Another stab at TTT

Randall Holmes

March 28, 2019: generalizing and debugging pass; fixes to some proofs; added intro material on type theories and NF(U) and section divisions; fix to one of the local approximation constructions. Mar 27: subtle point about images under σ maps of atoms noted and fixed, this time correctly, though this involved massive debugging of indexing of litters in supports, which may have introduced new bugs.

1 Introduction to type theories and NF(U)

We review a familiar theory. TST is a many-sorted theory with sorts indexed by the natural numbers with primitive relations of logic and membership. Each variable x (considered as a typographical object) has a natural number type $\texttt{type}(x)$; there is a countable supply of variables of each type. An atomic formula $u = v$ (again, considered as a typographical object with u and v as constituent typographical objects) is well-formed iff $\texttt{type}(u) = \texttt{type}(v)$ and an atomic formula $u \in v$ is well-formed iff $\texttt{type}(u) + 1 = \texttt{type}(v)$.

The first axiom scheme of TST is extensionality, the collection of formulas of the shape

$$(\forall xy : (\forall z : z \in x \leftrightarrow z \in y) \rightarrow x = y).$$

Note that the common type of x and y will be positive and one greater than the type of z, The second axiom scheme is comprehension, the collection of formulas of the shape

$$(\exists A : (\forall x : x \in A \leftrightarrow \phi)),$$

where ϕ is a formula in which A does not appear free. This completes the basic definition of the theory, though axioms of Infinity and Choice are often adjoined.

The intuitive picture of course is that type 0 is inhabited by individuals of an unspecified character, while each type $n+1$ is inhabited by sets of type n objects. The axiom of extensionality expresses the natural criterion for identification of objects of each positive type, and the axiom of comprehension tries to assert that any condition whatsoever on type n objects determines a type $n+1$ object, and of course does not succeed in doing this.

We now describe some variations. TSTU differs from TST only in weakening the axiom of extensionality to apply only to nonempty objects:

$$(\forall xyw : w \in x \wedge (\forall z : z \in x \leftrightarrow z \in y) \rightarrow x = y).$$

Intuitively this theory differs from TST in allowing an arbitrary collection of atoms with no elements in each positive type, in addition to the sets provided by comprehension. It is convenient in formalizing TSTU to provide a constant $\emptyset^{\mathbf{n}}$ in each type $n+1$ with no elements; this allows us to talk about sets as opposed to atoms, the objects with no elements in positive types which are not equal to the appropriate $\emptyset^{\mathbf{n}}$.

We describe the theories NF and NFU.

NF is the single sorted theory with equality and membership whose axioms are extensionality

$$(\forall xy : (\forall z : z \in x \leftrightarrow z \in y) \rightarrow x = y).$$

and the scheme of stratified comprehension,

$$(\exists A : (\forall x : x \in A \leftrightarrow \phi)),$$

where A is not free in ϕ and ϕ admits a stratification, a function σ from variables appearing (free or bound) in ϕ to natural numbers such that in each atomic subformula $u = v$ we have $\sigma(u) = \sigma(v)$ and in each atomic subformula $u \in v$ we have $\sigma(u)+1 = \sigma(v)$. A formula with a stratification is said to be a stratified formula. In effect, the comprehension axioms of NF are those which could be made formulas of TST by a suitable assignment of types to constituent variables. We note against the criticism that this is merely a syntactical trick that the stratified comprehension scheme is equivalent to a finite conjunction of its instances. It is possible to present the axioms of NF in a way which makes no reference to types at all. We note in support of those who see an essential connection to the typed theory that the very first thing one would want to do with such a presentation of NF would be to prove the stratified comprehension scheme as a metatheorem.

The theory NFU differs from NF only in having the weak extensionality axiom instead of the strong one. It would be usual to specify a particular empty object $\emptyset$ as the empty *set* and refer to any others as the atoms.

Now we discuss typical ambiguity in TST and TSTU. Suppose that we have a bijection $x \mapsto x^+$ from variables to variables of positive type, with the property that $\texttt{type}(x^+) = \texttt{type}(x) + 1$ for every x. If ϕ is a formula, the result ϕ^+ of replacing each variable (free or bound) with its image under $(x \mapsto x^+)$ is also a formula. Moreover, if ϕ is an axiom, so is ϕ^+, and if ϕ is a theorem, so is ϕ^+ (because the type-raising operation commutes nicely with proof rules).

We define the Ambiguity Scheme (briefly, Amb) as the collection of all closed formulas $\phi \leftrightarrow \phi^+$. It is a theorem of Specker that TST + Amb is equiconsistent with NF, and the proof adapts to prove that TSTU + Amb is equiconsistent with NFU.

We now describe a version of Jensen's proof of the consistency of TSTU + Amb, and so of NFU, carried out by constructing a suite of models of TSTU in ZF.

Let λ be a limit ordinal. Construct a function V with domain λ such that for each $\alpha < \beta < \lambda$, we are given an injection $f_{\alpha,\beta}$ from $\mathcal{P}(V(\alpha))$ into $V(\beta)$. With each $s : \omega \to \lambda$ which is strictly increasing, we associate a model M_s of TSTU in which type i is implemented as $V(s(i))$ and the membership of type i objects in type $i+1$ objects is defined as $x \in_i y \equiv_{\texttt{def}} (\exists z : x \in z \wedge f_{s(i),s(i+1)}(z) = y)$.

Let Σ be an arbitrary finite set of closed formulas of TSTU. Let n be greater than the type of any variable which appears in a formula in Σ. We define a partition of $[\lambda]^n$ in which the partition to which a set A belongs is determined by the truth values of the formulas in Σ in models M_s of TSTU where A is the range of $s\lceil n$. This partition of $[\lambda]^n$ into $\leq 2^{|\Sigma|}$ partitions has an infinite homogeneous set H by Ramsey's theorem, which contains the range of a strictly increasing $h : \omega \to \lambda$. The model M_h satisfies TSTU and the restriction of the Ambiguity Scheme to formulas in Σ. By compactness, TSTU + Amb is consistent, and by Specker's result NFU is consistent.

Jensen's consistency proof is the starting point of my 1995 development of "tangled type theory". It is not at all clear how to adapt Jensen's argument to the theory with strong extensionality: if it were straightforward, the problem would not be open!

My initial observation was that Jensen's proof could be presented in terms of a different type theory with urelements in each type. TTTU_λ is a first order sorted theory with sorts belonging to λ. $u = v$ is well-formed iff $\texttt{type}(u) = \texttt{type}(v)$. $u \in v$ is well-formed iff $\texttt{type}(u) < \texttt{type}(v)$ (but this theory is **not** a cumulative type theory: we are not saying that type α is included in type β when $\alpha < \beta$). For any formula ϕ of the language of TSTU and strictly increasing $s : \omega \to \lambda$, the formula ϕ^s is the formula of the language of TTTU_λ obtained by injectively replacing each variable of type i in ϕ with a variable of type $s(i)$. Note that this does yield a formula of TTTU_λ. We then declare that the axioms of TTTU_λ are the formulas ϕ^s for each sequence s and each axiom ϕ of TSTU.

