Classe di Lettere e Filosofia

Corso di perfezionamento in Filosofia

XXXIV ciclo

Extensionality and Predicativity in Type Theory

Settore Scientifico Disciplinare **M-Fil/02**

Candidato:
Dr. Michele Contente

Relatore:
Prof. Mario Piazza

Anno accademico 2022–2023

Ai miei nonni

Contents

Contents **4**

Introduction **6**

1 Martin-Löf Type Theories **22**
1.1 Judgements and Types 22
1.2 Type Constructors 26
1.2.1 Varieties of Inductive Definitions 35
1.2.2 Universes 36
1.3 Equality in Type Theory 41
1.4 Meta-theoretical Properties 46
1.5 Variants of **MLTT** 49
1.6 Historical Notes 55

2 Homotopy Type Theory and the Univalence Axiom **64**
2.1 The true nature of Identity 64
2.2 The Homotopical Perspective 68
2.3 Homotopy Levels & HIT 71
2.4 Homotopies & Equivalences 78
2.5 The Univalence Axiom 81

3 Invariance under Equivalence **84**
3.1 The Principle of Invariance 85
3.2 The Proof-Relevant Relations Model 88
3.3 Invariance under Equivalence 108
3.4 Discussion 113

4 MF & the Compatibility with HoTT **118**
4.1 Motivations for **MF** . . . 118
4.2 The system **MF** . . . 123
4.3 The compatibility of **MF** with **HoTT** . . . 129
4.3.1 Preliminaries . . . 130
4.3.2 The interpretation of **mTT** . . . 135
4.3.3 An alternative interpretation of **mTT** . . . 147
4.3.4 Canonical Isomorphisms . . . 151
4.3.5 The H-category of h-sets . . . 169
4.3.6 The interpretation of **emTT** . . . 172
4.4 Discussion . . . 184
4.4.1 A Different Interpretation . . . 184
4.4.2 True Judgements and Realizability . . . 186
4.4.3 A comparison with other interpretations . . . 188

5 The Constructive Conception of Predicativity **191**
5.1 The problem of impredicativity . . . 192
5.2 Weyl's Classical Predicativity . . . 196
5.2.1 Interpretations of *Das Kontinuum* . . . 197
5.2.2 A Forgotten Interpretation of *Das Kontinuum* . . . 205
5.3 The Iteration of the Logico-Mathematical Process . . . 221
5.4 Constructive Predicativity . . . 236
5.4.1 Inductive Definitions and $\mathcal{W}$-types . . . 242
5.4.2 On the Method of Meaning Explanations . . . 245
5.4.3 Generalised Predicativity and its Limits . . . 256

Conclusion **266**

Bibliography **267**

Introduction

Intuitionism and the BHK explanation

In 1907, a young Dutch mathematician, Luitzig Egbertus Jan Brouwer, presented some radical and controversial theses about the nature of mathematics and its relationship with logic and language. Mathematics, Brouwer argued, is primarily an activity: the mathematician's acts are *mental constructions*. The primordial input for mathematical activity is the intuition of the flow of time. All mathematical constructions ultimately result from the unfolding and the iteration of this basic intuition.

According to Brouwer, the fundamental intuition as well as the derived constructions are pre-linguistic. We can certainly produce linguistic descriptions of mathematical constructions, but these constructions do not depend on language and also their properties are independent of any linguistic fact. Mathematics is a language-less activity.

Logic is not prior to mathematics in any sense of the word. When we try to give a linguistic form to our mathematical constructions, then we can observe some regularities or recurrent patterns in our linguistic recordings. The aim of logic is to individuate such patterns and provide a more systematic description of them. In particular, it aims to classify valid inferences which govern the relationship between statements about constructions.

