Homotopy Type Theory: Unified Foundations of Mathematics and Computation*

Steve Awodey Robert Harper

December 14, 2015

Homotopy type theory is a recently-developed unification of previously disparate frameworks, which can serve to advance the project of formalizing and mechanizing mathematics. One framework is based on a computational conception of the type of a construction, the other is based on a homotopical conception of the homotopy type of a space. The computational notion of type has its origins in Brouwer's program of intuitionism, and Church's λ-calculus, both of which sought to ground mathematics in computation (one would say "algorithm" these days). The homotopical notion comes from Grothendieck's late conception of homotopy types of spaces as represented by ∞-groupoids [12]. The computational perspective was developed most fully by Per Martin-Löf, leading in particular to his Intuitionistic Theory of Types [23], on which the formal system of homotopy type theory is based. The connection to homotopy theory was first hinted at in the groupoid interpretation of Hofmann and Streicher [14, 13].[1] It was then made explicit by several researchers, roughly simultaneously.[2] The connection was clinched

*Thanks to Daniel Grayson, Michael Mislove, and Vladimir Voevodsky for helpful comments on an earlier draft. Of course, the authors alone are still responsible for any errors or misstatements.

[1] The importance of equality of elements of a type in constructive mathematics was also emphasized by Bishop [7]. Quotient types, which were introduced in NuPRL as a further development of Bishop's and Martin-Löf's ideas [9], may be seen as a particular case of this connection.

[2] Awodey and Warren [5] showed that the basic system of Martin-Löf type theory can be interpreted in a Quillen model category (an abstract framework for doing homotopy theory); Lumsdaine [22] and van den Berg and Garner [27] showed that every type in the system has the structure of an ω-category (a structure closely related to that of an ∞-groupoid); Gambino and Garner [11] showed that the type theory itself supports a weak factorization system (the basic building block of a Quillen model structure); and both Streicher [25] and Voevodsky [28] proposed interpretations into the category of simplicial sets, using ideas from homotopy theory.

by Voevodsky's introduction of the *univalence axiom*, which is motivated by the homotopical interpretation, and which relates type equality to homotopy equivalence [18, 4].

Constructive foundations are often regarded as incompatible with classical mathematics. By contrast, the framework of homotopy type theory is fully compatible with classical mathematics, and indeed allows for a classical conception of proposition, as well as a conception of set that is compatible with such principles as the axiom of choice. The key to achieving this unification is to avoid postulating generally certain reasoning principles, such as the decidability of every type, although these may still be postulated "locally", for example the decidability of every *proposition*. It is notable that these same reasoning principles are also those that are usually avoided in constructive foundations, opening the door to the unification of the constructive (computational) and homotopic (spatial) interpretations of types, the implications of which are only just beginning to be understood. Moreover, by not insisting on these principles globally, it is possible to consider a far richer notion of type than has previously been considered in the computational approach, namely one in which types are abstract spaces that may have non-trivial higher-dimensional structure, like the n-spheres for all $n \geq 0$. In conventional foundations, such as axiomatic set theory, these objects are presented as structured sets representing certain conceptions of space, such as topological spaces. Here, instead, such higher-dimensional objects arise "synthetically" in much the way that lines and triangles in Euclid's geometry are primitive abstract objects, rather than being comprised of analytic point-sets. This provides a new perspective on some familiar constructions in homotopy theory, such as the homotopy groups of a space [26, 21, 19] and the construction of so-called Eilenberg-MacLane spaces [20], with specified homotopy groups. Moreover, new proofs of some standard results have a distinctively "logical" flavor, in combination with more "geometric" and "topological" elements.

What is it that makes this new unification possible? Although it may be too early to formulate a single, deep unifying principle, it is possible to make a few observations that will give the reader a sense of its inevitability. First, all of the constructions of Intuitionistic Type Theory, including especially the identity type, are homotopy invariant, in the sense that type families and mappings between types inherently respect identifications (paths, homotopies, or deformations). Moreover, the formation of indexed products and sums of types, which correspond to analogous constructions on spaces, respect the homotopically motivated notion of equivelence of types,corresponding to the homotopy equivalence of spaces. This invariance