We note that the structure described above is a model of TTTU_λ, with type α implemented as $V(\alpha)$ and $x \in_{\alpha,\beta} y$ for each $\alpha < \beta < \lambda$ implemented as $(\exists z : x \in z \wedge f_{\alpha,\beta}(z) = y)$. Jensen's proof of the consistency of NFU can be recast as a two step argument: first, there is a model of TTTU_λ, and then, the existence of a model of TTTU_λ implies the consistency of TSTU + Amb and so of NFU. We won't write this out here, but we do write out the analogous result for NF immediately.

Define TTT_λ (tangled type theory with λ types) exactly as above, except that its axioms are of the form ϕ^s where ϕ is an axiom of TST.

We will argue that TTT_λ is consistent if and only if NF is consistent, but first we pause to point out that this is an extremely weird theory. Each element of a type β has an extension relative to each lower type α, and each of these extensions, considered by itself, determines the object of type β exactly. So type β is understood as the "power set" of *each* lower type. These cannot be true power sets. If $\alpha < \beta < \gamma$, and we had type β implementing each subset of type α, and type γ implementing each subset of type β, then type γ would by Cantor's theorem be larger than the collection of all subsets of type α, and we could not define an extensional membership relation of type α objects in type γ objects. It should also be noticed that the well-formed formulas determining instances of comprehension are quite restricted, being determined in each case by a fixed subsequence of types in λ playing the role of the sequence of types of TST.

If NF is consistent, then TTT_λ is consistent: given a model of NF, let each type of the model of TTT_λ be the domain of the model of NF, and let the membership relation of the model of NF serve as the membership relation $\in_{\alpha,\beta}$ for each $\alpha < \beta < \lambda$. (One could also use $D \times \{\alpha\}$ as type α, where D is the domain of the model of NF, and then define $(x, \alpha) \in_{\texttt{TTT}} (y, \beta)$ as $\alpha < \beta < \lambda \wedge x \in_{\texttt{NF}} y$; this is a matter of style. We think of a model of type theory as determined by a sequence of types and membership relations between each appropriate pair of types, which means that we do not need to assume that the types are disjoint. If the types are taken to be disjoint, then of course we can use a single relation to implement membership).

If TTT_λ is consistent, then NF is consistent. Let Σ be an arbitrary finite set of closed formulas of TST. Let n be greater than the type of any variable which appears in a formula in Σ. We define a partition of $[\lambda]^n$ in which the partition to which a set A belongs is determined by the truth values of the formulas in Σ in models M_s of TST where A is the range of $s\lceil n$, the model M_s having type $s(i)$ of the model of TTT_λ as type i for each i, and $\in_{s(i),s(i+1)}$, the restriction of the membership of TTT_λ to membership of type $s(i)$ objects in type $s(i+1)$ objects, as the implementation of membership of type i objects in type $i+1$ objects. M_s is a model of TST because the translation of each axiom of TST into its language is an axiom of TTT_λ. This partition of $[\lambda]^n$ into $\leq 2^{|\Sigma|}$ partitions has an infinite homogeneous set H by Ramsey's theorem, which contains the range of a strictly increasing $h : \omega \to \lambda$. The model M_h satisfies TST and the restriction of the Ambiguity Scheme to formulas in Σ. By compactness, TST + Amb is consistent, and by Specker's result NF is consistent.

Of course this is not a proof of the consistency of NF, because the existence of a model of TTT_λ is a far from obviously true hypothesis.

Further, it is very difficult to get one's mind around how TTT works. In the version of my NF proof which I will present, I do not attempt to construct a model of TTT_λ (though I do in fact succeed in doing so, and may discuss this after the main result is shown), but instead construct a model of ZFA in which there is a peculiar system of cardinals called a *tangled web*. The relationship of this to TTT_λ is that TTT_λ in fact sees tangled webs internally (or something approximating them) when considering the relationships between systems of cardinals related to its own type structure [I believe that an ω-model of TTT_λ actually sees tangled webs]. I'm willing to talk about this, though the details are maddening, but it is not relevant to verifying my proof of Con(NF), which only requires that we understand the definition of a tangled web and the proof that existence of a tangled web implies the consistency of NF.

2 The model construction

What follows is in intention a description of a model of tangled type theory with λ types and a proof that it really is that. This is extremely nasty and I am still debugging, but I think it ought to straighten out with effort.

working set theory: We work in ZFAC.

parameters of the construction: This construction has three important parameters:

1. A regular uncountable cardinal κ. which may be taken initially to be ω_1. Sets of size $< \kappa$ are called *small*. All other sets are called *large*. Types in the structure we define will not contain all subsets of lower types but will contain all (nonempty) small subsets of lower types as elements.
2. A limit ordinal $\lambda < \kappa$. Ordinals less than λ are indices of types and may be called type indices on occasion. One may initially take λ to be ω. Taking $\lambda < \kappa$ helps us keep supports small. $\lambda \leq \kappa$ might be fine.
3. A strong limit cardinal $\mu > \kappa$ with cofinality $\geq \lambda$. This will be the common cardinality of all types of the structure we build, from the standpoint of the ambient ZFAC. One may initially take μ to be $\beth_\omega$.

preliminary description of the types: The structure we build will have types indexed by the ordinals below λ. Each type will be of size μ. The types are pairwise disjoint sets. We will note in advance that type $\alpha < \lambda$ in our interpretation of TTT with λ types will be represented by type $\alpha + 2$ in this structure.

Type 0 consists of μ atoms. So does any type indexed by a limit ordinal $< \lambda$.