Brouwer drew some consequences of his views about logic and mathematics in [19, 8]. On his conception, the law of the excluded middle (LEM) turns out to be not generally valid. Assuming the validity of LEM amounts to asserting that there is a method that for any formula φ either gives us a construction for φ or shows that such a construction cannot be realized. Then, Brouwer argued, every undecided mathematical statement yields a counterexample to LEM. The belief in the universal validity of this principle is grounded on an illegitimate extension of reasoning over finite domains to infinite ones. Indeed,

Brouwer acknowledged that LEM is valid over finite domains and possibly in other cases as well; what he denied was the universal validity of such a principle.
Brouwer did not articulate a comprehensive theory of the meaning of logical operators. Nor did he give an axiomatization of the logic underlying his intuitionistic mathematics. It was his student Arend Heyting which formulated the first axiomatic systems for intuitionistic logic and mathematics [59, 60, 61]. In particular, the works [62, 63] contain the first discussion of the meaning of intuitionistic logical constants. The truth of a statement φ is to be explained in terms of proofs or constructions:

> "Here, then, is the *Brouwerian assertion* of p: it is known how to prove p." ([62], p.959)

In some later publications, Heyting developed this basic idea and clarified the intuitionistic reading of logical constants in terms of constructions[1]
A similar explanation of the meaning of logical constants in intuitionistic logic was elaborated by Kolmogorov around the same time [76]. For Kolmogorov, intuitionistic logic could be regarded as a logic of *problems*(propositions) where the constructions represent the *solutions* to such problems.
These ideas on the meaning of logical constants gradually evolved into the 'standard' explanation of intuitionistic logic, which is known under the name *BHK interpretation or explanation*[2]:

(1) There is no proof of $\bot$.

(2) A proof of $A \wedge B$ consists of a proof of A and a proof of B.

(3) A proof of $A \vee B$ consists of a proof of A or a proof of B.

(4) A proof of $A \to B$ consists of a method which transforms any proof of A into a proof of B.

[1]The writings of this period shown how Heyting was strongly influenced by Husserlian phenomenology and, in particular, by Husserl's pupil Oscar Becker.

[2]Sundholm has argued in [149] that the name 'explanation' should be preferred to 'interpretation': first, because BHK is not an interpretation in the technical sense which is established in mathematical logic; second, because, in this context, the formulas are taken to be *meaningful* propositions and the clauses are meant to elucidate this intended meaning.

(5) A proof of $\exists x.A(x)$ consists of an individual a in the relevant domain D and a proof of $A(a)$.

(6) A proof of $\forall x.A(x)$ consists of a method which transforms a proof of $a \in D$ into a proof of $A(a)$.

Since in intuitionistic logic $\neg A \ :\equiv \ A \to \bot$, then the meaning of negations amounts to say that there is a method which transforms any proof of A into a proof of $\bot$.
Notwithstanding the informal nature of these clauses, the BHK explanation is sufficient to rule out logical principles such LEM and double negation elimination that are not intuitionistically valid. On the other hand, it is more doubtful that this explanation univocally determines the meaning of logical constants and, as consequence, many attempts have been made to make it more precise.

Computational Trinitarianism

One possible way of making BHK more precise is the *Curry-Howard Correspondence*, which turned out to be one of the most deep and fruitful perspectives on the relationship between logic and computation. The intuitionistic explanation of logical constants strongly suggests a 'proofs-first' view of logic: proofs or constructions are what determine the meaning of a proposition and they are first-class citizens in intuitionistic mathematics. Then, it is quite natural to hold that an adequate language for intuitionistic mathematics is one in which proofs are explicitly presented and which allows to reason about them.
Let us consider the standard natural deduction rules for conjunction and implication:

$$\frac{A \qquad B}{A \wedge B}\,(\wedge\text{-I}) \qquad \frac{A \wedge B}{A}\,(\wedge\text{-E}_1) \qquad \frac{A \wedge B}{B}\,(\wedge\text{-E}_2)$$

$$\frac{\begin{array}{c}[A] \\ \vdots \\ B\end{array}}{A \supset B}\,(\supset\text{-I}) \qquad \frac{A \supset B \qquad A}{B}\,(\supset\text{-E})$$

