$, since substituting the i-counterparts of constituents of F for those constituents, and then applying the result to $i(a)$ is the same as substituting the i-counterparts of constituents of Fa for those constituents. Putting these two things together gives us that $f(i, ia) = if(1, a)$.

Indeed we can generalize even further: according to our interpretation we should have $f(i \circ j, ia) = i(f(j, a))$. To calculate the LHS you replace all of the simple constituents in F with their $i \circ j$-counterparts, and apply the result to ia. To calculate the RHS you replace all the constituents in F with their j-counterparts, apply to a, then replace everything in the result with its i-counterpart. It should be clear why these two things are identical, and that the condition of membership in B^A is well-motivated.

The action of M on B^A tells us what the i-counterpart of each function $f : M \times A \to B$ is. As before we write if for the action of i on f. If f corresponds to a property F then if intuitively corresponds to the result of replacing all the constituents in F with their i-counterparts. Recall that $f(j, a)$, on the above interpretation, is the result of substituting the constituents of F with their j-counterparts and applying the result to a. So by that reasoning, $(if)(j, a)$ is the the result of substituting the j-counterparts in [the result of substituting for i-counterparts in F], and then applying to a. This is, of course, exactly the same as $f(j \circ i, a)$; so our definition of if makes sense.

We now outline a natural class of quasi-functional substitution structures based on the above observations and definitions. As with Henkin structures, we allow for non-full interpretations of functional types $\sigma \to \tau$: cases where $A^{\sigma\to\tau}$ contain some but not all of the equivariant functions in B^A. However, because we want the interpretation of $A^{\sigma\to\tau}$ to be an M-set, we have to make sure we only allow subsets of B^A that are closed under taking substitutions. That is, $A^{\sigma\to\tau}$ must not only be a subset of B^A, but a *subact*.

Definition 24 (M-set structures)**.** *An M-set structure based on A^e and A^t is a pair (M, A) where M is a monoid, and A is an applicative structure such that:*

1. *A^e and A^t are M-sets.*
2. *$A^{\sigma\to\tau}$ is subact of ${A^\tau}^{A^\sigma}$. In other words:*
    - *$A^{\sigma\to\tau}$ is a subset of the full M-set function space ${A^\tau}^{A^\sigma}$*
    - *If $f \in A^{\sigma\to\tau}$ then $if \in A^{\sigma\to\tau}$ for any $i \in M$*
3. *$App^{\sigma\tau}(f, a) = f(1, a)$ where $f \in A^{\sigma\to\tau}$ and $a \in A^\sigma$*

An M-set structure based on A^e and A^t is full *iff $A^{\sigma\to\tau}$ is the full M-set function space ${A^\tau}^{A^\sigma}$ for each σ, τ.*

[44] B^A, so defined, satisfies the categorical conditions for being an exponential object.

Remark 5. *Given M-sets B and B' there is a unique full M-set structure such that $A^t = B$ and $A^e = B'$. We call this the full M-set structure based on B and B'.*

It is straightforward to show that every M-set structure is a substitution structure:

Proposition 12. *If (M, A) is an M-set structure, (M, A) is a substitution structure with Sub^σ given by the supplied actions.*

Proof. It suffices to show that $iApp(f, a) = App(if, ia)$. Simply by applying definitions we have: $App(if, ia) = (if)(1, ia) = f(1 \circ i, ia) = f(i \circ 1, ia)$. Finally, by the equivariance condition, the final term is just $i(f(1, a)) = iApp(f, a)$ as required. □

Let us now turn to functionality. Since in M-set structures, $App(f, a) = f(1, a)$, it follows that M-set structures are not obviously guaranteed to be functional. Indeed, it's possible that $f(1, a) = g(1, a)$ for every a, but that f and g are distinct because $f(j, a) \neq g(j, a)$ for some a and $j \in M$. Recall that a substitution structure is *quasi-functional* when $f = g$ if and only if $App(if, a) = App(ig, a)$ for every $a \in A^\sigma$, $i \in M$.

Proposition 13. *M-set structures are quasi-functional*

The reason is clear: $App(if, a) = (if)(1, a) = f(1 \circ i, a) = f(i, a)$. Thus $f(i, a) = g(i, a)$ for all i and a, clearly implies that $f = g$.

Moreover, when M is a group, then M-set structures are functional (as proved in section 2.1 for arbitrary quasi-functional substitution structures), so it is possible to simply treat $A^{\sigma\to\tau}$ as a set of functions from A^σ to A^τ, rather than from $M \times A^\sigma$ to A^τ. Indeed, when M is a group, M-set structures are just conjugation structures, and full M-set structures are full conjugation structures. This justifies our remark that M-set structures generalize conjugation structures to arbitrary monoids:

Proposition 14. *Suppose M is a group and (M, A) a (full) M-set structure. Then (M, A) is isomorphic to a (full) conjugation structure.*

Proof. Proof sketch: we noted above that M-set structures are functional when M is a group. Proposition 1, recall, tells us that every functional substitution structure based on a group is a congjugation structure. Thus every M-set structure based on a group is a conjugation structure. Now suppose that (M, A) is additionally a full M-set structure. To show that it is full simpliciter, let g be any function from $A^\sigma \to A^\tau$. We want to find an $f : M \times A^\sigma \to A^\tau$ that is equivariant and has the applicative behaviour of g. This is achieved by setting $f(i, a) = i(g(i^{-1}a))$. (Since $App(f, a) = f(1, a) = g(a)$.) □

Finally, we show that all quasi-functional substitution structures are isomorphic to M-set structures. Thus the identification of the function space with with certain functions in $M \times A \to B$ is no accident.

Theorem 15 (Representation theorem for quasi-functional substitution structures)**.** *Every quasi-functional substitution structure is isomorphic to an M-set structure.*

Proof. Let A be a quasi-functional substitution structure. We build an M-set structure B and an isomorphism between them simultaneously, by induction. (Recall that an isomorphism is a bijective M-logical relation, which I'll write $R^\sigma(a) = b$.)

Let $B^\sigma = A^\sigma$ and R^σ be the identity function, for base types.

Now suppose, for induction that B^σ and B^τ are defined and that $R^\sigma : A^\sigma \to B^\sigma$ and $R^\tau : A^\tau \to B^\tau$ are defined and bijections. For each $f \in A^{\sigma\to\tau}$ we define a function $R^{\sigma\to\tau}(f)$:

- $R^{\sigma\to\tau}(f)(i, R^\sigma(a)) = R^\tau(App(if, a))$ for all $a \in A^\sigma$ and $i \in M$

Since by inductive hypothesis R^σ is surjective, this specifies the value of $R^{\sigma\to\tau}(f)(i, b)$ for all $b \in B^\sigma$. We let $B^{\sigma\to\tau}$ be the set of all such functions $R^{\sigma\to\tau}(f)$ for $f \in A^{\sigma\to\tau}$. By this stipulation we ensure that $R^{\sigma\to\tau} : A^{\sigma\to\tau} \to B^{\sigma\to\tau}$ is surjective. It is also injective: if $R^{\sigma\to\tau}(f) = R^{\sigma\to\tau}(g)$ then, by for any i and $a \in A^\sigma$, $R^\tau(App(if, a) = R^\tau(App(ig, a))$. Since, by inductive hypothesis, R^τ is injective, this means $App(if, a) = App(ig, a)$ for all i and $a \in A^\sigma$, so $f = g$ by quasi-functionality.

It remains to show $iR^{\sigma\to\tau}(f) = R^{\sigma\to\tau}(if)$ (i.e. the condition Rab implies $Riaib$). This is straightforward. □

4.2 The category of M-sets

The M-sets form a category where the objects are M-sets and the arrows are equivariant maps between M-sets. Indeed it is cartesian closed, with the exponential objects as defined in the previous section. This means that, even though our applicative structure is not functional, it nonetheless satisfies all the equations of the λ-calculus (see Mitchell [23] chapter 7 section 2). Those not familiar with the relevant category theory may skip to the end of the section, where a direct interpretation of λ-terms is given.

Definition 25 (Arrow between M-Sets)**.** *We say that f is an arrow between M-sets A and B, written, $f : A \to B$, iff f in an equivariant map between A and B. That is:*

- *f preserves the monoid action: $f(ix) = if(x)$ for all $i \in M$.*[45]

This category is in fact cartesian closed, as witnessed by the fact that there is a natural isomorphism (which we denote λ) between the arrows from $A \times B$ to C and the arrows from A to C^B. It is defined as follows:

Definition 26. *Given any arrow $f : A \times B \to C$, we define $\lambda f : A \to C^B$ as follows:*

- *$\lambda f(a)(j, b) = f(ja, b)$ for every $a \in A$ and $b \in B$.*

We can prove that λf preserves the monoid action as follows:

$$\lambda f(ix)(j, y) = f(j(ix), y)$$

$$i(\lambda f(y))(j, y) = \lambda f(a)(j \circ i, b) = f(j \circ i(a), b).$$

Applying the inverse of λ to the identity arrow $C^B \to C^B$ gives us an arrow $(C^B \times B) \to C$ which defines application:

Definition 27 (Application)**.** *The application arrow, $app : B^A \times A \to B$, can be defined as follows. Suppose that $a \in A$, and $f \in B^A$, i.e. $f : M \times A \to B$.*

- *$app(f, a) = f(1, a)$*

here $1 \in M$ is the identity mapping.