Type $\alpha + 1$ consists of μ subsets of type α (the exact way these are chosen to be specified later), an atom $\emptyset^{\alpha+1}$ (the empty set for that type) and μ further atoms; we will refer to the atoms other than $\emptyset^{\alpha+1}$ in type $\alpha + 1$ and all the atoms in type 0 or any type with limit ordinal index as "nonsets".

partition of the nonsets into litters; local cardinals: The nonsets of each type are partitioned into μ sets of size κ called *litters*. Each litter which is a subset of type α belongs to type $\alpha + 1$. For any litter L of type $\alpha+1$, the collection of subsets of type $\alpha+1$ which contain only nonsets and have small symmetric difference from L is called the local cardinal of L, briefly written $[L]$. If L belongs to type $\alpha+1$, $[L]$ belongs to type $\alpha+2$ (and so all of its elements belong to type $\alpha+1$). We refer to the set of local cardinals of type $\alpha+1$ sets as $K_{\alpha+2}$.

discussion of near-litters: Elements of any $\bigcup K_{\alpha+2}$ are called *near-litters*. For any near-litter N, we let N° represent the unique litter with small symmetric difference from N. We let $[N]$ denote the local cardinal $[N^\circ]$ to which N belongs.

subcollections of the local cardinals: Each set $K_{\alpha+2}$ is partitioned into κ sets $K^\epsilon_{\alpha+2}$ ($\epsilon < \kappa$) of size μ called echelons. An element of any $K^\epsilon_{\alpha+2}$ is said to have echelon ϵ. A litter has the same echelon as its local cardinal, An element of a litter has the same echelon as the litter to which it belongs. The echelon of a near-litter is defined as the supremum of the echelons of its elements. We provide notation $\texttt{echelon}(x)$ for the echelon of a nonset, near-litter or local cardinal x. Echelons $K^\epsilon_{\alpha+2}$ should certainly not be expected to be elements of a type.

Each set $K_{\alpha+2}$ has designated subsets $\texttt{rng}_0(\sigma_{\beta+2,\alpha+2})$ for each β with $\beta+1 < \alpha$, and a designated subset $\texttt{rng}_0(\beta_{\alpha+2})$, and each set $K_{\beta+3}$ has designated subsets $\texttt{rng}_0(\tau_{\beta+2,\alpha+2})$, when $\beta+1 < \alpha$. The designated sets are all disjoint from one another and each of them meets each echelon in the appropriate $K_{\gamma+2}$ in a set of cardinality μ.

the recursive construction of the type structure: We give a description of the recursive construction of the types.

base types already described: Type 0 and any types present with limit ordinal index have already been described: they are collections of μ atoms.

types just above base types: Type $\nu+1$. where ν is non-successor, consists of the unions of small and co-small collections of litters of type $\nu+1$. the empty set of type $\nu+1$, and the atoms of type $\nu+1$.

description of double successor types commences: From this point on the type we are constructing is type $\alpha + 2$, and all lower types are supposed already constructed. We are given certain information about types $\beta < \alpha + 2$ to start with.

reprise of general facts about types: The types are pairwise disjoint. Each type is of size μ. Each type whose index is not a successor is a collection of μ atoms (termed nonsets). Each type whose index is a successor $\beta+1 < \alpha+2$ contains μ nonempty subsets of type β, the special atom $\emptyset^{\beta+1}$, and μ other atoms, termed nonsets. As noted above, the nonsets in each type (including types with index higher than $\alpha+2$) are equipped with the partition into litters, and the litters and local cardinals belong to the appropriate types.

Definition (potential elements): We define the potential elements of type $\beta + 2$ as the subsets of type $\beta + 1$ and the atoms of type $\beta + 2$. Notice that there are more potential elements than actual elements by Cantor's theorem combined with our stated intention that each type has μ elements.

special maps already constructed: Some special maps have been provided for coding purposes.

σ **maps:** A map $\sigma_{\beta+2,\gamma+2}$ injectively mapping type $\beta + 2$ into $\texttt{rng}_0(\sigma_{\beta+2,\gamma+2}) \subseteq K_{\gamma+2}$ has been defined whenever $\beta < \alpha$, for all $\gamma > \beta + 1$.

τ **maps:** An injective map $\tau_{\beta+2,\gamma+2}$ from type $\gamma+1$ into $\texttt{rng}_0(\tau_{\beta+2,\gamma+2}) \subseteq K_{\beta+3}$ has been defined for each $\gamma \leq \alpha$ and each β with $\beta + 1 < \gamma$.

β **maps:** For each type $\beta + 2$ with $\beta < \lambda$, an injective map $\beta_{\beta+2}$ from the nonsets of type $\beta + 2$ into $\texttt{rng}_0(\beta_{\beta+2}) \subseteq K_{\beta+2}$ has been defined.

We provide the map $\beta'_{\beta+2}$ sending each atom x to $\beta_{\beta+2}(x) \cup 0$ and each subset $\beta_{\beta+2}(x) \cup k$ of type $\beta + 1$ to $\beta_{\beta+2}(x) \cup (k + 1)$ and fixing all other subsets of type $\beta + 1$ (where a natural number 0, k, $k + 1$ here represents the set of all elements of type $\beta + 1$ (other than nonsets) of that cardinality; i.e., these are the Frege natural numbers over a type) has been defined for each $\beta < \alpha$. What we need to notice about $\beta'_{\beta+2}$ is that it is a bijection from potential elements of type $\beta+2$ to the subsets of type $\beta+1$. We note also that we can understand what set is intended by $\beta'_{\alpha+2}(X)$ for x any atom of type $\alpha+2$ or subset of type $\alpha+1$, though we cannot yet tell whether this will belong to type $\alpha+2$. We observe, though it is not needed for the induction, that for $\beta < \alpha$ it is actually the case that $\beta'_{\beta+2}$ sends elements of type $\beta+2$ to elements of type $\beta+2$, serving as a bijection from objects of type $\beta+2$ to sets of type $\beta+2$.

note about notation: For each of the special functions g, there is no particular reason to believe that $\texttt{rng}(g)$ is exactly $\texttt{rng}_0(g)$; there is a technical condition on how the functions are constructed which makes it inconvenient to try to enforce exact equality of the ranges with the sets set aside originally to provide space for them.

motivation of σ maps; preliminary codes: We have been told above that the map $\sigma_{\beta+2,\alpha+2}$ maps type $\beta+2$ injectively to a subset of $K_{\alpha+2}$. The idea is that a subset B of type $\beta+2$ can be coded into type $\alpha+2$ in a preliminary way as $\bigcup \sigma_{\beta+2,\alpha+2}$"$B$, a potential element of type $\alpha+2$: we call this the preliminary code of B and refer to all unions of elements of the range of $\sigma_{\beta+2,\alpha+2}$ as "preliminary codes for subsets of type $\beta+2$ (of potential type $\alpha+2$)". We expect that all sets in type $\beta+3$ and some other collections of type $\beta+2$ objects will have preliminary codes which are actually sets of type $\alpha+2$.