Then, keeping in mind BHK-clauses for $\wedge$ and $\supset$, we can annotate the previous rules with proof-terms standing for constructions:

$$\frac{a : A \qquad b : B}{(a, b) : A \wedge B}\,(\wedge\text{-I})$$

$$\frac{c : A \wedge B}{\mathsf{pr}_1(c) : A}\,(\wedge\text{-E}_1) \qquad \frac{A \wedge B}{\mathsf{pr}_2(c) : B}\,(\wedge\text{-E}_2)$$

$$\frac{\begin{array}{c}[x : A] \\ \vdots \\ b : B\end{array}}{\lambda x.b : A \supset B}\,(\supset\text{-I}) \qquad \frac{f : A \supset B \qquad a : A}{f(a) : B}\,(\supset\text{-E})$$

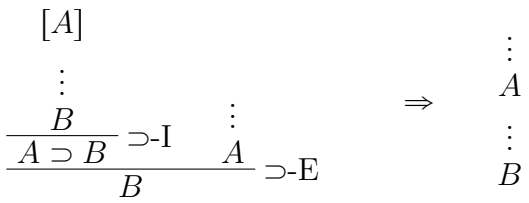
Not only these rules are immediately justified under the BHK explanation, but we can further observe that they correspond to the rules of the simply typed λ-calculus with product types $\times$ and arrow types $\rightarrow$. As a result, proofs turn out to correspond to *programs* in a very specific sense. More generally, the following correspondence is established:

$$\begin{array}{rcl} \text{Propositions} & \sim & \text{Types} \\ \text{Proofs} & \sim & \text{Programs} \\ A \wedge B & \sim & A \times B \\ A \supset B & \sim & A \rightarrow B \end{array}$$

In a natural deduction system, a derivation may contain some steps produced by certain combinations of the application of the rules that can be eliminated. Such steps are called *detours* and are formed when the application of the elimination rule for a logical constant $*$ immediately follows the application of the introduction rule for $*$. The *reduction* rules are those procedures that specify how to eliminate such detours. In the case of $\wedge$ and $\supset$ they are the following:

$$\dfrac{\dfrac{\begin{array}{cc}\vdots & \vdots \\ A & B\end{array}}{A \wedge B}\,\wedge\text{-I}}{A}\,\wedge\text{-E}_1 \qquad \Rightarrow \qquad \begin{array}{c}\vdots \\ A\end{array}$$

and the case where the last inference is ($\wedge$-E$_2$) is analogous. For $\supset$ we have



The figure on the left is a *redex*, while that on the right is a *reduct*. A reduction procedure provides a way to eliminate redexes. These rules are fundamental for the *normalization* theorem, stating that every proof can be reduced to a proof in *normal form*, that is, without redexes. Now the interesting fact is that these reduction rules formulated in the λ-calculus yield the following equations:

$$\begin{aligned}
&\mathsf{pr}_1((a,b)) \equiv a \\
&\mathsf{pr}_2((a,b)) \equiv b \\
&(\lambda x.b)(a) \equiv b[a/x]
\end{aligned}$$

Reading such equations from left-to-right amounts to say that the program on the left *evaluates* to the program on the right. Therefore, the correspondence is extended as follows:

$$\text{reduction} \ \sim \ \text{evaluation}$$

Finally, it must be remarked that the correspondence can be extended to the other logical constants, establishing a strong link between intuitionistic logic and a typed functional programming language. Under this perspective, it is clearer why intuitionistic proofs have computational content.
On a more abstract level, it is possible to regard types/propositions as *objects* and proof-terms/programs as *morphisms*. In particular, for any object A, there is a morphism $\mathbf{1}_A : A \to A$, which is the identity, and for any proof of B from A and any proof of C from B, we can form a proof of C from A by defining an operation of composition between morphisms. Such data are required to satisfy some simple axioms. In this way, we are led to introduce the definition of a category:

Definition 0.0.1. A category $\mathcal{C}$ consists of the following data:

- Objects: $A, B, C \ldots$
- Morphisms: $f, g, h \ldots$
- For each morphism f, there are two objects $\mathsf{dom}(f)$ and $\mathsf{cod}(f)$. We will write $f : A \to B$, if $\mathsf{dom}(f) = A$ and $\mathsf{cod}(f) = B$.
- For each object A, there is a morphism $\mathbf{1}_A : A \to A$.
- Given morphisms $f : A \to B$ and $g : B \to C$, there is a morphism $g \circ f : A \to C$.
- These data satisfy the following axioms: for any $f : A \to B$,

$$f \circ \mathbf{1}_A = f = \mathbf{1}_B \circ f$$

for all $f : A \to B$, $g : B \to C$, $h : C \to D$,

$$h \circ (g \circ f) = (h \circ g) \circ f.$$

Now simply typed λ-calculus with product and arrow types can be regarded as a concrete *presentation* of a freely generated structure. The structure presented by this system is that of a *cartesian closed category*, that is a category having all products and exponentials. Moreover, a cartesian closed category satisfies some equations which correspond to the equations derived from the reduction procedures in the underlying syntactic presentation. Therefore, there are three different perspectives on the same object connecting logic, programming and categories. The single object is the notion of *computation*, that manifests in these three apparently different fields which are instead deeply related. Making enter categories into the picture, we arrive at what is known as *Curry-Howard-Lambek Correspondence*. We will get to see that the correspondence is much more far-reaching than what has been discussed so far and can be extended to more sophisticated systems and structures.

Intuitionistic Type Theory

Type theory as a foundation of mathematics was first introduced by Russell [134]. It grew out of the discussion about antinomies in Frege's system and in naive set theory. The fundamental point of the *doctrine of types* consisted in

the fact that every propositional function has a type as its domain. The type is the domain containing those elements applied to which the propositional function yields *meaningful* propositions. Paradoxical properties are ruled out as meaningless.

A more modern reformulation of Russell's simple theory of types was given by Church [21] and was based on his λ-calculus: the ground types are the types of individuals ι and the type of propositions o and, moreover, types are closed under functions, that is, if A and B are types, then there is a type $A \to B$. The λ-calculus notation allowed to give a primitive notion of functions, that are not defined anymore as a particular kind of relations. A unary predicate is represented as a (propositional) function with type $\iota \to o$. In the subsequent years, many reformulations of this system were proposed. However, the possibility of presenting type theory as a typed λ-calculus suggested that the system, first proposed as a foundational framework, may have a relationship with the notion of computation. The discovery of the Curry-Howard Correspondence [34, 35, 68] brought to light such a relationship and demonstrated its importance.

The decisive step was undertaken by Per Martin-Löf at the beginning of the seventies. He introduced a system, called *Intuitionistic Type Theory*, that could be regarded, at the same time, as a foundation for mathematics *and* a rich functional programming language. More generally, around the same period, the exploration of the propositions-as-types correspondence led also to the introduction of different type theories. What distinguished Intuitionistic Type Theory was the fact that it was explicitly presented as a comprehensive foundational system with a rich type structure allowing the formalisation of a great variety of mathematical notions.

The theory is based on the λ-calculus notation and supports *dependent* types and several different inductively defined types. A dependent type $B(x)\ [x : A]$ can be thought of as an indexed family of types. In this way, terms enter into the definition of types. Type theory with dependent types extend first-order logic, since the domains on which such families vary are richer and new ones can be built by means of the inference rules for type constructors.

In this theory, propositions and types are *identified*. It follows that logical rules are not given separately from those for types. Logical rules become particular cases of general operations over types. At least in the way the theory has been conceived by Martin-Löf, the identification between propositions and types is absolutely fundamental and is the cornerstone of the whole system. Nevertheless, we will have chance to see later that there are other possible

ways of conceiving the relation between propositions and types in dependently typed settings.