[45] Spelling that out fully: $f(\mu_A(i, x)) = \mu_B(i, f(x))$, where μ_A and μ_B are the actions of A and B respectively.

app may be shown to preserve the action as follows. $app(i(f,a)) = app((if, ia)) = if(1, ia) = f(i, ia)$, $i(app(f,a)) = i(f(1,a)) = f(i, ia)$, where the last identity holds by the fact that f preserves the action (note also that this is exactly the condition we needed to show that *app* satisfies condition for being a substitution structure).

It follows by standard results that any M-set applicative structure interprets the λ-calculus (see Mitchell [23]). It is also instructive to see how to construct interpretations of λ-terms explicitly.

- $[\![c]\!]^g \in A^\sigma$ for each constant $c \in \Sigma^\sigma$
- $[\![x]\!]^g = g(x) \in A^\sigma$ for each variable $x \in Var^\sigma$
- $[\![\alpha\beta]\!]^g = [\![\alpha]\!]^g(1, [\![\beta]\!]^g)$
- $[\![\lambda x\alpha]\!] = (i,a) \mapsto [\![\alpha]\!]_i^{(i\circ g)[x\mapsto a]}$

Here $[\![\cdot]\!]_i$ is the interpretation function generated by setting $[\![c]\!]_i = i[\![c]\!]$.

Let me end by offering something by way of an explanation for where the definition of the full M-set function space came from — an argument that this definition is in some sense inevitable. Throughout this argument we write $Hom(A,B)$ for the set of equivariant functions between two M-sets A and B

We'll begin with a general representation theorem for M-sets: they can always be represented by certain functions from M to A. Consider M as an M-set with the action given by left composition, and let A be an arbitrary M-set. Then $Hom(M,A)$ can itself be thought of as an M-set, where $if := f(\cdot \circ i)$. We then have a representation theorem for M-sets:[46]

Proposition 16 (Yoneda for M-sets)**.** *For any M-set A, A is isomorphic to $Hom(M,A)$.*

Now suppose that the category of M-sets contained an exponential object B^A, satisfying the following natural isomorphism, which uniquely characterizes the exponential:[47]

- $Hom(X \times A, B) \cong Hom(X, B^A)$

In particular we would have to have: $Hom(M \times B, C) \cong Hom(M, C^B)$, but by our special case of the Yoneda lemma, $Hom(M, C^B) \cong C^B$. Thus C^B has to be isomorphic to the set of equivariant functions from $M \times B$ to C.

[46] It is a special case of the Yoneda lemma. Consider the category with one object $*$ and arrows $i : * \to *$ for each $i \in M$, with composition of arrows defined by composition in the monoid. M itself may be identified with the functor $Hom(-,*)$ that maps the only object, $*$, to the set M (since by stipulation $Hom(*,*) = M$), and maps each arrow i to a function f_i on M, defined by $f_i(j) = j \circ i$. An M-set A may similarly be understood as a functor, which maps $*$ to a set, $A(*)$, and takes each arrow on $*$, $i \in M$, to a function on $A(*)$, which we think of as the action of i on $A(*)$. Under these identifications it may be easily checked that the equivariant functions from M (as on M-set) to A correspond to the natural transformations between $Hom(-,*)$ and the functor A: $Nat(Hom(-,*), A)$. The Yoneda lemma states that $Nat(Hom(-,*), A) \cong A(*)$ naturally in $*$. ($A(*) \cong Hom(y(*), A)$ alone means there is a bijection between the left and right-hand side. The naturality ensures that the bijection commutes with every arrow from $*$ to $*$ — that is, it ensured the bijection is indeed a homomorphism of M-sets.) For further information see Leinster [21].

[47] To see why this isomorphism characterizes exponentiation, suppose F and G are two endofunctors that are candidates for being exponentiation by A: i.e. they satisfy the equation $Hom(X, FB) \cong Hom(X \times A, B) \cong Hom(X, GB)$ naturally in X and B. Thus the hom functors are isomorphic, $Hom(-, FB) \cong Hom(-, GB)$, so by the Yoneda lemma, $FB \cong GB$, naturally in B.

4.3 Decomposition principles

Given some simple entities, $a_1...a_n$ (of types $\sigma_1...\sigma_n$), let us say that c of type τ is *constructable* from $a_1...a_n$ iff it is possible to get c by combining only elements of $a_1...a_n$ and pure elements by application. Assuming we are working in a substitutions structure with combinators (such as a full M-set structure), and since combinators are always pure, it follows that c is constructable from $a_1...a_n$ iff there is a single pure element, q, of type $\sigma_1 \to (\sigma_2 \to ...\tau)...)$ such that $q(a_1)...(a_n) = c$.[48] If we have product types, this is also equivalent to there being an uncurried pure element q of type $(\sigma_1 \times ... \times \sigma_n) \to \tau$ such that $q(a_1, ..., a_n) = c$. Since we will be working with M-set structures, and since uncurried pure elements are easier to deal with in that setting (see the definition of currying in definition 26), we shall primarily use the second formulation.

Definition 28 (Constructability)**.** *Let (M, A) be a substitution structure with combinators and products, and let $c \in A^\tau$, and $a_1 \in A^{\sigma_1}...a_n \in A^{\sigma_n}$. c is* constructable *from $a_1...a_n$ iff there is a pure element $q \in A^{(\sigma_1 \times ... \times \sigma_n) \to \tau}$ such that $App(q, (a_1, ..., a_n)) = c$*

In this section we shall investigate the connection between metaphysical definability and constructability. In particular, we see that in M-set structures that have a basis, constructability and metaphysical definability always coincide.

First we begin by showing that if something can be constructed out of fundamental entities (relative to a given fundamental basis) using a pure element, then that pure element is unique. Before proceeding, I remind the reader that in an M-set structure $A^{\sigma\to\tau} \subseteq M \times A^\sigma \to A^\tau$ and $App(f, a) = f(1, a)$

Lemma 17. *Suppose that (M, A) is an M-set structure with a basis X, that $a_1...a_n$ are distinct elements of X and that $c \in A^\tau$. If there is a pure element $q \in A^{\sigma_1 \times ... \times \sigma_n \to \tau}$ such that $App(q, (a_1, ..., a_n)) = q(1, (a_1, ..., a_n)) = c$ then it is unique.*

Proof. Let $b_1...b_n$ be any tuple from $A^{\sigma_1 \times ... \times \sigma_n}$, and $j \in M$, and let i be such that $ia_m = b_m$ for $m = 1...n$ (as guaranteed by the fact that $a_1...a_n$ are members of a basis and are pairwise distinct). Then suppose that q and q' are pure elements with $q(1, (a_1...a_n)) = c = q'(1, (a_1...a_n))$.

- $q'(j, (b_1, ..., b_n)) = q'(i, (b_1, ..., b_n))$ since q' is pure.
- $= q'(i, (ia_1, ..., ia_n))$ by construction of i.
- $= iq'(1, (a_1...a_n))$ since q' is equivariant.
- $= ic$ since $q'(1, (a_1...a_n)) = c$

By parallel reasoning $q(j, (b_1...b_n))$ is also identical to ic. So $q(j, (b_1...b_n)) = q'(j, (b_1...b_n))$. Since j and $b_1...b_n$ were arbitrary, q and q' are identical. □

This result has some surprising consequences. For instance, elements of our fundamental basis cannot ever stand in an interesting structural relation to themselves. For every basis element is the result of applying the identity function to itself, and so cannot also be the result of appyling another pure element to itself. One application of this is that there cannot be any symmetric relations in a fundamental basis.[49] For if R is equal to its converse,

[48] Recall that $q(a_1)...(a_n)$ is short for $App(App(...App(q, a_1), a_2)...a_n)$.
[49] See Goodman [17] for a similar argument.

$R = CR$, then the converse combinator C must be identity, since C is also pure. But this is a reductio, since C is not identity.[50]

In M-set structures we may think of pure elements as ways of logically constructing things out of simple entities. It is natural, then, to ask how this notion of construction relates to our notion of metaphysical definability. It can be shown that they in fact coincide in full M-set structures with bases: if $a_1...a_n$ belong to a basis, then c is metaphysically definable from $a_1...a_n$ iff there is a pure element mapping $a_1...a_n$ to c.

Theorem 18. *Suppose that (M, A) is a full M-set structure with a basis X. Then whenever $c \in A^\sigma$ is metaphysically definable from distinct $a_1...a_n \subseteq X$, there is a unique pure element $q : \sigma_1 \times ... \times \sigma_n \to \tau$ such that $q(1, (a_1, ..., a_n)) = c$*

Proof. For any $b_1...b_n$, we shall define $q(j, (b_1, ..., b_n))$ as follows. Pick any i such that $ia_m = b_m$ for each m (such an i exists since $a_1..a_n$ belong to the basis X). Then set $q(j, (b_1, ..., b_n)) = ic$ (since (M, A) is full such a q will exist). Since, by proposition 8, any k that agrees with i on $a_1...a_n$, agrees with i on c, this definition does not depend on the rest of our choice of i. $q(k \circ j, (kb_1, ..., kb_n)) = q(k \circ j, (kia_1...kia_n)) = kic = kq(j, (b_1...b_n))$, so q is equivariant. q is independent of j, so q is pure, and $q(1, (a_1...a_n)) = c$.