essentially follows from the basic fact that Martin-Löf's ingenious concept of the *identity type* corresponds to the path space of a space, and since everything in the formal system respects identity, everything in the interpretation respects homotopy, which is determined by identfication along paths. Second, a characteristic feature of both intuitionistic type theory and homotopy theory is an emphasis on *structure* over *property*. Under the propositions-as-types conception of intuitionistic logic,[3] types express propositions, and objects of the type are proofs of those propositions in the form of mathematical constructions that provide evidence for their truth. A similar emphasis can be discerned in abstract homotopy theory, in which, *e.g.*, paths (homotopies) may be seen as evidence for the "identification" of two points, and similarly for paths between the corresponding values of two functions. Two points are not merely "equal", as a property, but rather are identified by a (not necessarily unique) deformation, construction, or procedure. This approach extends to higher dimensions, in that one may speak of the identifications of two (parallel) identifications, at all higher dimensions. Such a structure of a hierarchy of identifications via path connectedness is found in standard settings for homotopy theory such as simplicial sets, cubical sets, and globular sets, all of which stress the role of cells as identifications.

We thus already see an analogy between the constructively motivated concept of *proof relevance*, in which proofs are mathematical objects classified by a type, and the homotopically motivated distinction between structure and property. An important advantage resulting from proof relevance is that it naturally supports a comprehensive approach to mechanized mathematics in which computer systems, such as Coq [10] and Agda [1], can be used to verify the correctness of mathematical arguments, of either a classical, set-theoretic form, or a constructive, type-theoretic form. In either case the proof of a theorem constitutes a formal mathematical object whose validity can be independently checked, avoiding the need to rely on the correctness of the proof checker itself. Once a proof has been obtained, others can not only check its formal correctness by the usual means, but can also submit the proof to another checker, to ensure that it is valid according to the rules of homotopy type theory. This approach to verification is fundamentally the same as that proposed by de Bruijn in the Automath system [3], albeit applied to a language with richer foundational commitments than were required there. It should be contrasted with the approach of systems such as NuPRL [24] or HOL [15, 16], that rely on a small trusted code base to ensure the validity of proofs.

[3] See, *e.g.*, [17] for its original formulation.

The idea of identifications of points in a space along a continuous path, and of higher identifications of paths as homotopies, etc., leads to Voevodsky's conception of a hierarchy of *homotopy levels*, or *h-levels* for short, which is definable within type theory. Whereas the usual hierarchy of size is determined by type universes or large cardinals in type theory or set theory, the hierarchy of h-levels is based instead on the internal structure of types. Roughly speaking, the lowest level consists of the types that have at most one element, up to path-connectedness; these are called *propositions*, and they correspond to the (empty or) contractible spaces. The next level, called *sets*, consists of those types whose identity types are themselves propositions — two elements of a set are "equal in at most one way". After that come the types whose identity types are sets; these are the *groupoids*. And so on, with the types at level $n+1$ being those whose identity types have level n, for all $n \geq 0$. Just the recognition that this hierarchy of h-levels is present in the system of all types has been a huge advance in our understanding of type theory; previously, it was simply a mystery that some types were fully determined by their elements, while others seemed to behave as though they had some further structure. The construction of quotient types, for example, is now greatly simplified when one knows that the equivalence relation being factored out is a family of propositions, and not of "higher-dimensional" types. For another example, for types A and B that are propositions, the relevant notion of equivalence is *logical equivalence*, represented by the type $A \leftrightarrow B$. For sets, the relevant notion is *isomorphism* $A \cong B$, and for groupoids, there is the notion of *(categorical) equivalence* $A \simeq B$. Each of these concepts results by specializing the single, uniform notion of *equivalence of types* $A \simeq B$ (also due to Voevodsky) to the respective cases of propositions, sets, and groupoids.

In this setting, Voevodsky's univalence axiom can be stated as the assertion that the type $A \simeq B$ of all equivalences between two types A and B is itself equivalent to their identity type,

$$(A \simeq B) \;\simeq\; \mathsf{Id}(A, B). \tag{UA}$$

Thus in particular, logically equivalent propositions will be identified, as in the original, extensional type theory of Church [8]. Isomorphic sets, too, will be identified "up to homotopy", *i.e.*, by paths between them in the universe of all types, and similarly for equivalent groupoids, and equivalent types in general. Note that this stipulation also serves to specify the otherwise underdetermined identity type $\mathsf{Id}(A, B)$.