The type structure of Intuitionistic Type Theory is significantly richer than Russell's original system and the theory is intuitionistic. However, it is still consistent to extend the theory with classical principles, which means that anti-classical axioms of intuitionistic mathematics or of other schools of constructive mathematics are not derivable within the system.

Usually, the theory is presented in a natural-deduction format. A fundamental distinction underlying the system is that between *judgements* and *propositions*. According to Martin-Löf [100], these two notions must be carefully distinguished in order to give a correct philosophical account of logic. A judgement is an act of knowledge: e.g., when we say that a given proposition is true, we are making a judgement. The premises and the conclusion of an inference or of a chain of inferences, like a derivation, are judgements. Further, judgements cannot be combined by means of the logical operations. What we may combine by means of logical constants are propositions. A proposition is something that can be held to be true and hence enter into a judgement. A construction is a proof-object for a proposition and must not be confused with the demonstration [100] of a judgement, which is a tree-like object. However, in carrying out the derivation, we build a concrete proof-term for the proposition occurring in the judgement standing for the conclusion. In this sense, Intuitionistic Type Theory is also a theory of (typed) constructions, which makes more precise the insight underlying the BHK explanation. On the other hand, the possibility of introducing more complex types can be regarded as implying a generalization of BHK to more sophisticated mathematical constructions.

The correspondence between types and categories extends to the richer setting of dependent type theory and is a lively research topic. The *extensional* version of type theory (without universes) corresponds to the internal language of *locally* cartesian closed categories, modulo some coherence issues. On the other hand, intensional type theory should correspond to the internal language of certain higher-categories. In particular, the intensional theory with the *Univalence* Axiom [155] should provide an internal language for 'elementary ∞-toposes'.

Extensionality

In the traditional set-theoretical frameworks, a thorough extensional point of view is adopted from the very beginning. The axiom of extensionality asserts that two sets are equal if they have the same elements. Further, the usual set-theoretic definition of functions automatically makes valid the statement for which two functions are equal if they yield the same values for all arguments. Extensionality principles are meant to rule out any intensional matter, e.g. as a set has been defined or as the values of a function have been computed. A first justification of extensionality relies on a realist view of the mathematical universe: mathematical objects exist independently of us and their properties do not depend on human cognitive activity. However, extensional principles may be justified also on a weaker ground, that is not explicitly committed to realist metaphysical theses about mathematical objects. For instance, a possible defense may appeal to the practice: mathematics is developed in an extensional way and is not primarily concerned with intensional issues. Such issues usually do not show up in mathematics, and if they do, then they are largely held to be secondary.

In the previous sections, the link existing between proofs and programs has been emphasized. It can be argued that objects like proofs and algorithms are crucially shaped by the way they are presented: e.g. the way we perform a certain operation or we give a certain definition are epistemically fundamental in these cases. Then, they can be regarded as being primarily intensional objects. This conclusion fits well with the general intuitionistic point of view: not only realist accounts of mathematics are to be rejected, but also the claim that set-theoretical mathematics can be identified with mathematics *tout court*. Intuitionistic mathematics regards constructions and assertions about constructions as fundamental objects and, in the light of the previous discussion, it follows that computational aspects are fundamental as well. Then, it is natural to hold a language to be adequate for intuitionistic (and, more in general, constructive) mathematics whenever it allows to deal with proofs/programs directly. Set-theoretic language does not seem to be adequate in this sense.

On the contrary, type theory provides a setting where constructions and computation are regarded to be essential aspects and where we can handle constructions and make assertions about them. Better, type theory gives a privileged point of view about the relationship between intensional and extensional features. Even if we are not committed to a purely intensional