Finally, q is unique by lemma 17. □

Corollary 19. *Suppose $a_1...a_n \in X$. Then c is metaphysically definable from $a_1...a_n$ if and only if there is some pure element, q, such that $App(q, (a_1...a_n)) = c$.*

Proof. It remains to show the right-to-left direction. If $q(1, (a_1...a_n)) = c$ and $ia_m = a_m$ for $m = 1...n$, then $ic = iq(1, (a_1...a_n)) = q(i, (ia_1...ia_n)) = q(1, (a_1...a_n)) = c$. So c is metaphysically definable from $a_1...a_n$. □

4.4 The term model as a full M-set structure

Recall the term model T from example 5. A *surjective substitution* on $\mathcal{L}(\Sigma)$ is a substitution that is surjective onto $\mathcal{L}^\sigma(\Sigma)$ for each σ. If there are infinitely many constants of each type, every substitution instance of a particular term α can also be achieved by applying a surjective substitution to α. Thus there is a sense in which no generality is lost by restricting to surjective substitutions.

Example 16. *Let M be the monoid of all surjective substitutions of $\mathcal{L}(\Sigma)$. Let $A^t = T^t$ and $A^e = T^e$ be equivalence classes of closed $\eta\beta$ equivalent terms in some language $\mathcal{L}(\Sigma)$ that contains infinitely many constants of all types, considered as M-sets with the evident action.*

Let (M, A) be the full M-set structure generated by A^t and A^e. Then (M, A) is a substitution structure with a basis given by the interpretations of the constants in Σ.

Indeed, this example is quite special: we can actually show that this structure is equivalent to the term substitution structure T, where elements of T^σ are classes of $\eta\beta$ equivalent terms for arbitrary type σ (where we have again restricted attention to surjective substitutions).

This is quite striking, as we know by contrast that the full set-theoretic structure based on T^t and T^e is not the term model, and indeed, is uncountable at every type except for the

[50]At least, C is not identity given the plausible assumption that there is at least one non-symmetric relation.

base types. This means the constraint on the full function space for M-sets is actually quite significant. (Indeed, this result reflects a famous theorem of Friedman's but for full M-set structures (Friedman [14]): Friedman's result states that there is a partial homomorphism from the full set-theoretic hierarchy to the term model, whereas we get a strict isomorphism.)

Theorem 20. *Suppose that M is the monoid of surjective substitutions of $\mathcal{L}(\Sigma)$, and (M, T) is the closed term substitution structure (i.e. T^σ consists of equivalence classes of closed $\eta\beta$ equivalent σ terms). Then there is a full M-set structure (M, A) and an isomorphism of substitution structures $\rho : T \to A$.*

An isomorphism ρ is a typed collection of bijections $\rho^\sigma : T^\sigma \to A^\sigma$ that commutes with application and commutes with the monoid action: i.e. $\rho(App(f, a) = App(\rho(f), \rho(a))$ and $\rho(ia) = i\rho(a)$. An isomorphism is thus clearly an M-logical relation that is a bijective function on each type.

To show this we must first prove a slightly fiddly lemma

Lemma 21. *Suppose that for each constant a of type σ, there is a term ϕ^a of type τ in $\beta\eta$ normal form. Suppose moreover, that it is subject to the constraint that if $ja = jb$ then $[j\phi^a] = [j\phi^b]$ for any constants a and b of the same type. Then there exists some term $\alpha : \sigma \to \tau$ such that $[\phi^a] = [\alpha a]$ for each a.*

Proof. Pick some constant a arbitrarily, and let b and d be distinct constants that don't appear in ϕ^a. Let us write $x, y \mapsto z$ for the substitution that substitutes z for both x and y.

Firstly we prove that for any constant $c \neq b$, ϕ^c does not contain b. Let $k = a, c \mapsto d$. Then $k\phi^a =_{\eta\beta} k\phi^c$ since $ka = kc = d$. Indeed $k\phi^a = k\phi^c$, since ϕ^a and ϕ^b are in normal form, and k substitutes constants, so that $k\phi^a$ and $k\phi^c$ are in normal form too. $k\phi^a$ does not contain b, since ϕ^a does not contain b, and k introduces only d. So $k\phi^c$ does not contain b, and thus ϕ^c does not contain b since k leaves b alone.

Now ϕ^b can be decomposed into αb where α does not contain b (let $\alpha = \lambda x \phi^b[x/b]$). Let c be any constant, and let $i = b, c \mapsto c$. Since $ib = ic$, we know that $i\phi^c =_{\eta\beta} i\phi^b =_{\eta\beta} i(\alpha b) =_{\eta\beta} i(\alpha)ib = \alpha ib = \alpha c$. Secondly, $i\phi^c = \phi^c$ since, as we showed above, b does not occur in ϕ^c. Thus $\phi^c = \alpha c$ for arbitrary c.

□

The proof of theorem 20 can now proceed

Proof. This is proved by constructing ρ as an M-logical relation. It suffices to show that ρ is functional (i.e. single-valued) and surjective, and that ρ's converse is as well.

Let A be the full M-set structure generated by letting $A^t = T^t$ and $A^e = T^e$ (thus $A^{\sigma\to\tau}$ is given by M-set exponentiation). Let ρ be the M-logical relation between T and A generated by the identity function on the base types. Clearly ρ is a bijection on base types; so suppose, for induction, that it is a bijection for type σ and τ. We may thus use the function notation $\rho^\sigma([\alpha]) = a$ instead of $\rho^\sigma[\alpha]a$, and similarly for ρ^τ.

Let us show that ρ is a total function on type $\sigma \to \tau$. That is, for any $\alpha \in \mathcal{L}^{\sigma\to\tau}(\Sigma)$ we want an $f : M \times A^\sigma \to A^\tau$ such that $\rho[\alpha]f$. By the definition of a M-logical relation this means that for any $[\beta]$ in T^σ, $\rho([\beta]) = b$ implies $\rho([i\alpha][\beta]) = f(i, b)$. We simply fix $f(i, \rho([\beta])) = \rho([(i\alpha)\beta])$. Since ρ is surjective at type σ this fixes the value of $f(i, b)$ for all i and b. ρ must moreover be functional on type $\sigma \to \tau$, since $\rho[\alpha]g$ implies that $g(i, \rho([\beta])) = \rho([i\alpha][\beta]) = f(i, [\beta])$ for all β and i, thus fixing $f(i, b) = g(i, b)$ for all i and b.

Showing that the converse of ρ, ρ^c, is total and functional is slightly harder. Given $f \in A^{\sigma\to\tau}$, we want $[\alpha]$ such that $\rho^c f[\alpha]$.

By the inductive hypothesis ρ^c is a total function on types σ and τ. Moreover ρ^c commutes with every $i \in M$ at types σ and τ: since ρ is an M-logical relation, $\rho^c(a) = b$ implies $\rho^c(ia) = i\rho^c(a) = ib$ for any $i \in M$.

Set ϕ^b as the $\eta\beta$ normal form element of $\rho^c(f(1, \rho(b)))$. If $ja = jb$ then $j\phi^b = j\rho^c(f(1, b)) = \rho^c(j(f(1, b))) = \rho^c(f(j, jb)) = \rho^c(f(j, ja)) = \rho^c(jf(1, a)) = j\rho^c(f(1, a)) = j\phi^a$. So by lemma 21 it follows that there is an α such that $\rho^c(f(1, a)) = \phi^a = [\alpha a]$ for all a. Since ρ^c is an isomorphism on type σ we know that the action of $i \in M$ is surjective on A^σ, since by stipulation it is surjective on T^σ.

Now suppose that $\rho^c(a) = [b]$. It suffices to show that $\rho^c(f(j, a)) = [(j\alpha)b]$ for any j. Since j is surjective on A^σ, pick c such that $jc = a$. Now $\rho^c(f(j, a)) = \rho^c(f(j, jc)) = \rho^c(jf(1, c)) = j\rho^c(f(1, c)) = j[\alpha]\rho^c(c) = [j\alpha]j\rho^c(c) = [j\alpha]\rho^c(jc) = [j\alpha]\rho^c(a) = [(j\alpha)b]$. Thus $\rho^c(f) = [\alpha]$. It follows that ρ^c is surjective at type $\sigma \to \tau$.

ρ^c is also functional. Note that if $\rho^c f[\beta]$ then, by the definition of an M-logical relation, $[(j\beta)a] = [(j\alpha)a]$ for any constant a and substitution j. In particular $[\beta a] = [\alpha a]$. But $\beta a = \alpha a$ is provable in the pure $\eta\beta$ theory only if $\beta x = \alpha x$ is also provable (substitution x for a in the first proof), and so by the ξ and η rules, one can prove $\beta = \alpha$, so $[\beta] = [\alpha]$.[51] □

Note that this shows, as with the Friedman proof, that the M-set structures have the 'least model completeness' property with respect to $\eta\beta$ equivalence: there is some M-set structure such that any pair of $\eta\beta$-inequivalent terms have distinct denotations in it. But it also shows the slightly more striking result that the term model is itself a full M-set structure.

5 Metaphysical Theorizing

In this section we indicate how to go about expressing metaphysical theories in a higher-order language that is interpreted by a substitution structure. Some difficulties for the approach are discussed, and some object language principles of interest are outlined as well as a simple model of them.

5.1 Expressing substitution-theoretic concepts in the object language

Our approach so far has been thoroughly model-theoretic. It is natural to wonder to what extent we can capture the notions we have defined model theoretically in an object language. As we shall see at the end of the section, one needs to be careful: certain notions, like that of a substitution, can lead to paradoxes when formalized in the object language.