This is not really the place for a systematic introduction (for that, see [26]), but a brief example may serve to convey a bit of the flavor of the new

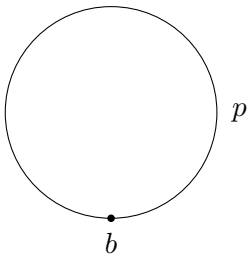

Figure 1: The 1-sphere S^1

approach, especially the distinctive intermingling of logical and homotopical ideas. As is the case in conventional Martin-Löf type theory, the basic types of booleans $\mathbb{B}$ and natural numbers $\mathbb{N}$ can have at most one identification between any two elements; that is, given say $n, m : \mathbb{N}$ and $p, q : \mathsf{Id}_{\mathbb{N}}(n, m)$ in the identity type of n and m, we always have some $\alpha : \mathsf{Id}_{(\mathsf{Id}_{\mathbb{N}}(n,m))}(p, q)$ identifying p and q. In this sense, there is no real information in the type $\mathsf{Id}_{\mathbb{N}}(n, m)$, apart from whether or not it is inhabited. Such types with at most one identification between any two elements are called "sets". Any types that can be constructed from $\mathbb{B}$, $\mathbb{N}$, or any other sets, by means of the usual type constructors of dependent sum $\Sigma_{x:A}\, B(x)$ and dependent product $\Pi_{x:A}\, B(x)$ (which include $A \times B$ and $A \to B$ as special cases) are also sets, and the same is true for the identity types $\mathsf{Id}_A(a, a')$ for $a, a' : A$, for any set A.

An example of a type that is not a set is the circle (or "1-sphere") S^1, which has a base point $b : S^1$ and a generating loop $p : \mathsf{Id}_{S^1}(b, b)$. There are then many different self-identifications, which may be labelled

$$\mathsf{refl}(b),\ p,\ p \cdot p,\ \ldots : \mathsf{Id}_{S^1}(b, b).$$

Here $\mathsf{refl}(b)$ is the trivial identification, *i.e.*, the canonical witness to the reflexivity of identity. There is also the identification p, which is different from $\mathsf{refl}(b)$ in the sense that $\mathsf{Id}_{(\mathsf{Id}_{S^1}(b,b))}(\mathsf{refl}(b), p)$ is empty. We can think of p homotopically as the continuous "path" that goes once around the circle.

By the (function witnessing the) transitivity of equality,

$$(-) \cdot (-) : \mathsf{Id}_{S^1}(a, b) \times \mathsf{Id}_{S^1}(b, c) \longrightarrow \mathsf{Id}_{S^1}(a, c),$$

there are also the "paths" $p \cdot p$, $p \cdot p \cdot p$, ... And by symmetry,

$$(-)^{-1} : \mathsf{Id}_{S^1}(a, b) \longrightarrow \mathsf{Id}_{S^1}(b, a),$$

there are similarly the paths p^{-1}, $p^{-1} \cdot p^{-1}, \ldots : \mathsf{Id}_{S^1}(b,b)$. Although S^1 is therefore not a set, it can be shown that $\mathsf{Id}_{S^1}(b,b)$ is one; that is, the types $\mathsf{Id}_{(\mathsf{Id}_{S^1}(b,b))}(x,y)$ are either inhabited (by reflexivities) or empty, depending on whether or not $x = y$, for all $x, y : \mathsf{Id}_{S^1}(b,b)$. Indeed, one can show that $\mathsf{Id}_{S^1}(b,b) \cong \mathbb{Z}$, *i.e.*, the fundamental group of the type S^1 is the integers, as it should be (see [21] for the details). The proof of this uses the univalence axiom, together with the specification of S^1 as a new kind of *higher* inductive type, generalizing the usual inductive specification of the natural numbers and similar structures. For S^1, the inductive specification essentially says that S^1 is "the type freely generated by the base point $b : S^1$ and the loop $p : \mathsf{Id}_{S^1}(b,b)$", in the same sense that the usual inductive specification of $\mathbb{N}$ says that it is the type freely generated by $0 : \mathbb{N}$ and the successor function $s : \mathbb{N} \longrightarrow \mathbb{N}$.