every potential element of type $\alpha+2$ codes a subset of type $\beta+2$: We now exhibit our scheme for ensuring that every potential element of type $\alpha+2$ can be taken to code a subset of type $\beta+2$. The map $\tau_{\beta+2,\alpha+2}$ is an injection from type $\alpha+1$ into $K_{\beta+3}$ (as we have already been told). The idea is that this allows us to take any subcollection B of type $\alpha+1$ (or atom B in type $\alpha+2$) and map it deviously into the collection of preliminary codes of subcollections of type $\beta+2$, thus eventually allowing construction of a bijection between potential elements of type $\alpha+2$ and the preliminary codes by the Schroder-Bernstein method. To transform a particular potential element of type $\alpha+2$, first apply $\beta'_{\alpha+2}$ to be sure we get a set (and we do describe above how to apply this to any potential element of type $\alpha+2$), then take the union of the images of the elements of this set under $\tau_{\beta+2,\alpha+2}$, which is a subset of type $\beta+2$, then code this to a potential element of type $\alpha+2$ as above. Note that there is no reason to believe that the subsets of type $\beta+2$ coded in this process are actually sets of type $\beta+3$. For x a subset of type $\alpha+1$ or atom of type $\alpha+2$, we define $f^0_{\beta+2,\alpha+2}(x)$ as $\bigcup \sigma_{\beta+2,\alpha+2}``(\bigcup \tau_{\beta+2,\alpha+2}``(\beta'_{\alpha+2}(x)))$. The map $f^0_{\beta+2,\alpha+2}$ sends the subsets of type $\alpha+1$ and atoms of type $\alpha+2$ injectively into the collection of preliminary codes for subcollections of type $\beta+2$ (and we will provide that images under $f^0_{\beta+2,\alpha+2}$ of objects of type $\alpha+2$ are sets of type $\alpha+2$). The map $f_{\beta+2,\alpha+2}$ maps each subset of type $\alpha+1$ (and atom of type $\alpha+2$) which is not a preliminary code for a subset of type $\beta+2$, and each iterated image under $f^0_{\beta+2,\alpha+2}$ of such an object, to its image under $f^0_{\beta+2,\alpha+2}$ and fixes all other potential elements of type $\alpha+2$. This map is a bijection from potential elements of type $\alpha+2$ to preliminary codes for subcollections of type $\beta+2$ of potential type $\alpha+2$, and allows every potential element of type $\alpha+2$ to be interpreted as a subset of type $\beta+2$.

definition of type-skipping membership: The membership relation $x \in_{\beta+2,\alpha+2} y$ is defined as holding if $\sigma_{\beta+2,\alpha+2}(x) \subseteq f_{\beta+2,\alpha+2}(y)$, if $\alpha+2 > \beta+3$. $x \in_{\beta+2,\beta+3} y$ is defined as $x \in \beta_{\beta+3}(y)$. In both cases y should be an actual not a merely potential element of type $\alpha+2$. With care, these will turn out to be the membership relations of the desired model of tangled type theory (which will have type $\alpha+2$ of our structure as its type α).

definition of allowable permutations: We now define groups of permutations on parts of this structure, where we allow permutations π of the atoms which move only nonsets (of course we do not want them to move the "empty sets" in each type) to act on sets by the rule $\pi(A) = \pi``A$.

An $\alpha+2$-allowable permutation π is a permutation of atoms moving only nonsets (we do want to fix the empty sets in each type!) satisfying the following conditions:

domain: The permutation π is defined on the atoms in types $\leq \alpha+2$, with induced action on sets with no other atoms in their transitive closures.

respects local cardinals: The action of π fixes each $K_{\beta+2}$ for $\beta \leq \alpha + 2$ (e.g., it fixes $K_{\alpha+4}$).

respects f and β maps: The action of π fixes each $f_{\beta+2,\alpha'+2}$ for $\alpha' \leq \alpha$; and fixes each $\beta_{\beta+2}$ for $\beta \leq \alpha + 2$.

weird conjugation condition with σ maps: For any $\beta + 1 < \gamma \leq \alpha$, the composition $\sigma^{-1}_{\beta+2,\gamma+2} \circ \pi \circ \sigma_{\beta+2,\gamma+2}$, which we denote by $\pi_{\beta+2,\gamma+2}$, has the precise action of a $\beta + 2$-allowable permutation (note that it is defined on suitable atoms and iterated singletons of atoms, so there is enough information to determine uniquely what $\beta + 2$-allowable permutation if any it would agree with). There is no presumption whatever that $\pi_{\beta+2,\gamma+2}$ agrees with π. We define $\pi_{\alpha+1,\alpha+2}$ as the permutation obtained by restricting π to act only on atoms of type $\leq \alpha + 1$ (which will be $\alpha + 1$-allowable if α is successor, or invariably if we adopt the following point).

ν- and $\nu + 1$- allowable permutations? It does look as if the definition works for nonsuccessor ordinals and their successors, with some fine tuning. There may be some use for it. A ν-allowable permutation is a permutation of the atoms in types $\leq \nu$ whose action fixes $K_{\nu+2}$ and whose restriction to atoms of types $\leq \gamma$ for any fixed $\gamma < \nu$ is a γ-allowable permutation. A $\nu + 1$-allowable permutation of the atoms in types $\leq \nu + 1$ is a permutation of the atoms in types $\leq \nu + 1$ whose action fixes $K_{\nu+3}$ and whose restriction to atoms of types $\leq \gamma$ for any fixed $\gamma < \nu + 1$ is a γ-allowable permutation.

Definition (extended type index): An *extended type index* is defined as a nonempty finite subset A of λ with the property that any maximal open interval included in $[\mathtt{min}(A), \mathtt{max}(A)] \setminus A$ is of the form $(\gamma + 2, \delta + 2)$, an interval between double successors. For any extended type index A with more than one element, we define A_1 as $A \setminus \{\mathtt{min}(A)\}$. Note that this allows one or two smallest elements, but no others, of the extended type index to be other than double successors.

Definition (derivatives of an allowable permutation): The collection of derivatives of an α-allowable permutation π is the smallest collection of permutations which contains π and contains each $\pi'_{\gamma+2,\delta+2}$ if it contains π' (including the ones with successive indices obtained by restriction). We reiterate that there is no presumption of agreement between any derivatives of π, including ones of the same index.

We define a compact notation for derivatives. For any nonempty set A of double successor ordinals $< \lambda$ and $\mathtt{max}(A)$-allowable permutation π, we define a derivative π_A. Define $\pi_{\{\alpha\}}$ as π. Define π_A where $|A| \geq 2$ as $(\pi_{A_1})_{\mathtt{min}(A),\mathtt{min}(A_1)}$. If the minimum of A is not a double successor, π_A is an appropriate restriction of π_{A_1}.

It is important to note that $\sigma_A \circ \pi_A = (\sigma \circ \pi)_A$ for any $\mathtt{max}(A)$-allowable permutations σ and π.