The natural setting for this sort of project is *higher-order logic*. A higher-order language is a language $\mathcal{L}(\Sigma)$ in which Σ contains, in addition to non-logical constants of various types, a collection of logical constants that includes at least a quantifier $\forall_\sigma$, of type $(\sigma \to t) \to t$, identity $=_\sigma$ of type $(\sigma \times \sigma) \to t$, and a constant $c_f : (t \times ... \times t) \to t$ for each n-ary truth functional connective f. We shall adopt the usual convention of writing $\forall_\sigma x\phi$ as short for $\forall_\sigma \lambda x\, \phi$.

Making further assumptions about the structure of propositions allows us to take a smaller set of truth functional connectives as primitive. For example, if the propositions

[51] The ξ rule says that if you can prove $\alpha = \beta$ then you can prove $\lambda x\alpha = \lambda x\beta$. This rule is admissible in the theory of $\eta\beta$ equivalence — see Mitchell [23] chapter 4.

in type t form a Boolean algebra, every truth functional connective in an M-set structure is definable from the Boolean operations $\to$ and $\bot$. Moreover, in the Boolean models considered below, $\bot$ is definable as $\forall_{t\to t}\forall_t$, and identity $=_\sigma$ as $\lambda xy\forall_{\sigma\to t}X(Xx \to Xy)$.[52] Since we will be working with such Boolean models in what follows, we shall simply take the quantifiers and conditional as primitive.

Definition 29 (Logical model)**.** *A logical model for a higher-order language $\mathcal{L}(\Sigma)$ is a triple $(A, v, [\![\cdot]\!])$ where A is an applicative structure (with product types), $[\![\cdot]\!]$ an interpretation function on Σ, and $v : A^t \to \{0,1\}$ such that:*

- $v(App([\![\forall_\sigma]\!], f)) = 1$ *iff* $v(fa) = 1$ *for all* $a \in A^\sigma$
- $v(App([\![\to]\!], (p, q)) = 1$ *iff* $v(p) = 0$ *or* $v(q) = 1$.

A model is Leibnizian *iff* $v(App([\![=_\sigma]\!], (a, b))) = 1$ *iff* $a = b$. *Where, recall,* $=_\sigma$ *is defined in terms of* $\forall_{\sigma\to t}$ *and* $\to$ *as above.*

A logical model on a substitution structure is defined as a logical model on the underlying applicative structure.

When investigating the logic of fineness of grain it's natural consider languages with additional logical constants that express further concepts of interest. Here is a representative, but non-exhaustive list:

- $Pure^\sigma$ of type $\sigma \to t$ expressing the property of being a pure entity of type σ.
- Fun^σ of type $\sigma \to t$ expressing the property of being a fundamental entity of type σ,
- $Cofun^{\sigma_1...\sigma_n}$ expressing the relation between type $\sigma_1...\sigma_n$ things that holds when they are cofundamental with one another.
- $Def^{\sigma_1...\sigma_n\tau}$ of type $(\sigma_1 \times ... \times \sigma_n \times \tau) \to t$, expressing metaphysical definability,
- $SubInst^\sigma$ of type $\sigma \times \sigma \to t$ expressing the relation of one thing being a substitution instance of another.
- For each $i \in M$, Sub_i^σ of type $\sigma \times \sigma \to t$ expressing the relation that holds between a and b when $b = ia$.

Supposing that $(A, v, [\![\cdot]\!])$ is a logical model based on a substitution structure (M, A) that is generated (has a basis), then the interpretations of these constants must be subject to the following constraints:

- $v([\![Pure^\sigma]\!]a) = 1$ iff a is pure.
- $\{a \in \bigcup_\sigma A^\sigma \mid v([\![Fun^\sigma]\!]a) = 1\}$ is a basis for (M, A).
- $v([\![Cofun^{\sigma_1...\sigma_n}]\!](a_1...a_n)) = 1$ iff there's a basis Y such that $a_1...a_n \in Y$
- $v([\![Def^{\sigma_1...\sigma_n\tau}]\!](a_1...a_n, c)) = 1$ iff c is metaphysically definable from $a_1...a_n$
- $v([\![SubInst^\sigma]\!](a, b)) = 1$ iff there is an $i \in M$ such that $ia = b$.

[52] $\forall_{((t\to t)\to t)}\forall_t$ intuitively says that every operator is satisfied by every proposition. The reason $\forall_{((t\to t)\to t)}\forall_t$ denotes the bottom element of the Boolean algebra in these models is because we are using the Boolean structure of A^t to interpret the quantifiers in a natural way.

- $v([\![Sub_i^\sigma]\!](a,b)) = 1$ iff $ia = b$.

Another useful predicate, metaphysical *anti*-definability, $ADef^{\sigma_1 \ldots \sigma_n \tau}(a_1 ... a_n, c)$ states that c is metaphysical definable from everything *apart* from $a_1 ... a_n$.

It is worth remarking that Sub_i^σ denotes a functional relation between things of type σ. One might wonder whether there is an actual functional element $s_i \in A^{\sigma \to \sigma}$ such that s_i maps each element $a \in A^\sigma$ to ia. Curiously, this is not in general possible. Consider the following two identities:[53]

- $s_i(s_j a) = i(s_j a) = i(ja)$
- $s_i(s_j a) = i(s_j a) = (is_j)(ia)$.

The first we get from the definition of s_i and s_j, and the second from the fact that substitutions distribute over application in substitution structures. Together they imply that, for any $a \in A^\sigma$, $ija = fia$ for some $f \in A^{\sigma \to \sigma}$ (specifically, $f = is_j$). But this equation cannot hold in full generality. For suppose that $ia = ib$. Then $f(ia) = f(ib)$ so $ija = ijb$. But this places severe limitations on the substitution structure: it means there cannot be more than two fundamental elements in any given type. For if there were three distinct elements a, b and c, one could choose i so that it mapped a and b to the same thing, and a and c to different things, and then choose j so that it mapped a to itself and b to c, so that $ia = ib$ but $ija \neq ijb$. (If Sub_i^σ is a legitimate relation but s_i is not in $A^{\sigma \to \sigma}$ this demonstrates a failure of the principle that for every functional relation of type $\sigma \to \sigma \to t$ there is a corresponding function of type $\sigma \to \sigma$.[54])

Note that Sub_i^σ is really a family of relations, with one for each $i \in M$. It would nice to be able to quantify over substitutions internally in the object language. There are several ways to achieve this. The easiest to implement involves augmenting the type theory with an extra type of substitutions, s. One can then introduce a relation $Sub^\sigma : (s \times \sigma \times \sigma) \to t$ with the following constraints:

- $A^s = M$
- $v([\![Sub^\sigma]\!](i,a,b)) = 1$ iff $ia = b$.

Given this primitive, one can then define (up to extensional equivalence) several other notions. For example, metaphysical definability would be defined as $Def(a_1, ..., a_n, c) := \forall_s i(Sub^{\sigma_1}(i, a_1, a_1) \wedge ... \wedge Sub^{\sigma_n}(i, a_n, a_n)) \to Sub^\tau(i, c, c))$. Or we could define a purity predicate, $Pure(a) = \forall_s i Sub(i, a, a)$.

The constraints we placed on the interpretations of Fun, Def, etc, do not single out their interpretations uniquely, they merely fix their extensions. One degree of freedom that this affords us is over whether to interpret Fun and Def as denoting fundamental elements themselves, or as denoting pure elements, or as elements that are neither fundamental nor pure. In section 5.3 we will consider a model in which the only fundamental elements are individuals, and Fun, Def, etc, are all interpreted by pure elements. However, when the fundamental elements include things with relational types (types ending in a t) then this approach is subject to some difficulties: in remark 7 we consider a paradox which uses the premise that purity is pure (formally: $Pure^{\sigma \to t}(Pure^\sigma)$).

[53] A little care is needed in interpreting our notational conventions here. Since s_i is an element of the substitution structure, $s_i a$ is short for $App^{\sigma\sigma}(s_i, a)$ whereas ia is short for $Sub^\sigma(i, a)$, the action of i on a.

[54] See the principle Plenitude in Dorr [9].

Another degree of freedom this affords us concerns the *intensional* behaviour of the constants Fun, $Pure$ etc. If the logical model contains intensional operators — i.e. elements $f \in A^{t \to t}$ and $p, q \in A^t$ such that $v(fp) \neq v(fq)$ even though $v(p) = v(q)$ — then you might wish to impose the constraints on the interpretations of these constants that is modally robust.

Let me end by discussing some limitations on metaphysical theorizing in terms of substitutions. We begin with the idea of taking the substitution relation as a primitive in the object language:

Remark 6. *Suppose that we have taken the above notion of substitution as primitive, and that there is exactly one fundamental element — a proposition of type t, represented by a propositional constant p. (The following problem does not seem to depend essentially on the idea that there is a most one fundamental thing, but it makes the presentation simpler.) Now we may define a diagonalization relation,* $Diag(A, B)$ *which intuitively says that the result of substituting A for p in A is B:* $Diag(A, B) := \forall i(Sub(i, p, A) \to Sub(i, A, B))$*. Although we have not laid out any axioms for Sub, it is clear that on its intended interpretation, one ought to have the following schema:* $Diag(A, q) \leftrightarrow q = A[A/p]$ *when A is closed and As only non-logical constants are p.*

Now we may emulate the proof of the diagonal lemma to generate a liar proposition. Let $\nu = \forall q(Diag(p, q) \to q)$*. Set* $\lambda = \forall q(Diag(\neg\nu, q) \to q)$*. Since we have* $\forall q(Diag(\neg\nu, q) \leftrightarrow q = \neg\forall q(Diag(\neg\nu, q) \to q))$*, we can simplify this using the definition of* λ *to get (*)* $\forall q(Diag(\neg\nu, q) \leftrightarrow q = \neg\lambda)$*. Now we can reason as follows:* λ *(i.e.* $\forall q(Diag(\neg\nu, q) \to q)$*) entails* $Diag(\neg\nu, \neg\lambda) \to \neg\lambda$*, so by (*)* $\neg\lambda$*. Conversely* $\neg\lambda$ *entails* $\exists q(Diag(\neg\nu, q) \wedge \neg q)$*, so by (*)* λ*. So* $\lambda \leftrightarrow \neg\lambda$*, which is a contradiction.*[55]

There are several morals one might draw from this. One is to deny that there can be fundamental elements of type t. This idea is not entirely without motivation, as we typically think of propositions as being constructed out of fundamental properties, relations and objects at other types, and not as primitive fundamental things in their own right. However, paradoxes like the above can arise with fundamental entities in any relational type by similar reasoning. In section 5.3 we shall see, however, that one can have a primitive substitution relation and fundamental entities at type e without inconsistency.