standpoint, type theory provides a framework to investigate various forms of extensionality and their properties. In an extensional framework like set-theory, intensional aspects are ruled out and they show up only in very specific cases. In this sense, type theory is much more flexible: we can state different extensionality principles and reflect on their status preserving relevant conceptual distinctions. An instructive example is the distinction between functions understood as functional relations and functions understood as rules or operations. As observed above, functions in set theory turn out to be automatically extensional. In the basic intensional version of type theory, the axiom of function extensionality, that is, two functions are equal if their values are equal on every argument, is not derivable and a type-theoretic function f, an element of the type $A \to B$, can be soundly interpreted as a rule. Even if f and g have the same values at all arguments, they cannot be identified unless they are equal *qua* rules. As I hinted above, there is also an extensional version of type theory. The main difference with the basic intensional theory concerns the rules for the identity types, that are the types of equalities associated to any type. The rules for extensional identity types are such that they abolish the distinction with the syntactical judgemental equality. In this way, we cannot have access to an intensional notion of equality in the theory and extensional principles like function extensionality are derivable.

I just hint at the fact that the extensional identity types block the *homotopical* interpretation of type theory, since the relevant higher-dimensional structure is generated by intensional identity types. The work on homotopy type theory and the Univalent Foundations provides a further perspective on the interface between intensionality and extensionality in type theory. Even if the homotopical interpretation relies on the intensional version of identity types, the Univalence Axiom can be regarded as a very general and far-reaching extensionality axiom, which implies the function extensionality principle, as first shown by Voevodsky, and other extensional principles as well. However, the resulting theory is not extensional in the strong sense mentioned above. Rather, it can be shown that Univalence is inconsistent with the assumption of certain principles characterising extensional type theory. Among the consequences of Univalence, there is the statement that isomorphic sets are equal. This is a striking consequence from the standard set-theoretical perspective: in set theory, it is easy to define properties involving references to the primitive membership relation which allow to distinguish between isomorphic sets.

I think that this brief and informal account is already sufficient to vindicate the claim that type theory provides an excellent perspective to investigate the relationship between intensional and extensional aspects. In particular, from a foundational perspective, this fact seems to be of independent interest, whatever one's views about the matter might be.

Predicativity

Type theory, as presented by Russell and Whitehead in [135], assumed that entities are not only subdivided in types, but also in *orders*. The resulting system is known as the *ramified* theory of types. Very roughly, there is a hierarchy of types where the type 0 is the type of individuals, the type 1 is the type of properties of individuals and so on. Then, orders are assigned to types to measure the logical complexity of the definition of a given property: e.g., a first order property is defined without referring to the totality of all properties, while a second order property is defined by referring *only* to first order ones. The motivations behind this further stratification are to be found in the discussion between Russell and Poincaré about the nature of paradoxes. Roughly speaking, they identified the element common to all these paradoxes in allowing viciously circular definitions. Such circularity is not always manifest, but it amounts to the fact that a given entity is defined by referring to the collection to which it belongs. A definition of this kind is *impredicative* and must be avoided in mathematics. Such a position was quite radical. Around the same years, Zermelo's work on the axiomatization of set theory included a different diagnosis of antinomies and a defense of the legitimacy of impredicative definitions. Zermelo argued that definitions of this kind pervade mathematics and cannot be sacrificed.

The ban on impredicative definitions seems to be connected with a rejection of a realist account of mathematics. On a realist perspective, there is no problem with impredicative definitions, because a definition is not understood as having a constitutive character.

Poincaré also maintained that another erroneous view leading to antinomies was that of considering infinite collections as completed or closed totalities. A *predicative* account must explain the construction of a given collection as a process structured in stages: at each stage we can refer only to those objects that have been already constructed. The ramified hierarchy essentially gives a way to formally substantiate this idea.

Under this view, a set-theoretic operation like taking the power set of $\mathbb{N}$ seems especially dubious: it is not possible to describe a rule that generate all the elements of $\mathcal{P}(\mathbb{N})$ and hence it is difficult to make sense of this operation as a step-by-step construction process.
The complexities of the ramified theory of types are well-known and developing mathematics within it can be quite painful. Such difficulties led Russell and Whitehead to introduce the controversial Axiom of Reducibility.
Turning to dependent type theory, it must be recalled that the first version proposed by Martin-Löf [96] was impredicative. It contained a type of all types V with the axiom $V : V$ and allowed for the encoding of many type constructors in a uniform way. However, Jean-Yves Girard was able to show that systems with similar assumptions were inconsistent [55]. Indeed, the later versions of type theory do not include impredicative axioms and are usually counted as predicative.
As we will see, the notion of predicativity embodied in Intuitionistic Type Theory is quite generous compared to the traditional characterizations of predicativity occurring in the literature. This point has been clearly expressed by Erik Palmgren in [115]:

> "The ordinal Γ_0 is usually called the Feferman-Schütte bound for predicativity. The *proof-theorist's notion of predicativity* is based on the idea that an ordinal is predicative if it can be reached by a certain autonomous progression of theories starting from Peano arithmetic. This is to be contrasted with what we could call the *constructivist's notion of predicativity*, which recognises a construction as predicative if it has a clear inductive structure."

The strength of the constructive notion of predicativity is essentially due to the fact that certain inductive constructions, which are usually not counted as predicative in a classical set-theoretical framework, are held to be predicatively justified in this context. In the case of type theory, the predicative status of these constructions appears to be strictly related to the constructive understanding of logical operations. Constructive theories satisfying this strong form of predicativity are called *generalized* predicative.
I think that the notion of generalized predicativity stands for further philosophical analysis: as far as I know, in the literature there are only vague insights about the justification of this kind of predicativity, but there are

no detailed discussions [3]. Whether or not this conception of predicativity is tenable, I think that a point similar to what I made about extensionality can be made about predicativity if analyzed from a type-theoretic perspective: different layers of predicativity can be carefully distinguished and the whole relation between predicative and impredicative aspects can be investigated in a more fruitful manner. Once again, meaningful distinctions are brought to light that can be appreciated independently of any preconceived philosophical and foundational standpoint.

Overview

The present work consists of five chapters. The first chapter is an introduction to basic (intensional) Martin-Löf Type Theory and to the main variants of it. Indeed, as the title of the chapter suggests, there is no single formulation of this theory in the literature. Rather, there is a family of formal systems that can be considered as variations over a basic core theory. I explain the basic concepts of the system and then I give the rules for the main type constructors. The exposition of the rules is enriched with some comments whose aim is to clarify the meaning of these rules or to point out some relevant consequences of them. Special attention is paid to the different notions of equality in type theory: the difference and the relationship between propositional and judgemental equality are discussed. Further, the main meta-theoretical properties of type theory are informally reviewed and the main variants of type theory occurring in the literature are briefly discussed. Finally, the chapter ends with some historical notes aimed at shedding light on the context in which Martin-Löf worked out his theory.
The second chapter provides an introduction to Homotopy Type Theory and to Univalent Foundations, which is a new proposal in the foundations of mathematics and is widely acknowledged as the main recent novelty in the field of type theory, since it has brought to light rich connections with higher categories and abstract homotopy theory. I review some basic facts about the homotopical perspective on type theory, that springs from the interpretation of intensional identity types. Then, I explain two important innovations introduced in the context of homotopy type theory: the stratification of types in homotopy levels and the notion of higher inductive types, that are inductive types including also rules for constructing elements of the associated

[3] However, for recent work that seems to be heading in this direction, see [31, 30].

identity types. In particular, we will see that some examples of higher inductive types have important consequences on the representation of logical notions within the system, providing an alternative to the propositions-as-types interpretation of logic. Then, I discuss the concepts of homotopy and equivalences as formalized in homotopy type theory in order to present the Univalence Axiom. The last section is devoted to this principle and briefly recalls its main consequences.