Another type that is not a set is the universe $\mathcal{U}$ of all (small) types. According to the Univalence Axiom, identifications between types $A, B : \mathcal{U}$ correspond to equivalences $A \simeq B$, which as we said above are generalized type isomorphisms. In fact, as already stated, if A and B themselves are sets, then an equivalence between them is just an isomorphism in the usual sense: a pair of maps back and forth that compose to the respective identity mapppings. Now the booleans $\mathbb{B}$, for example, have two different isomorphisms $\mathbb{B} \cong \mathbb{B}$, namely the identity and the operation of "negation" $\neg : \mathbb{B} \longrightarrow \mathbb{B}$, which swaps the truth values $0, 1 : \mathbb{B}$. Thus by univalence there are two distinct identifications in $\mathsf{Id}_{\mathcal{U}}(\mathbb{B}, \mathbb{B})$, corresponding to these distinct isomorphisms, and so $\mathcal{U}$ is not a set, but a "higher-dimensional" type, like S^1.

Now observe that by the basic recursive property of S^1 as "the type freely generated by a point with a loop on it", there is a map

$$\mathsf{rec}(\mathbb{B}, n) : S^1 \longrightarrow \mathcal{U},$$

determined by sending the base point $b : S^1$ to the booleans $\mathbb{B} : \mathcal{U}$ and the generating loop $p : \mathsf{Id}_{S^1}(b,b)$ to (the loop corresponding under univalence to) negation, say $n : \mathsf{Id}_{\mathcal{U}}(\mathbb{B}, \mathbb{B})$. As a type of the form $S^1 \longrightarrow \mathcal{U}$, this $\mathsf{rec}(\mathbb{B}, n)$ is thus a family of types over S^1, sometimes called a "dependent type" and written

$$x : S^1 \vdash \mathsf{rec}(\mathbb{B}, n)(x).$$

Homotopically, such a type-family is interpreted as a "fibration" $E \longrightarrow S^1$, where the total space E is just the sum type $\Sigma_{x:S^1} \mathsf{rec}(\mathbb{B}, n)(x)$, equipped with its usual indexing projection. In the present case, the "fiber" is then the type $\mathsf{rec}(\mathbb{B}, n)(b) = \mathbb{B}$, and the action on (elements of) $\mathbb{B}$ induced by the

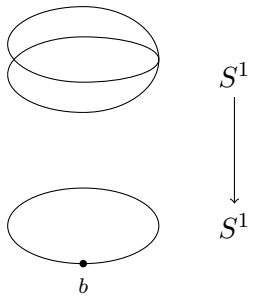

Figure 2: The twisted double cover of S^1

path $n : \mathsf{Id}_{S^1}(b, b)$ in the base is exactly the operation of negation $\neg : \mathbb{B} \longrightarrow \mathbb{B}$. Thus, from a homotopical point of view, we have constructed the "twisted double cover" of the circle (see Figure 2). This construction from homotopy theory is closely related to the celebrated Hopf fibration which, among other things, can be used to compute some of the higher homotopy groups of the spheres S^2 and S^3. Indeed, one can construct the Hopf fibration in homotopy type theory in much the same way as the foregoing example, using univalence, negation, winding around the circle, and other constructions derived from combinations of logical, type-theoretic, and homotopical ideas (see [26], §8.5).

We can now say in a bit more detail how the univalent framework of homotopy type theory subsumes and extends the classical, set-theoretic framework for doing mathematics, by making use of the hierarchy of h-levels, which includes sets within a broader framework of homotopy types. At the bottom level, the propositions (the types having at most one element, up to higher identification) correspond to conventional, proof-irrelevant propositions; whether we also assert the law of excluded middle in the form that every such proposition is either inhabited or empty is a further, consistent assumption that may be made if classical logic is desired. Next, the sets (for which equality is a proposition that is taken to be "self-evident" or "proof-irrelevant") correspond to the usual sets, but now without any commitment to choice principles, or whether membership is a boolean proposition. Those further principles can still be consistently taken as axioms if needed, but they are not required, even with the introduction of infinite sets such as the type of natural numbers. Voevodsky's new insight, which plays such an important role in homotopy type theory, is that, besides the familiar concepts of proposition (classically formalized in predicate calculus) and set (classically given by the Zermelo-Fraenkel axioms), there is an infinite hierarchy of fur-

ther dimensions extending beyond just these two. The groupoids (the next h-level above the sets), such as our example S^1, are the natural setting for systems of set-theoretic structures, such as groups and rings, that one may wish to regard as identified up to isomorphism. Because two groups, say, can be isomorphic in many different ways, however, the evidence for an identification is not a trivial proposition, but consists in the mutually inverse pair of homomorpisms, *i.e.*, the isomorphism, that warrant it. Here we see explicitly how proof-relevance (from constructivism) and the "property-structure" distinction (from homotopy theory) coincide.