Another option is to avoid theorizing with a substitution relation Sub as introduced, and to theorize instead with weaker notions such as purity, metaphysical definability and cofundamentality. Here, again, further work is needed to show that these weaker notions do not also lead to paradoxes. Here is a paradox that arises just involving the predicate $Pure$, and a fundamental constant p:

Remark 7. *Suppose, again, that there is only one fundamental proposition, denoted by a constant p of type t. Then, given the intended extension of Pure, the substitution relation 'C is the result of substituting B for p in A' ought to be definable as* $\forall X((Pure(X) \wedge Xp = A) \to XB = C)$*. The diagonalization relation is thus* $Diag(A, B) := \forall X((Pure(X) \wedge Xp = A) \to XA = B)$*. Let* λ *and* ν *be defined as in remark 6, except with Diag defined as above. Then we may run the diagonal argument exactly as in 6.*

We can refine this result further: we do not need exactly one fundamental entity for the diagonalization argument to work, all we need is that p is the only non-pure expression

[55]One thing to note about this version of the liar paradox is that instead of resting on a linguistic truth or falsity predicate, it instead rests on the idea that quantification into sentence position ranges over things that behave, in certain respects, like sentences — specifically in their substitutional structure.

appearing in the diagonal sentence λ (and thus also ν). For suppose that λ can be written as αp where α is a closed expression containing only constants that denote pure elements. Then, $Diag(\alpha p, q) := \forall X((Pure(X) \wedge Xp = \alpha p) \to X(\alpha p) = q)$, and so (by instantiating X for α) we can derive: $Diag(\alpha p, q) \to q = \alpha(\alpha p)$. Conversely, we can prove that $q = \alpha(\alpha p) \to \forall X((Pure(X) \wedge Xp = \alpha p) \to X(\alpha p) = q)$ if we make the assumption that the decomposition of something into a pure operation and a fundamental argument is unique. This is valid given our constraints on the interpretation of Pure; see the discussion of Separated Structure in the next section. Thus we have $Diag(\alpha p, q) \leftrightarrow q = \alpha(\alpha p)$ (or $Diag(\lambda, q) \leftrightarrow q = \lambda[\lambda/p]$), and a similar biconditional for ν, which is exactly what we needed to derive the contradiction in remark 6.

Thus the situation is this: since the paradoxical sentence λ is definable in the lambda calculus from the logical constants $\to, \forall_{t\to t}$, p and $Pure$, paradox arises from the assumption that $\to, \forall_{t\to t}$ and $Pure$ are all themselves pure. In particular, it is natural to look for models in which $Pure$ is not interpreted by a pure element.[56]

5.2 Object language principles

I will not attempt here to give an extensive axiomatization of the predicates we introduced in the last section. I will instead look at a series of principles in the object language intended to capture various theses of interest concerning propositional granularity. Let us begin with a very strong thesis:

Structure $\phi\alpha = \psi\alpha \to \phi = \psi$

Here ϕ can be any term of type $\sigma \to \tau$ and α any term of type σ. Structure captures, in purely logical terms (i.e. without the need for extra predicates like Fun) an important aspect of the structured view: that if you apply two properties to the same argument and get the same result, the properties must be the same. Indeed, the structured picture intuitively motivates a seemingly stronger principle that $\phi\alpha = \psi\beta \to \phi = \psi \wedge \alpha = \beta$; both fall out of the general picture that a difference in constituents suffices for a difference in the resulting complex. It turns out that Structure is problematic enough as it is, so we shall focus on it.

Structure is inconsistent with the usual rules of the λ-calculus — a feature sometimes taken to be a part of higher-order logic. Self-composing the identity operator with itself yields the identity operator back. So does applying higher-order identity function of operators, since that maps every operator to itself. The λ-calculus calculus captures these facts with the identity $(\lambda X \lambda x\, X(Xx))(\lambda y\, y) = (\lambda X\, X)(\lambda y\, y)$, which is provable using β-equivalence and relabeling bound variables. But Structure immediately implies that self-composition is identity: $(\lambda X \lambda x\, X(Xx)) = (\lambda X\, X)$. This means that self-composing any operator with itself yields that operator back; this is clearly not the case since $\lambda x\, \neg\neg x$ is distinct from $\neg$. Or similarly, consider the property of self-loving, $\lambda x\, Lxx$, and the property of loving a, $\lambda x\, Lxa$. If you apply both of these properties to a you get the same proposition: Laa. But it's not true that self-loving is the same property as the property of loving a: one can have people who love themselves but not a, and people who love a but do not love themselves.[57]

Thus, if Structure is to even get off the ground, we must work in a version of higher-order logic that doesn't accept the usual rules of the λ-calculus: perhaps a version that accepts the

[56] I explore this option in other work, which is presently unpublished. See [ANON][REF].

[57] See Hodes [18], Dorr [9] and Goodman [17] for some other examples of this sort of phenomenon.

material equivalence, but not the identity, of β-equivalent terms of type t.[58] Unfortunately, Structure is inconsistent even in these weaker theories due to the infamous Russell-Myhill paradox (see, for example, Hodes [18], Dorr [9]).

However, there are weakenings of Structure that bear investigation. Here is one I am particularly interested in, when we have restricted ourselves to a fundamental language:

Separated Structure $\phi a = \psi a \to \phi = \psi$, where a is a constant, and ϕ and ψ does not contain a.

This version of structure is not obviously susceptible to a Russell-Myhill paradox. Moreover, it also appears to be consistent with the λ-calculus. In our first counterexample, $\neg p$ violates the constraint that the argument be a constant, and in the second counterexample, $\lambda x\, Lax$ violates the condition that the predicate not involve the argument a.

Separated Structure is evidently sensitive to the language it is stated in. For example, consider a false instance of Structure — where $\phi\alpha = \psi\alpha \to \phi = \psi$ is false — and consider the extended language in which we have introduced primitive constants F and G, to corefer with ϕ and ψ respectively, and constant a to corefer with α. The instance $Fa = Ga \to F = G$ now satisfies the syntactic restrictions on Separated Structure, but expresses the same falsehood as $\phi\alpha = \psi\beta \to \phi = \psi$. Separated Structure is thus only a good schema if it is stated in a fundamental language; this minimally includes the constraint that the constants do not denote structurally complex entities. This is vindicated by the following proposition:

Proposition 22. *Suppose (M, A) is a quasi-functional substitution structure with a full fundamental language $\mathcal{L}(\Sigma)$ relative to interpretation $[\![\cdot]\!]$. Suppose that $a \in \Sigma^\sigma$ and also that ϕ and ψ are two $\mathcal{L}^{\sigma\to t}(\Sigma)$ terms not containing a. Then if $[\![\phi a]\!] = [\![\psi a]\!]$, $[\![\phi]\!] = [\![\psi]\!]$.*

If $(M, A, v, [\![\cdot]\!])$ is additionally a Leibnizian logical model based on (M, A), then every instance of Separated Structure is true in the model.

Proof. By quasi-functionality and fullness, it suffices to show that $(\rho(i)[\![\phi]\!])(d) = (\rho(i)[\![\psi]\!])(d)$ for every $i \in Sub(\mathcal{L}(\Sigma))$, $d \in A^\sigma$ (where ρ is the monoid homomorphism associated with the fundamental language, as in definition 22). Let i and d be arbitrary. By fullness, d is of the form $d = [\![\delta]\!]$ for some term δ. Let j be like i except $j(a) = \delta$. So $\rho(j)[\![\phi a]\!] = [\![(j\phi)\delta]\!] = [\![(i\phi)\delta]\!] = [\![i\phi]\!](d) = (\rho(i)[\![\phi]\!])(d)$. By a symmetrical argument $\rho(j)[\![\psi a]\!] = \rho(i)([\![\psi]\!])(d)$. Since $\rho(j)[\![\phi a]\!] = \rho(j)[\![\psi a]\!]$ we get that $\rho(i)[\![\phi]\!](d) = \rho(i)[\![\psi]\!](d)$ as required.

Note that as a corollary to this proposition, that a quasi-functional, Leibnizian logical model with a fundamental language $\mathcal{L}(\Sigma)$ will validate Separated Structure. For if $v([\![\phi a = \psi a]\!]) = 1$, then $[\![\phi a]\!] = [\![\psi a]\!]$ by Leibnizianess, and by proposition 22, $[\![\phi]\!] = [\![\psi]\!]$, and so $v([\![\phi = \psi]\!]) = 1$. □

A similar result holds for logical models based on generated quasi-functional substitution structures.