Definition (support): A $\alpha + 2$-support is a well-ordering on a small set of pairs (x, A) where each x is a nonset or near-litter and each A is an extended type index with maximum $\alpha + 2$, and $\mathtt{min}(A)$ is the type of x if x is a nonset and the type of the nonset elements of x if x is a near-litter, and moreover if (x, A) and (y, A) are in S, x and y have no common elements. An object X has $\alpha + 2$-support S iff S is an $\alpha + 2$-support and for each $\alpha + 2$-allowable permutation π, if $\pi_A(x) = x$ for every $(x, A) \in S$, it follows that $\pi(X) = X$.

definition of support; specification of the sets in type $\alpha + 2$: The sets in type $\alpha + 2$ are exactly those nonempty sets X which have an $\alpha + 2$-support $\leq_S$. The empty set in each type is of course already implemented.

condition for this construction to succeed: To show that this works, we need to show that this collection of sets is of size $\leq \mu$ for each n.

choosing more special maps, with attention to echelon: Note that once the sets in type $\alpha + 2$ are specified, we can specify each map $\sigma_{\alpha+2,\gamma+2}$ for $\gamma > \alpha + 1$, and once each set in type $\alpha + 1$ is specified, we can specify each map $\tau_{\beta+2,\alpha+2}$ (we need this information for subsequent stages). The echelon of the image of an object of type $\beta + 2$ under a σ, τ, or β map must dominate the echelons of the first projections of elements of some $\beta + 2$-support of the object (this condition is why we make no requirement that the ranges of the special functions coincide exactly with the sets set aside to accommodate their ranges). It is also useful to note that the set

$\beta'_{\alpha+2}(x)$ for any x has the same support as x whenever x has one, so is in type $\alpha + 2$ iff x is in type $\alpha + 2$.

We have now actually given a complete description of a model of tangled type theory. But it will take considerable effort to show that this works.

3 The proof that the model works

Nothing above is to be construed as proving that any of this works. I am merely describing a construction. The construction will succeed as long as the described collection of sets to be taken as the sets of type $\alpha + 2$ is not of size greater than μ. We certainly do not claim that it is obvious that it will not be too large. There is further work after showing the construction succeeds to show that it satisfies appropriate axioms.

relation of τ maps to allowable permutations: We begin by computing the required relationship between $\alpha+2$-allowable permutations and the maps $\tau_{\beta+2,\alpha+2}$. If x is an element of type $\alpha + 1$, then $\{x\}$ is fixed by $\beta'_{\alpha+2}$; the union of $\tau_{\beta+2,\alpha+2}``\{x\}$ is simply $\tau_{\beta+2,\alpha+2}$; the union of the elementwise image of this set under $\sigma_{\beta+2,\alpha+2}$, $\bigcup \sigma_{\beta+2,\alpha+2}``\tau_{\beta+2,\alpha+2}(x)$ is then immediately seen to be $f^0_{\beta+2,\alpha+2}(\{x\})$ and in fact is $f_{\beta+2,\alpha+2}(\{x\})$, this last because $\{x\}$ is certainly not a preliminary code. Let π be an $\alpha + 2$ allowable permutation. We know that $f_{\beta+2,\alpha+2}$ is fixed by π, so $\pi(f_{\beta+2,\alpha+2}(\{x\}))$ must be $f_{\beta+2,\alpha+2}(\{\pi(x)\})$, that is,

$$\bigcup \sigma_{\beta+2,\alpha+2}``\tau_{\beta+2,\alpha+2}(\pi(x)).$$

We also know that $\pi(\bigcup \sigma_{\beta+2,\alpha+2}``\tau_{\beta+2,\alpha+2}(x))$ must be

$$\bigcup \sigma_{\beta+2,\alpha+2}``\pi_{\beta+2,\alpha+2}(\tau_{\beta+2,\alpha+2}(x)),$$

whence it follows that $\tau_{\beta+2,\alpha+2}(\pi(x)) = \pi_{\beta+2,\alpha+2}(\tau_{\beta+2,\alpha+2}(x))$.

Definition (special modifications of type indices): Define A_2 as $(A_1)_1$.

Definition (strong support): We now describe a notion of strong support, which we will need for careful analysis of the behavior of allowable permutations. A strong support $\leq_S$ is one in which

1. any element (x, A) of the domain of $\leq_S$ in which x is a near-litter has x actually a litter.
2. for any element (x, A) of the domain of $\leq_S$ which has x an atom, we have $(L, A) \leq_S (x, A)$, where L is the litter containing x.

3. if (x, A) is in the domain of $\leq_S$ and x is a litter and $\mathtt{min}(A)+1 = \mathtt{min}(A_1)$ and $[x] = \sigma_{\beta+2,\mathtt{min}(A_2)}(y)$, then

   $$\{(y, B\backslash A_2) : (y, B) \leq_S (x, A) \wedge \mathtt{max}(B\backslash A_2) = \beta+2 \wedge \mathtt{echelon}(y) < \mathtt{echelon}(x)\}$$

   is a $\beta + 2$-support of y.
4. if (x, A) is in the domain of $\leq_S$ and x is a litter and $\mathtt{min}(A)+1 = \mathtt{min}(A_1)$ and $[x] = \tau_{\mathtt{min}(A)+1,\alpha+2}(y)$, and $\mathtt{min}(A_2) = \alpha + 2$, then

   $$\{(y, B\backslash A_2) : (y, B) \leq_S (x, A) \wedge \mathtt{max}(B\backslash A_2) = \alpha+1 \wedge \mathtt{echelon}(y) < \mathtt{echelon}(x)\}$$

   is an $\alpha + 1$-support of y.
5. if (x, A) is in the domain of $\leq_S$ and x is a litter and $\mathtt{min}(A)+1 = \mathtt{min}(A_1)$ and $[x] = \beta_{\mathtt{min}(A)+2}(y)$ and $\mathtt{min}(A_2) = \mathtt{min}(A_1) + 1$, then $(y, A_2) \leq (x, A)$.

It is straightforward to establish the existence of a strong support whose domain extends the domain of a given support, using the properties of our assignment of echelons.

We now establish a result on freedom of action of allowable permutations.

Definition (locally small approximation): A $\alpha+2$-locally small approximation is a map π_0 taking pairs (x, A) where A is an extended type index with maximum $\alpha + 2$ and x is a nonset of type $\mathtt{min}(A)$ or a local cardinal of type $\mathtt{min}(A) + 2$ to an object of the same kind as x. For convenient statement of further conditions, we define $\pi_0^A(x) = \pi_0(x, A)$ The further conditions are

1. Each map π_0^A is an injective map with domain equal to its range.
2. The intersection of the domain of each π_0^A with any litter is small (empty being a frequent implemention of "small" here).
3. The domain of each π_0^A for which $\mathtt{min}A_1 = \mathtt{mn}(A) + 1$ includes every local cardinal of litters of the appropriate type which is not either in the range of a β, σ or τ map; it fails to contain the local cardinal of an image under $\tau_{\beta+2,\tau+2}$ iff $\beta + 2, \tau + 2$ are the two smallest elements of A_1; it fails to contain any image under a β or σ map. The domain of each π_0^A for which $\mathtt{min}A_1 \neq \mathtt{min}(A) + 1$ includes every local cardinal of the appropriate type.