In this way we can now distinguish, within Martin-Löf type theory, an infinite hierarchy of different "homotopical dimensions" that were not fully recognized previously, despite such models as Hofmann and Streicher's two-dimensional groupoid interpretation [14] that strongly hinted at the importance of higher dimensions of structure. Type theory was, of course, originally conceived as a foundation for constructive mathematics, in which all constructions, including proofs of propositions, have direct computational meaning in accordance with Brouwer's original program. This fundamental connection with computation has proved enormously influential in computer science, in particular in the theory of programming languages and the foundations of mechanized proof. Homotopy type theory makes essential use of the concept of proof relevance, which is so central to the constructive program, and emphasizes a notion of abstract types that is familiar from the theory of programming languages (*e.g.*, the identity type is itself an abstraction, rather than being encoded in terms of a concrete definition of homotopy). The grand challenge as of this writing is to extend the computational interpretation to the univalence axiom, and therewith to the full hierarchy of h-levels, providing a computational meaning for, say, mappings among higher-dimensional structures such as the spheres and toruses of arbitrary dimension. Recent advances, such as the landmark development of a constructively valid model using cubical sets [6], strongly suggest that such a unification will be achieved in the near future. The potential implications for computer science are only beginning to be explored [2].

Perhaps the most important application of the unification of classical and constructive mathematics is the possibility of applying systems of mechanized proof verification to broad swaths of classical mathematics that were previously formalizable only via elaborate coding into set theory, and only in systems based on classical logic, which generally lack the benefits resulting from the computational interpretation of constructive systems (*e.g.*, the generation of independently verifiable proof certificates). The direct formalization of everything from quotient sets to cohomology simplifies and

streamlines the formalization of even advanced mathematics, and has the potential to eventually make formal verification into a practical tool for the everyday mathematician. Interestingly, this practical development makes the logical foundations of mathematics finally relevant to the actual practice of mathematics, rather than being just a theoretical possibility. The result may be a new "post-Gödel" attitude toward foundations; for when their only interest was theoretical, the phenomenon of incompleteness seemed to lessen the importance of logical foundations in principle. But with the actual practical benefits of formalization (increased rigor and certainty, ease of remote collaboration, accumulation of results), the theoretical incompleteness phenomenon diminishes in importance, and logical foundations can become a useful addition to the toolbox of the working mathematician.[4]

It is a curious fact, made all the more interesting by the above-mentioned developments, that two of the most successful systems for mechanized proof, NuPRL [9] and Coq [10], are both based on constructive type theory. Why ought that be the case? Homotopy type theory may provide a clue in the importance of proof-relevance, and the associated distinction between property and structure, in both constructive mathematics and homotopy theory. The univalent approach of homotopy type theory exploits the axiomatic freedom provided by constructive mathematics, allowing it to rely far less on elaborate encodings which impede the process of formalization required to admit machine-checked proof. This experience parallels the development of high-level (abstract) programming languages that provide a synthetic concept of computation, rather than one based on low-level machine models such as the Turing machine or Random-Access Machine. Thus we find that whether we are discussing mechanized mathematical proof or verified computer programming, Church's λ-calculus emerges as a central concept. Perhaps this explains why constructive mathematics and mechanized proof are so tightly linked. That they should also be entwined with homotopy theory — one of the most abstract, geometrical, and rarified areas of modern mathematics — is an intriguing and challenging fact inviting further investigation.

References

[1] The Agda proof and programming system. `http://wiki.portal.chalmers.se/agda/pmwiki.php`.

[4] See the code section of `http://homotopytypetheory.org` for links to shared repositories of mechanized mathematics in homotopy type theory using the Coq and Agda systems. Readers are invited to contribute!