It's worth noting that just as Structure can be strengthened to include the idea that distinctness of arguments entails the distinctness of the whole, so can Separated Structure. The resulting principle states: $\phi a = \psi b \to \phi = \psi \wedge a = b$, where a and b are constants, and ϕ and ψ do not contain a or b. However the result is no longer consistent with the ordinary λ-calculus. For example, one can prove $(\lambda x\, c)a = (\lambda x\, c)b$, for arbitrary a and b, since both sides reduce to c. This counterexample hinged on the use of vacuous λ-binding.

[58]Such a theory has no straightforward analogue of β-equivalence for types other than t (especially non-relational types that do not end in a t).

There's a variant theory of the λ-calculus called the λ_I-calculus which bans vacuous variable binding, and may be consistent with the strengthening of Separated Structure.[59] We shall not investigate this idea any further here.

It is possible to formulate language insensitive variants of Separated Structure at the expense of adding extra constants such as Fun and $Pure$. Consider the following principle:

Quantified Separated Structure $\forall XY(Pure(X) \wedge Pure(Y) \wedge Fun(z_1...z_n) \wedge X(z_1)...(z_n) = Y(z_1)...(z_n) \rightarrow X = Y)$

Here $Fun(z_1...z_n)$ is short for the sentence stating that $z_1...z_n$ are all fundamental, and different from one another.[60].

This principle informally states that the decomposition of a proposition into fundamental constituents via a pure operation is unique. Given that we are operating in a fundamental language, as we must for Separated Structure to make sense, we may make the assumption $Fun(c)$ for each non-logical constant c and $c \neq c'$ for distinct constants c and c'. If we additionally assume $Pure(\alpha)$ for every closed term containing only logical constants, we can prove each instance of Separated Structure. Let $c_1...c_n$ be the fundamental constants appearing in ϕ and ψ. We can rewrite ϕa as $\alpha c_1...c_n a$ and ψa as $\beta c_1...c_n a$, where α and β contain only non-logical constants. This is achieved by replacing the fundamental constants c_i in ϕ or ψ with a variable x_i of matching type and prefixing the result with the string $\lambda x_1...\lambda x_n$ (note that if ϕ or ψ doesn't contain c_i, the resulting λed out relation will ignore its ith argument). Then we may infer, from Quantified Separated Structure, that $\alpha = \beta$ and thus $\alpha c_1...c_n = \beta c_1...c_n$. Finally, by β-equivalence we may infer that $\phi = \psi$.

Another class of natural principles concerns the relation between metaphysical definability and constructability explored in section 4.3. Thus we might want:[61]

Metaphysical Decomposition $D(z_1...z_n) \rightarrow (Def(z_1, ..., z_n, c) \leftrightarrow \exists q(Pure(q) \wedge q(z_1)...(z_n) = c))$

Given theorem 19 it's easy to see that this principle holds in any Leibnizian generated quasi-functional logical model.

5.3 A Boolean model

To illustrate all of this, let me now briefly describe a concrete class of models which satisfies the constraints I have outlined so far. These are models where the propositions form a Boolean algebra. As we have already noted (examples 2 and 4), this does not preclude a sensible notion of substitution being defined on them. The basic idea is to let the algebra of propositions, A^t, be given by the powerset of M, thinking of the substitutions themselves as logically possible worlds. (The view of substitutions as logical possibilities follows naturally from the Bolzano-style definition of a logically necessary sentence as a sentence whose substitution instances are all true.) The action on this algebra is given by division:

- $A^t = P(M)$

[59] See Dorr [9] for a theory of granularity that relies essentially on the λ_I-calculus.

[60] That is the conjunction that contains the conjunct $z_i \neq z_j$, for each variable z_i and z_j of the same type with $i \neq j$

[61] This principle is a natural generalization of a principle suggested by Jeremy Goodman (unpublished notes). Goodman's principle is restricted to things definable from individuals, and is not stated in terms of metaphysical definability but a closely related notion (see the annihilator congruence in section 2.4).

- $ip = \{j \mid j \circ i \in p\}$.

We also assume some action of M on A^e. We may then consider the full M-set structure based on the M-sets A^t and A^e. It is easily seen that the action is faithful at A^t. We can turn this into a logical model by defining a valuation $v : A^t \to A^t$ as follows:

- $v(p) = 1$ iff $1 \in p$.

and by defining interpretations for the logical constants as follows:[62]

- $[\![\forall_\sigma]\!](j, f) = \{i \mid 1 \in f(i, a)\ \forall a \in A^\sigma\}$ where $j \in M$ and $f \in A^{\sigma \to t}(\subseteq M \times A^\sigma \to A^t)$
- $[\![\to]\!](j, (p, q)) = (M \setminus p) \cup q$ where $j \in M$ and $p, q \in A^t$.

Additional non-logical constants can be added in a similar way. For example:

- $[\![Fun^\sigma]\!](j, a) = M$ if $\sigma = e$ and $\emptyset$ otherwise.
- $[\![Def^{\sigma_1 \ldots \sigma_n \tau}]\!](j, (a_1..a_n, c)) = \{i \mid ic$ is metaphysically definable from $ia_1...ia_n\}$
- $[\![Sub^\sigma_k(j, (a, b))]\!] = \{i \mid ika = ib\}$

Without having shown that a substitution structure of this form has a basis we do not have a way to interpret predicates like Fun that are interpreted by basis elements. A natural compromise is to stipulate that Fun applies to the interpretation of non-logical constants in Σ (i.e. excluding logical constants that denote pure elements, like Fun itself). It follows that when the interpretation makes $\mathcal{L}(\Sigma)$ into a fundamental language for the substitution structure, Fun only applies to basis elements:

- $[\![Fun^\sigma]\!](j, a) = \{i \mid ia = [\![c]\!]$ for some non-logical constant $c \in \Sigma^\sigma\}$

Note that as we have defined them, the logical constants $\forall_\sigma$, $\to$, Def, Fun, and so on all denote pure elements. For example, the value of $[\![\to]\!](j, (p, q))$ is independent of j, and so $i[\![\to]\!] = [\![\to]\!]$ for all $i \in M$.

Let me end by demonstrating, with a simple example, that there *are* freely generated models of this type.[63] Let D be a non-empty set representing the individuals, and identify M with the set of substitutions of individuals: $M = D^D$. We obtain a full M-set structure by setting $A^e = D$, with the obvious action of M, and setting $A^t = P(M)$ as above, and taking the full M-set function space for functional types. A^e is a basis for the resulting M-set structure: for any type preserving function from $f : A^e \to A^e$, there is a unique substitution $i \in M$ that has the same behavior on the basis as f (just set $i := f$). By proposition 10, this also means that this structure has a fundamental language: it is the language which has a unique constant, c_a, for each individual $a \in A^e$.[64] (Indeed, in this

[62]Note that we have given the definition of $\to$ in its uncurried form (of type $(t \times t) \to t$) as it is simpler to parse. The curried conditional is defined by $[\![\to']\!](i, p)(j, q) = (M \setminus jp) \cup q$.

[63]Brown [6] also describes the following model, but for different purposes. In the special case where $M = (A^e)^{(A^e)}$, it turns out that this model is (non-obviously) isomorphic to a model considered by Jeremy Goodman in relation to a theory of individual aboutness. The description of Goodman's models, and the equivalence argument would take us too far afield.

[64]$[\![\cdot]\!]$ maps each constant $c_a \in \Sigma^e = \Sigma$ to the individual, a. If $i : \Sigma^e \to \mathcal{L}^e(\Sigma)$, we define $\rho(i)$ as follows: it maps an individual a, to the individual denoted by $i(c_a)$. (Note that $i(c_a)$ may not be a constant, as it may not be in $\eta\beta$ normal form. As it happens, the $\eta\beta$ normal form of any term in $\mathcal{L}^e(\Sigma)$ is a constant. For if a term of type e is of the form $\alpha\beta$, then α must be a λ term, as there are no functional constants, and so $\alpha\beta$ can be reduced and is not a normal form. No term of type e is of the form $\lambda x\, \alpha$ since e is not a functional type. So the normal form of any type e term has to be a constant).

case one can turn it into a full fundamental language by adding a logical constant for every pure element in the model.)

This example traded on properties specific to type e. It is, in general, an open question whether there are models like this that have bases in other types. For example, I do not know whether it is possible to find a Leibnizian logical model that has $\mathcal{L}(\{p : t\})$ as a fundamental language — i.e. a model based on a substitution structure generated by a single proposition p of type t.

5.4 Further model constructions

The following determines an extremely natural mapping from sentences of $\mathcal{L}(\Sigma)$ to the powerset of M.

Definition 30. *Suppose that $(B, [\![\cdot]\!], v)$ is a logical model of $\mathcal{L}(\Sigma)$ with a truth valuation $v : B^t \to \{0,1\}$. Suppose that M is a monoid of substitutions on $\mathcal{L}(\Sigma)$. Then we may define a mapping $M^{[\![\cdot]\!]} : \mathcal{L}^t(\Sigma) \to P(M)$ as follows:*

- $M^{[\![\cdot]\!]}(\phi) = \{i \in M \mid v([\![i\phi]\!]) = 1\}$

We omit the superscript when it is given by context.

If we define $iX = \{j \mid j \circ i \in X\}$ it is easy to show that $iM(\phi) = M(i\phi)$.