A locally small approximation π_0 is realized by an $\alpha+2$-allowable permutation π iff $\pi_A(x) = \pi_0(x, A)$ for every x and A such that (x, A) is defined.

Definition (exception of a permutation): An exception of an n-allowable permutation π is a nonset x which belongs to a litter L for which either $\pi(x) \notin \pi(L)^\circ$ or $\pi^{-1}(x) \notin \pi^{-1}(L)^\circ$.

Theorem (freedom of action): Every $\alpha+2$-locally small approximation π_0 is actually realized by an $\alpha+2$-allowable permutation π with no exceptions of any π_A not found in the domain of π_0^A.

Proof of freedom of action theorem: We prove this by proving a more precise stronger result. Let π_0 be our locally small approximation. For each pair of litters L, M of the same type and each extended type index A with minimum the common type of elements of L and M, provide a map $\pi^A_{L,M}$, a bijection from the collection of elements of L not in the domain of π_0^A to the collection of elements of M not in the domain of π_0^A. The stronger result is that there is a uniquely determined $\alpha+2$-allowable permutation realizing the local approximation and agreeing with each $\pi^A_{L,M}$.

We prove the stronger result for $\alpha+2$ by induction, assuming the result true for $\beta+2 < \alpha+2$, and by induction on strong supports, showing that we can proceed to define values by recursion on the structure of a strong support, thus eventually assigning values at every relevant atom.

The character of the recursion is that we assume when considering (x, A) in a strong support $\leq_T$ along which we are computing that we have already computed $\pi_B(y)$ for each $(y, B) \leq_T (x, A)$.

If we consider an atom x with (x, A) appearing in a strong support, we have by inductive hypothesis already computed $\pi_A(L)$ where L is the litter containing x, because (L, A) appears earlier in the support. We can then compute $\pi_A(x)$, either as $\pi_0^A(x)$ or as $\pi^A_{L,\pi_A(L)^\circ}(x)$. Notice that in this case x is not an exception of π_A.

When we consider a litter L with (L, A) appearing in a strong support and either $\mathtt{min}(A)+1 \neq \mathtt{min}(A_1)$, or it is not an image under a β, σ, or suitable τ map we are given a value $\pi_0^A([L])$ which we can adopt as $\pi_A([L])$ at its local cardinal. We can then determine $\pi_A(L)^\circ$, and we can compute the value of π_A at each element of L, either by applying the local approximation or by applying $\pi^A_{L,\pi_A(L)^\circ}$, and the set of all these images will be $\pi_A(L)$.

If $[L] = \sigma_{\gamma+2,\beta+2}(x)$, and (L, A) is in the domain of the support and $\texttt{min}(A)+1 = \texttt{min}(A_1)$, then we have already values of each appropriate derivative $\pi_{B\cup A_2}$ at each first projection of an element of an $\gamma + 2$-support of x. We extend this assignment of values to a $\gamma + 2$-local approximation π'_0 to $\pi_{A_2\cup\{\gamma+2\}}$, using the same $\pi^B_{L,M}$'s where relevant and the same values at local cardinals not values of relevant β, σ, τ maps where stipulated or computed and otherwise choosing such values arbitrarily, which realizes a $\gamma+2$-allowable π'. We succeed in computing a value for $\pi'(x)$ which must be the same as $\pi_{A_2\cup\{\gamma+2\}}(x)$ for the permutation π we are trying to construct, and this gives us a computed value $\sigma_{\gamma+2,\beta+2}(\pi'(x))$ for $\pi_A([L])$, and so allows computation of a value for $\pi_A(L)$: we can compute the value of π_A at each element of L, either by applying the local approximation or by applying $\pi^A_{L,\pi'(L)^\circ}$.

If $[L] = \tau_{\gamma+2,\beta+2}(x)$, and (L, A) is in the support and $\texttt{min}(A) + 1 = \texttt{min}(A_1)$, we further note that we supply values in the local approximation if it is not the case that $\gamma + 2$ and $\beta + 2$ are the two smallest elements of A_1: this is the only case we have to deal with. We are given by inductive hypothesis computed values for all relevant derivatives $\pi_{A_2\cup B}$ at an $\beta + 1$-support of x. We extend these values to a $\beta + 1$-local approximation π'_0 to π_{A_2}, which we extend to a $\beta + 1$ allowable π', allowing us to compute $\pi'(x)$ which must agree with $\pi_{A_2}(x)$, and compute $\tau_{\gamma+2,\beta+2}(\pi'(x))$ as the value of $\pi_A(\tau_{\gamma+2,\beta+2}(x)) = \pi_A([L])$, and this allows computation of a value for $\pi_A(L)$: we can compute the value of π_A at each element of L, either by applying the local approximation or by applying $\pi^A_{L,\pi'(L)^\circ}$.

If $[L] = \beta_{n+2}(x)$ and (L, A) appears in the strong support and $\texttt{min}(A)+1 = \texttt{min}(A_1)$, with $\texttt{min}(A_1)+1 = \texttt{min}(A_2)$, we can compute $\pi_{A_2}(x)$ and from this determine $\pi_A([L])$ and $\pi_A(L)$ as above.

One needs to argue further that it is not possible for a different value to be computed for the value of a derivative of π along two different supports for the same atom or litter. If this happens, it must happen at a first point in some particular support: there must be a first item in the support for which more than one value can be computed along different supports. This cannot be an atom labelled with a type index, because the value computed at an atom is precisely determined by the value computed at the litter containing it, which precedes it. So one would have to have a litter labelled with a type index which had different values computed along two supports, one of which, moreover, has the property that values computed at all of its elements are uniquely determined independent of support used. But then we can construct a single support whose domain is the union of these two supports, along which a single value can be computed which must agree with the values computed along the other two.

We have two further things to show. We need to show that the collection of subsets of type $\alpha+1$ with support relative to $\alpha+2$-allowable permutations is of size $\leq \mu$ (it is obviously of size $\geq \mu$), and we need to verify that the axioms of TTT_λ hold in the resulting structure.

Definition: action of permutations on supports: For any support $\leq_S$ and allowable permutation π of suitable index, we define $\pi^*(\leq_S)$ as $\{((\pi_A(x), A), (\pi_B(y), B)) : (x, A) \leq_S (y, B)\}$. It is straightforward to verify that if $\leq_S$ is a support of X, $\pi^*(\leq_S)$ is a support of $\pi(X)$.