[2] Carlo Angiuli, Edward Morehouse, Daniel R. Licata, and Robert Harper. Homotopical patch theory. In *Proceedings of the 19th ACM SIGPLAN international conference on Functional programming, Gothenburg, Sweden, September 1-3, 2014*, pages 243–256, 2014.

[3] The Automath archive. `http://www.win.tue.nl/automath`.

[4] S. Awodey, A. Pelayo, and M.A. Warren. Voevodsky's univalence uxiom in homotopy type theory. *Notices of the Amer. Mathem. Soc.*, 60(08):1164–1167, 2013.

[5] S. Awodey and M.A. Warren. Homotopy-theoretic models of identity types. *Math. Proc. Camb. Phil. Soc.*, 146:45–55, 2009.

[6] Marc Bezem, Thierry Coquand, and Simon Huber. A model of type theory in cubical sets. (Unpublished Manuscript), March 2014.

[7] Errett Bishop and Douglas Bridges. *Constructive Analysis*. Number 279 in Grundlehren der mathematischen Wissenschaften: A Series of Comprehensive Studies in Mathematics. Springer, 1985. (Revision of first edition by Bishop published in 1967.).

[8] A. Church. A formulation of the simple theory of types. *Journal of Symbolic Logic*, 5:56–68, 1940.

[9] Robert L. Constable, *et al. Implementing Mathematics with the NuPRL Proof Development System*. Prentice Hall, 1985.

[10] The Coq Proof Assistant. `http://coq.inria.fr`.

[11] Nicola Gambino and Richard Garner. The identity type weak factorisation system. *Theoret. Comput. Sci.*, 409(1):94–109, 2008.

[12] A. Grothendieck. Pursuing stacks. Unpublished manuscript, 1983.

[13] M. Hofmann and T. Streicher. The groupoid model of type theory. In G. Sambin and J. Smith, editors, *Twenty-five years of constructive type theory*. Oxford University Press, 1995.

[14] Martin Hofmann and Thomas Streicher. The groupoid model refutes uniqueness of identity proofs. In *LICS*, pages 208–212, 1994.

[15] The HOL system. `http://www.cl.cam.ac.uk/research/hvg/HOL`.

[16] The HOL Light theorem prover. `http://www.cl.cam.ac.uk/~jrh13/hol-light/index.html`.

[17] W. A. Howard. The formulae-as-types notion of construction. In *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism*, pages 479–491. Academic Press, 1980.

[18] C. Kapulkin, P. LeFanu Lumsdaine, and V. Voevodsky. Univalence in simplicial sets. *arXiv*, 1203.2553, 2012.

[19] Daniel R. Licata and Guillaume Brunerie. $\Pi_n(S^n)$ in homotopy type theory. In *Certified Programs and Proofs - Third International Conference, CPP 2013, Melbourne, VIC, Australia, December 11-13, 2013, Proceedings*, pages 1–16, 2013.

[20] Daniel R. Licata and Eric Finster. Eilenberg-maclane spaces in homotopy type theory. In *Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), CSL-LICS '14, Vienna, Austria, July 14 - 18, 2014*, page 66, 2014.

[21] Daniel R. Licata and Michael Shulman. Calculating the fundamental group of the circle in homotopy type theory. In *Proceedings of the 2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '13, pages 223–232, Washington, DC, USA, 2013. IEEE Computer Society.

[22] P. Lumsdaine. Weak ω-categories from intensional type theory. In P.-L. Curien, editor, *Typed Lambda Calculi and Applications*, number 5608 in LNCS, pages 172–187. Springer, 2009.

[23] Per Martin-Lof. *Intuitionistic type theory*, volume 17. Bibliopolis, Naples, 1984. Notes by Giovanni Sambin.

[24] The NuPRL proof development system. `http://nuprl.org`.

[25] T. Streicher. Identity types vs. weak omega-groupoids: some ideas, some problems. Talk given in Uppsala at the meeting on "Identity Types: Topological and Categorical Structure", 2006.

[26] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. `http://homotopytypetheory.org/book`, Institute for Advanced Study, 2013.

[27] B. van den Berg and R. Garner. Types are weak ω-groupoids. *Proceedings of the London Mathematical Society*, 102(3):370–394, 2011.

[28] V. Voevodsky. A very short note on the homotopy λ-calculus. Unpublished note, 2006.