In what follows we fix a particular interpretation: for concreteness, say, a full 2 valued interpretation of $\mathcal{L}(\Sigma)$, $[\![\cdot]\!]$. With this interpretation fixed we just write $M(\phi)$ instead of $M^{[\![\cdot]\!]}(\phi)$.

We shall start off by proving a limited result for a quantifier-free language. Let $\mathcal{L}(\Sigma)$ be a higher-order language with the logical connectives $\to$ and $\neg$ and infinitely many non-logical constants at each type[65], and let M be the monoid of substitutions on non-logical constants of this language. Let $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$ be $\mathcal{L}(\Sigma)$ augmented by quantifiers $\forall_\sigma : (\sigma \to t) \to t$ for each σ, and let $M^\forall$ be the monoid of substitutions on the augmented language, treating $\forall_\sigma$ as a logical constant for every type σ and not available for replacement. Note that M also acts on $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$: it is the monoid of substitutions that don't replace any constants with terms involving quantifiers.

In what follows we shall run a construction that generates models for which $\mathcal{L}(\Sigma)$ and $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \text{ a type}\})$, respectively, are fundamental languages. However we shall see that the latter sorts of models are not Leibnizian: distinct elements end up being Leibniz equivalent in the model, making them less interesting. We'll then look at a construction of a Leibnizian model of $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$ whose whose substitutions consist of a proper submonoid of $M^\forall$ (namely the substitutions that do not substitute expressions containing quantifiers for constants described above). In this model the restricted substitutions commute with the intepretation function. It is open, as of writing, whether there could be a Leibnizian model where the full monoid $M^\forall$ commutes with the intepretation function.

$M(\cdot)$ has some nice logical features. In particular, it provides an interpretation of propositional logic in the Boolean algebra $\mathcal{P}(M)$, by mapping $\mathcal{L}^t(\Sigma)$ into $P(M)$ in such a way that the Boolean operations correspond to their set-theoretic analogues:

Proposition 23. *If M is a monoid of substitutions on a logical language $\mathcal{L}(\Sigma)$, and $(B, [\![\cdot]\!], v)$ is a logical model of $\mathcal{L}(\Sigma)$, then the connectives correspond to the set theoretic operations on $P(M)$:*

[65] It may be possible to relax this assumption to some degree, but it makes the following observations simpler.

1. $M(\neg\phi) = M \setminus M(\phi)$.

2. $M(\phi \wedge \psi) = M(\phi) \cap M(\psi)$

The following introduces a new sort of function space which will be useful in the following argument.

Definition 31. *A tethered M-set is a relation R between two M-sets, $A = Dom(R)$ and $B = Cod(R)$, such that Rab implies $Ria\,ib$ for all $i \in M$. (B is 'tethered' to A by R.)*

Given tethered M-sets R and S, we define the tethered function space $R \to S$ as follows:

- *$Dom(R \to S) = Dom(R) \to Dom(S)$ where $\to$ is M-set exponentiation.*
- *$Cod(R \to S) \subseteq Cod(R) \to Cod(S)$ is the set of $f : M \times Cod(R) \to Cod(S)$ such that*
  - *$f \in Cod(R) \to Cod(S)$ (i.e., $f(ij, ia) = if(j, a)$.)*
  - *There exists a $g \in Dom(R \to S)$ such that whenever Rab, and $i \in M$, $S(if)a\,(ig)b$.*
- *$(R \to S)fg$ iff whenever Rab, and $i \in M$, $S(if)a\,(ig)b$*

For $(f, g) \in (R \to S)$ and $(a, b) \in R$ we define application:

- *$App((f, g), (a, b)) = (f(1, a), g(1, b))$*

Note that R can be understood as an M-set itself, where i acts on a pair $(a, b) \in R$ by mapping it to $(ia, ib) \in R$. Note that clearly if $(f, g) \in (R \to S)$ then $(if, ig) \in (R \to S)$ since M is closed under composition.

Definition 32. *A tethered M-set structure is a substitution structure (M, R) such that:*

- *R^e and R^t are tethered M-sets.*
- *$R^{\sigma\to\tau} = R^\sigma \to R^\tau$ where $\to$ is the tethered function space constructor.*
- *For $(a, b) \in R^\sigma$, $i(a, b) = (ia, ib)$.*

Proposition 24. *Assume (M, R) is a tethered M-set structure. Let $A^\sigma = Dom(R^\sigma)$ and $B^\sigma = Cod(R^\sigma)$ for each σ. Then A and B are quasi-functional substitution structures with combinators, A is a full M-set structure and R is a surjective M-logical relation between A and B if it is surjective on types e and t.*

Proof. $A^{\sigma\to\tau} = Dom(R^{\sigma\to\tau})$ is a set of functions in $M \times Dom(R^\sigma) \to Dom(R^\tau)$, and by definition the action of M on $A^{\sigma\to\tau}$ is defined as in definition 23. Thus A is automatically quasi-functional. Similarly for $B^{\sigma\to\tau}$. Moreover, by definition $Dom(R^{\sigma\to\tau})$ is the full M-set exponent so A is full. The fact that R is a surjective M-logical relation follows from a straightforward induction. □

Proposition 25. *If R is a tethered M-set structure, and is functional on the base types (when considered as an M-logical relation), then R is functional at all types.*

Proof. Let $A^\sigma = Dom(R^\sigma)$ and $B^\sigma = Cod(R^\sigma)$ as in proposition 24.

By construction R is surjective at each type. For induction suppose it is functional at type τ: we show R is functional at type $\sigma \to \tau$. Suppose that $Rh\,f$ and $Rh\,g$. Then for any $a \in A^\sigma$ and $b \in B^\sigma$ such that Rab, it follows that $R(ih)a\,(if)b$ and $R(ih)a\,(ig)b$ for any $i \in M$. Since R^τ is functional it follows that $(ig)b = (if)b$ whenever $i \in M$ and Rab for some $a \in A^\sigma$. Since R is surjective it follows $(ig)b = (if)b$ for all $b \in B^\sigma$ and $i \in M$, and since B is quasi-functional, $f = g$. □

Now consider the substitution structure obtained as follows:

- M is the monoid of surjective substitutions on $\mathcal{L}(\Sigma)$
- $R^t = \{([\phi]_{\eta\beta}, M(\phi)) \mid \phi \in \mathcal{L}^t(\Sigma)\}$
- $R^e = \{([a]_{\eta\beta}, [a]_{\eta\beta}) \mid a \in \mathcal{L}^e(\Sigma)\}$
- $R^{\sigma\to\tau} = R^\sigma \to R^\tau$ where $\to$ is the tethered function space.

Note that since $iM(\phi) = M(i\phi)$, it follows that $(i[\phi]_{\eta\beta}, iM(\phi)) \in R$ whenever $([\phi]_{\eta\beta}, M(\phi)) \in R$. So $R^t ab$ implies $R^t ia\, ib$ as required (this clearly holds for R^e).

If we replace $\mathcal{L}(\Sigma)$ with $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$ and M with $M^\forall$ in the above definitions we get a substitution structure for $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$ based on the monoid $M^\forall$ instead. The following argument straightforwardly generalizes to that case as well.

Theorem 26. *$\mathcal{L}(\Sigma)$ is a fundamental language for the substitution structure (M, R) for some interpretation of $\mathcal{L}(\Sigma)$ in R.*

Proof. Let $A^\sigma = Dom(R^\sigma)$ and $B^\sigma = Cod(R^\sigma)$ as in proposition 24. Then by theorem 20, A is isomorphic to the term substitution structure. We can introduce an interpretation function $[\![\cdot]\!]^A$ on A so that $[\![\alpha]\!]^A$ corresponds to $[\alpha]_{\eta\beta}$ modulo this isomorphism.

Now note that R^e is functional (it is just identity) and R^t is functional, since if ϕ and ψ are $\eta\beta$ equivalent so are $i\phi$ and $i\psi$, and thus their interpretations are equal in any interpretation: so $M(\phi) = M(\psi)$. So by proposition 25, R is functional at every type. R is, in fact, an interpretation function from the term structure (which is isomorphic to A) into B. We can see this as follows. Suppose that $[\![\cdot]\!]^B$ is an interpretation function on B given by assigning values $[\![c]\!]^B = R([c]_{\eta\beta})$ to each constant c. It follows by the fundamental theorem of M-logical relations that:

- $R([\![\alpha]\!]^A) = [\![\alpha]\!]^B$

(Recalling that $[\alpha]_{\eta\beta} = [\![\alpha]\!]^A$.) So R (or at least $R \circ [\cdot]$) is identical to an interpretation function.

Moreover, since R is an M-logical relation, it follows that if Rab then $Ria\, ib$. That is to say: $i[\![\alpha]\!]^B = [\![i\alpha]\!]^B$. Thus $\mathcal{L}(\Sigma)$ is a fundamental language for B. Setting $[\![\alpha]\!]^R = ([\![\alpha]\!]^A, [\![\alpha]\!]^B)$ it's also a fundamental language for R, as claimed (M clearly commutes with $[\![\cdot]\!]^A$ since it is isomorphic to the term substitution structure).

□

Note: to ensure that theorem 26 applies, so that B is isomorphic to the term structure, you have to assume that M is the monoid of surjective substitutions on $\mathcal{L}(\Sigma)$ (as opposed to the monoid of all substitutions). A more general notion of tethered function space can be introduced so that $Dom(R)$ is simply identical to the term model without going through theorem 20, and M can be the full monoid of substitutions. Roughly the idea is to revise definition 31 by taking the function space constructors on Cod and Dom as primitive, so we can simply let $Dom(R^t) = T^t$ and $Dom(R^e) = T^e$ and $Dom(R^\sigma \to R^\tau) = T^{\sigma\to\tau}$.