Definition (coding function): For each element x of type $\alpha+2$ and $\alpha+2$-support $\leq_S$ for x, let the coding function $\chi_{x,\leq_S}$ be defined by

$$\chi_{x,\leq_S}(\pi^*(\leq_S)) = \pi(x)$$

for each $\alpha + 2$-allowable permutation π. Note that if

$$\pi^*(\leq_S) = (\pi')^*(\leq_S),$$

we have $(\pi' \circ \pi^{-1})^*(\leq_S) =\leq_S$ and therefore $(\pi^{-1} \circ \pi')(x) = x$, so $\pi(x) = \pi'(x)$: the definition is effective. Note that the domain of the coding function is an orbit in the collection of all supports under the starred action of the $\alpha + 2$-allowable permutations.

Proof that there are not more than μ objects in any type: We argue that there are $< \mu$ coding functions for type $\alpha+2$ sets for each α, from which it follows that there are $\leq \mu$ objects of type $\alpha + 2$, since each object is a value of a coding function at a support and there are μ supports.

For type 1 sets, the only support elements are atoms in type 0 and near-litters in type 1, and the coding functions simply determine which intersections of domain elements and their complements (treating atomic elements for the nonce as if they were singletons) are to be included in the set constructed. The orbits in the supports are exactly determined by the mix of near-litters and atoms and the information about the positions of near-litters in given positions in the order containing atoms in different positions in the order, for κ orbits in the supports, and no more than 2^κ coding functions with each orbit as domain.

We claim as an inductive hypothesis that type α has $\leq \exp^\alpha(\kappa)$ coding functions with strong supports in their domains [where $\exp(|x|)$ is defined as $2^{|x|}$, and iteration is defined as usual, taking suprema at limits].

For type $\alpha+1$ sets, we describe a representation of coding functions. We choose an $\alpha+1$- support $\leq_S$ of a set X (without loss of generality a strong support). Let $\leq_{S'}$ be the restriction of $\leq_S$ to items appropriate in an α-support. For each element $x \in X$, choose a strong support $\leq_T$ which is an end extension of $\leq_{S'}$ to which we apply the coding function $\chi_{x,\leq_T}$. By inductive hypothesis, the coding function $\chi_{x,\leq_T}$ is taken from a collection of $\leq \exp^\alpha(\kappa)$ coding functions. We claim that the coding function $\leq_{X,\leq_S}$ is exactly determined by the orbit of $\leq_S$ and the collection of $\chi_{x,\leq_T}$'s just chosen. Our claim is that X is the set of images of supports $\leq_U$ which are end extensions of $\leq_{S'}$ under functions in the collection of $\chi_{x,\leq_T}$'s just chosen. Every $x \in X$ is of this form because of the way the collection of coding functions was chosen. The question is whether any additional elements are found in this set. Suppose that $x' = \chi_{x,\leq_T}(\leq_U)$: $\leq_T$ and $\leq_U$ have a common orbit specification and a common initial segment $\leq_{S'}$. There is an allowable permutation of suitable index sending $\leq_T$ to $\leq_U$ and so x to x' because $\leq_U$ is in the domain of the coding function. Define a local approximation sending each (x, A) where (x, A) is in the domain of $\leq_T$ and x is a nonset or x is an exception of π_A either appearing as the first projection of a domain element of $\leq_T$ or mapped to the first projection of a domain element of $\leq_U$, or $\pi(x)$ is an element of $M \Delta M^\circ$ where (M, A_1) appears in the domain of $\leq_U$ [near-litters appearing in the domain of $\leq_T$ are litters but this may not be true of $\leq_U$] to $\pi(x)$, and also sending each (x, A) in the domain of $\leq_S \setminus \leq_{S'}$ to x. This can be extended to an allowable permutation π' with no exceptions other than those indicated. Now observe that π'_A must send each near-litter L with (L, A) in the domain of $\leq_T$ to $\pi_A(L)$ [which we call M]: certainly π'_A sends $[L]$ to $\pi_A([L]) = [M]$, because it acts correctly on a support of $[L]$; π_A and π'_A agree on preimages of elements in $M \Delta M^\circ$, so any object not in L mapped into M or any object in L mapped out of M by π'_A must be an exception of π', and π' agrees with π at all exceptions of π which are in L or mapped into M and has no exceptions π does not have. Similarly, π'_A fixes each litter L with (L, A) in the domain of $\leq_S \setminus \leq_{S'}$: suppose (L, A) were the first bad item. We would have $[L]$ fixed by π', so it could only fail to be fixed if there were an exception of π'_A in L or mapped into L, and there can be no object with this behavior in the domain of the local approximation. Since $\pi'(\leq_T) =<_U$, $\pi'(x) = x'$. Since $\pi^*(\leq_S) =\leq_S$, we have $\pi'(X) = X$. It follows that $x' \in X$.

Thus there are $\leq \exp^{\alpha+1}(\kappa)$ coding functions for type $\alpha + 1$ sets for each orbit in the $\alpha + 1$-supports.

The orbit of a strong support under allowable permutations is entirely determined by a formal computation which we now describe. We compute an *orbit specification* of a strong support $\leq_S$, which is also the specification of any $\pi^*(\leq_S)$. The orbit specification is a well-ordering of the same length as $\leq_S$. We use the notation S_γ for the γ'th item in $\leq_S$, and we use π_1 and π_2 for the first and second projection operators. We use S'_α for the corresponding item in the specification.

1. If $S_\alpha = (x, A)$ with x an atom, we define S'_α as $(1, \gamma, A)$, where $\pi_1(S_\gamma)$ is the litter containing x.
2. If $S_\alpha = (L, A)$ with $[L]$ not an image under a σ or β map or under a $\tau_{\gamma+2,\beta+2}$ with $\gamma + 2$ and $\beta + 2$ the two smallest elements of A_1, we define S'_α as $(2, A)$.
3. If $S_\alpha = (L, A)$ with $[L] = \sigma_{\gamma+2,\delta+2}(x)$, we define S'_α as $(3, g, L, A)$, where g is a coding function sending the $\gamma + 2$-strong support of x to x, and L is the set of ordinal positions of elements of this $\gamma + 2$ support in $\leq_S$, a subset of α.
4. If $S_\alpha = (L, A)$ with $[L] = \tau_{\gamma+2,\delta+2}(x)$, we define S'_α as $(4, g, L, A)$, where g is a coding function sending a $\beta + 1$-strong support of x

to x, and L is the set of ordinal positions of elements of this $\beta+1$ support in $\leq_S$, a subset of α.

5. If $S_\alpha = (L, A)$ with $[L] = \beta_{\beta+2}(x)$ and $\mathtt{min}(A_1) + 1 = \mathtt{min}(A_2)$, we define S'_α as $(5, \gamma, A)$, where $S_\gamma = x$.