Here is another perspective on the above construction. Suppose I built an M-set structure by setting $A^e = T^e$ and letting A^t be the M-set $P(M)$ with the action given by division — $iX = \{j \mid ji \in X\}$ (and letting $A^{\sigma\to\tau}$ be given by the M-set function space). Then we can set up an M-logical relation between A and the term model T by letting R^e be identity and R^t as above (i.e. $R^t = \{([\phi]_{\eta\beta}, M^{[\![\cdot]\!]}(\phi)) \mid \phi \in \mathcal{L}^t(\Sigma)\}$). We can generate an interpretation of $\mathcal{L}(\Sigma)$ in A by setting $[\![c]\!]^A$ to be anything that $[c]$ bears R^σ to as above. Unlike the more constrained construction above, the resulting interpretation has a lot of junk, and there's no guarantee that $i[\![c]\!] = [\![ic]\!]$. However, $i[\![c]\!]$ and $[\![ic]\!]$ are always related by R to $i[c]$: the former by the fact that Rab implies $Ria\,ib$, the latter by the fundamental theorem of M-logical relations. In fact the partial equivalence relation on A^σ given by:

- $a \sim_\sigma b$ iff there is a $c \in T^\sigma$ such that $R^\sigma ca$ and $R^\sigma cb$

is itself an M-logical relation on A. According to this relation $i[\![c]\!] \sim_\sigma [\![ic]\!]$ for each constant, and thus for each term (again, by the fundamental theorem). Therefore the quotient $A/\sim$ is a quasi-functional substitution structure, and by transferring the interpretation just constructed to the quotient structure, we can see that it has $\mathcal{L}(\Sigma)$ as a fundamental language.

The construction we in fact gave using tethered M-sets cuts out the quotienting stage by having a leaner function space constructor (indeed, you only admit functions that stand in the M-logical relation to something in the tethering space, so the need to quotient is eliminated).

As mentioned, the above construction works if we instead let T be the term structure for $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$, and set R^t to be the ordered pairs $([\phi], M^\forall(\phi))$ where $M^\forall(\phi) = \{i \in M^\forall \mid |i\phi| = 1\}$ where $|\cdot|$ is the same boring two valued structure we fixed in defining $M : T^t \to A^t$ earlier. The resulting model has an obvious valuation: $v(([\phi], M^\forall(\phi))) = 1$ iff $1 \in M^\forall(\phi)$. But the resulting valuation is quite clearly not Leibnizian. $v([\![\phi]\!]) = 1$ iff $1 \in M^\forall(\phi)$ iff $|1\phi| = |\phi| = 1$. Thus the constructed model agrees with the boring structure about the truth of all sentences. Since there are only two propositions according to interpretation $|\cdot|$, there are only two propositions according to the constructed model.

Lets return to our construction for the quantifier free language, $\mathcal{L}(\Sigma)$. Suppose that we follow the second take on the construction, where you have to quotient at the end. Recall that A is the full M-set structure generated by setting $A^t = P(M)$ and $A^e = T^e$, and we have a M-logical relation $R : T \to A$. Thus, as before, we have an interpretation of the quantifier free fragment of $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$ by picking interpretations $[\![c]\!]$ of constants c such that :

- $R[c][\![c]\!]$

We can get a model of $\mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$ by picking the interpretations of $\forall_\sigma$. We interpet $\forall_\sigma$ so that it quantifies over the entirety of A^σ:

- $[\![\forall_\sigma]\!](j, f) = \{i \mid 1 \in f(i, a) \text{ for all } a \in A^\sigma\}$.

Notice that the quantifiers so defined are pure: j is ignored. Thus every term $\alpha \in \mathcal{L}(\Sigma \cup \{\forall_\sigma \mid \sigma \in Typ\})$ has an interpretation $[\![\alpha]\!]$.

Now note that for any $i \in M$ (but *not* any $i \in M^\forall$) we have the following:

- $i[\![\alpha]\!] = [\![i\alpha]\!]$

This is straightforward to prove by induction on terms. It can be seen directly as well, by noting that by construction it holds of all the non-logical constants, and it trivially holds of the logical constants, because they are sinks, and our substitutions do not move logical constants. Moreover, we have seen already that substitutions fix the interpretations of combinators in a substitution structure. Thus the result can be seen by decomposing α into a combinator applied to a sequence of logical and non-logical constants, and the fact that substitutions commute with application.

6 Concluding Remarks

In the foregoing we have introduced a class of structures modeling propositions, properties, and other entities of the type hierarchy. This model incorporates a fragment of the structured picture of these entities in the form of the notion of a substitution. We have shown how a number of metaphysically useful concepts can be characterized in purely substitution-theoretic terms, and shown that they can be applied even the structured theory of propositions is not assumed — indeed, we saw that these notions can be modeled even in relatively coarse-grained settings, including settings where propositions are modeled by sets of indices. We investigated a concrete class of substitution structures, M-set structures, which are representative of quasi-functional substitution structures and which in particular allowed us to get a concrete model of the elements of functional types. We then showed how to introduce interpretations of higher-order languages capable of expressing fundamentality and other substitution-theoretic notions, and introduced some object language principles of interest.

References

[1] Peter Aczel. Replacement systems and the axiomatization of situation theory. *Situation theory and its applications*, 1:3–33, 1990.

[2] VI Arnold. *Mathematical Methods of Classical Mechanics (second edition).* Springer-Verlag, 1989.

[3] Andrew Bacon. The broadest necessity. *Journal of Philosophical Logic*, 47(5):733–783, 2018.

[4] Andrew Bacon. Is reality fundamentally qualitative? *Philosophical Studies*, pages 1–37, forthcoming.

[5] Max Black. The identity of indiscernibles. *Mind*, 61(242):153–164, 1952.

[6] Chad E Brown. M-set models. *Reasoning in Simple Type Theory: Festschrift in Honour of Peter B. Andrews on his 70th Birthday*, (17):175–186, 2008.

[7] M. J. Cresswell. Hyperintensional logic. *Studia Logica*, 34(1):25–38, 1975.

[8] M. J. Cresswell. *Structured Meanings.* MIT Press, 1985.

[9] Cian Dorr. To be F is to be G. *Philosophical Perspectives*, 30(1):39–134, 2016.

[10] M Mehdi Ebrahimi and M Mahmoudi. The category of m-sets. *Italian journal of pure and applied mathematics*, pages 123–132, 2001.

[11] Kit Fine. Properties, propositions and sets. *Journal of Philosophical Logic*, 6(1):135–191, 1977.

[12] Kit Fine. A theory of truthmaker content i: Conjunction, disjunction and negation. *Journal of Philosophical Logic*, 46(6):625–674, 2017.

[13] Kit Fine. A theory of truthmaker content ii: Subject-matter, common content, remainder and ground. *Journal of Philosophical Logic*, 46(6):675–702, 2017.

[14] Harvey Friedman. Equality between functionals. In *Logic Colloquium*, pages 22–37. Springer, 1975.

[15] Peter Fritz and Jeremy Goodman. Higher-order contingentism, part 1: Closure and generation. *Journal of Philosophical Logic*, 45(6):645–695, 2016.

[16] R. O. Gandy. On the axiom of extensionality–part i. *Journal of Symbolic Logic*, 21(1):36–48, 1956.

[17] Jeremy Goodman. Reality is not structured. *Analysis*, 77(1):43–53, 2017.

[18] Harold T. Hodes. Why ramify? *Notre Dame Journal of Formal Logic*, 56(2):379–415, 2015.

[19] Mati Kilp, Ulrich Knauer, and Alexander V Mikhalev. *Monoids, acts and categories.* de Gruyter Expositions in Mathematics, vol. 29 (Walter de Gruyter & Co., Berlin, 2000), 2000.

[20] Serge Lapierre. Structured meanings and reflexive domains. *Studia Logica*, 51(2):215–239, 1992.

[21] Tom Leinster. The Yoneda lemma: What's it all about. Online note available at: http://www.maths.ed.ac.uk/ tl/categories/yoneda.ps, 2000.

[22] David Lewis. General semantics. *Synthese*, 22(1-2):18–67, 1970.

[23] John C Mitchell. *Foundations for programming languages*, volume 1. MIT press Cambridge, 1996.

[24] John C Mitchell and Eugenio Moggi. Kripke-style models for typed lambda calculus. *Annals of Pure and Applied Logic*, 51(1-2):99–124, 1991.

[25] Daniel Nolan. Hyperintensional metaphysics. *Philosophical Studies*, 171(1):149–160, 2014.

[26] Gordon Plotkin. Lambda-definability and logical relations. Technical report, University of Edinburgh, 1973. SAI-RM-4, School of Artificial Intelligence, Edinburgh (1973).

[27] A. N. Prior. *Objects of Thought.* Clarendon Press, 1971.

[28] Agustin Rayo and Stephen Yablo. Nominalism through de-nominalization. *Noûs*, 35(1):74–92, 2001.

[29] Scott Soames. Direct reference, propositional attitudes, and semantic content. *Philosophical Topics*, 15(1):47–87, 1987.

[30] Robert Stalnaker. Varieties of supervenience. *Philosophical Perspectives*, 10:221–42, 1996.

[31] Timothy Williamson. Everything. *Philosophical Perspectives*, 17(1):415–465, 2003.

[32] Stephen Yablo. *Aboutness.* Princeton University Press, 2014.