A strong support has an orbit specification: by definition of strong support we can find suitable supports.

Nothing seems to prevent a support having more than one orbit specification. It should be clear (perhaps after a little computation) that if $\leq_S$ has an orbit specification, $\pi^*(\leq (S))$ has the same orbit specification.

The interesting fact is that if $\leq_S$ and $\leq_T$ are $\alpha+2$-supports and have the same orbit specification, there is an $\alpha+2$-allowable permutation such that $\pi^*(\leq_S) =\leq_T$. This π is constructed using the freedom of action theorem as realizing a certain local approximation. The local approximation sends (x, A) in the domain of $\leq_S$ to (y, A) in the corresponding position in $\leq_T$ if x, y are nonsets. Suppose that one has successfully set up the local approximation for all items in $\leq_S$ which appear before (L, A), L a near-litter. Let (M, A) be the corresponding item in $\leq_T$. We already have enough information to be sure that a permutation extending the local approximation constructed so far will have $\pi([L]) = [M]$: the trick is to choose a small set of further values to ensure that L is sent exactly to M. For each $a \in L^\circ \setminus L$, we map (a, A_-) to some (a', A_-) with $a' \in M$. For each $b \in M^\circ \setminus M$, we map (b', A_-) to (b, A_-) for some $b' \in L$. We then need to choose iterated images and preimages of the a's and b''s added to our domain, the only constraints being that elements of near-litters appearing in $\leq_S$ should have images in corresponding near-litters in $\leq_T$ and elements of near-litters appearing in $\leq_T$ should have preimages in corresponding near-litters in $\leq_S$. The fact that L is mapped exactly to M is enforced by the fact that the permutation realizing a local approximation by freedom of action has no exceptions not explicitly given in the local approximation. We choose countably many new values for each of a small collection of aberrant elements of near-litters, so we have added only a small collection of new values of the local approximation.

Thus there are exactly as many orbits in the $\alpha+2$-supports (containing a strong support, but this does not limit our result, as every object has a strong support) as there are orbit specifications, and there are clearly no more than $\exp^{\alpha+2}(\kappa)$ orbit specifications, since an orbit specification is a small list of objects taken from not very large sets, the largest being sets of coding functions which by inductive hypothesis have size less than $\exp^{\alpha+1}(\kappa)$.

Types ν and $\nu+1$ have μ elements for ν non-successor.

Coding functions for type $\alpha+2$, by considerations above, are each determined by an orbit specification for a support taken from a set of supports which has cardinality no more than $\exp^{\alpha+2}(\kappa)$ and a set of coding functions taken from a set of size no more than $\exp^{\alpha+2}(\kappa)$, so there are no more than $\exp^{\alpha+2}(\kappa)$ coding functions, so no more than μ elements of type $\alpha+2$, and our construction succeeds.

This completes the argument that each type is actually successfully constructed.

We verify that the axioms of TTT will hold in this structure if the construction succeeds. It is evident that extensionality will hold if one examines the construction: in each case. every element of type $n+2$ is associated with a uniquely determined subcollection of type $m+2$ as its preimage under the defined membership relation.

Now we need to verify comprehension.

First, we note the effect of permutations on the defiined membership relations:

$$x \in_{\beta+2,\alpha+2} y \equiv \sigma_{\beta+2,\alpha+2}(x) \subseteq f_{\beta+2,\alpha+2}(y) \equiv \pi(\sigma_{\beta+2,\alpha+2}(x)) \subseteq \pi(f_{\beta+2,\alpha+2}(y)$$

$$\equiv \sigma_{\beta+2,\alpha+2}(\pi_{\beta+2,\alpha+2}(x)) \subseteq f_{\beta+2,\alpha+2}(\pi(y)) \equiv \pi_{\beta+2,\alpha+2}(x) \in_{\beta+2,\alpha+2} \pi(y).$$

In an interpretation of an instance of the comprehension axion of TTT, we have a finite sequence s of types, and each atomic subformula is of the form $x = y$, where x, y are of the same type, or $x \in_{s(i)+2,s(i+1)+2} y$, where x is of type $s(i)+2$ and y is of type $s(i+1)+2$ (recall that we interpret type α of TTT using type $\alpha+2$ in our structure). Performing the transformation above for a fixed allowable permutation π of suitable index will cause each variable to be decorated with a fixed derivative of π determined by its type, and such decorations on bound variables can then be removed, because the maps are permutations of the domains to which the quantifiers are restricted. If x in an instance $\{x : \phi\}$

of comprehension is of type $\alpha + 2$ (interpreting type n of TTT, this establishes something, but not quite what we want. Action of an $\gamma+2$-allowable permutation, where $\gamma + 2$ is the highest type appearing in ϕ, will cause action of various derivatives of the permutation on the parameters (none of which are of type higher than $\gamma + 2$). Of course each parameter has a strong $\gamma + 2$-support (in which all near-litters are litters), so if we require that all applicable derivatives of our $\gamma+2$-allowable permutation fix all first projections of elements of the union of the supports of the parameters, they will fix the set defined by the instance of comprehension (this is excessive, I just don't want to fine tune the description). What this shows is that the extension defined by an instance of comprehension is fixed by the action of a certain derivative of any $\gamma + 2$-allowable permutation all of whose derivatives fix all relevant elements of a certain support. Here $\gamma + 2$ is the highest type mentioned in the instance of comprehension and the derivative takes it down to an $s(i + 1) + 2$-allowable permutation, where $s(i) + 2$ is the type of x.

We use the freedom of action theorem. Let π be an $s(i) + 3$-allowable permutation fixing appropriate support elements from the support described above along with its derivatives. Let x be an element of the extension in question, with $s(i) + 2$-strong support $\leq_T$. Let π' be an $\gamma + 2$ allowable permutation fixing all nonset elements of the relevant support (including ones π cannot see) and sending all atoms in $\leq_T$'s domain and all exceptions of π which lie in or map into elements of $\leq_T$'s domain to their images under appropriate derivatives of π: this exists and has no exceptions other than indicated values by the freedom of action theorem. It is then straightforward to see that litters in $\leq_T$ are also sent to the correct values and litters in the original support are fixed (their local cardinals are mapped correctly because a support is: if the litters were not, there would be exceptions which construction of π' using the freedom of action theorem averts), so $\pi'(x) = \pi(x)$, but also π' fixes the extension because it fixes all elements of the support, so π maps the extension into itself. Consideration of π^{-1} as well shows that π fixes the extension. Thus the extension has an $s(i) + 3$-support and is a set in type $s(i) + 3$ of our structure, and there is also a set with the same extension in type $s(i + 1) + 2 \geq s(i) + 3$.

This completes the verification that we have a model of tangled type theory, whence NF is consistent.