The third chapter is mainly a focus on the meaning of the Univalence Axiom and is the first original contribution of this work. I critically discuss Makkai's Principle of Isomorphism and an argument by Steve Awodey meant to justify Univalence as internalization of the Principle of Invariance under equivalence, which can be understood as a generalization of Makkai's principle. According to Awodey's argument, type theory provides a language in which all definable properties are invariant under equivalence and hence naturally satisfy the Principle of Invariance. In order to substantiate formally Awodey's philosophical analysis, I build a syntactic model of type theory (without universes) that can be regarded as the syntactical counterpart of the inverse span diagram model built in [71, 72]. The idea essentially amounts to defining some (proof-relevant) relations over the type constructors and interpret each type A as a triple consisting of a pair of types A_0, A_1 and a relation over them. In this sense, the interpretation gives rise to a translation between type theories, since the model is an extension of type theory with 'relational' type constructors. Then, a refinement of this model is applied to prove the Principle of Invariance: the relation is assumed to be a one-to-one correspondence and hence an equivalence. The invariance result basically follows from the soundness of this refined interpretation. Finally, I discuss the relation of this interpretation with Kapulkin and Lumsdaine's work and some possible extensions and applications of the model [4].

The fourth chapter presents a two-level type-theoretical foundations called Minimalist Foundations, that has been introduced by Maietti and Sambin in [92] and has been fully developed by Maietti in [89]. The 'minimalist' character of the theory is motivated by the need for having a foundational framework that can serve as common basis between the most relevant existing foundations. The first part of the chapter analyzes the motivations behind

[4] I would like to thank one of the referees for bringing to my attention that a similar question has been independently investigated in a recent paper by Bocquet [17], albeit using different methods and within a different context.

the introduction of the system and the conditions that a theory must satisfy in order to be counted as a minimalist framework in the sense of Maietti-Sambin. Then, I proceed by explaining the two-levels of the system, the intensional one **mTT** and the extensional one **emTT** which are related by a suitable quotient model construction, and their main features. The rest of the chapter is devoted to a proof of the compatibility of **MF** with **HoTT**, which is an original contribution of the present work[5]. First, I define two possible interpretations of the intensional level **mTT** and show that they are both sound. Moreover, these interpretations strictly preserves judgemental equalities and the distinction between **MF**-sorts. Then, I discuss the main difficulties in designing an interpretation of the extensional level **emTT** in **HoTT**, which are related to the presence of the extensional rules for propositional equality. In order to overcome these problems, I define the notion of canonical isomorphisms, which were already introduced in [89] and appeared in a similar form also in [65]. I give detailed proofs of the main properties of this class of isomorphisms and then I show how an interpretation of **emTT** in **HoTT** can be defined by relying on them. Finally, I prove a validity theorem for the interpretation and discuss the relation with other works and some possible developments.

The last chapter is devoted to the analysis of different conceptions of predicativity and, in particular, of generalized predicativity. After a brief historical introduction on the problem of impredicative definitions, I focus on Weyl's predicative mathematics as developed in *Das Kontinuum*. I review the philosophical ideas underlying Weyl's work and the main interpretations of it that can be found in the literature. Then, I give a detailed reconstruction of a neglected interpretation of *Das Kontinuum*, which has been proposed by Ettore Casari in [20]. I analyze the merits of this interpretation as well as some of his problems and connect it with the other interpretations of Weyl's work that have been provided by important logicians such as Feferman and Avron. I think that it is interesting to scrutinize Casari's reading of *Das Kontinuum*, because it is likely that it is the first formalization of Weyl's framework, but it is scarcely known. Casari's proposal is mainly grounded on the description made by Weyl of the logical and the mathematical processes, which provide a philosophical framework meant to justify a predicative and genetic view of the mathematical universe. I propose a more updated reading of this generative processes and present a system of (ramified) type

[5]This is based on a joint work with Milly Maietti.

theory which can be considered as the result of the formalization of Weyl's logico-mathematical process. The system is called $\mathfrak{LM}$ and turns out to be quite similar to the theory **IRTT** exposed by Palmgren in [117]. Among the original contributions of the chapter, there are an interpretation of $\mathfrak{LM}$ in **HoTT** and the investigation of the relation of a sub-system of $\mathfrak{LM}$ with the theory $\mathbf{ACA}_0$, which is well-known from Reverse Mathematics. The rest of the chapter provides an analysis of the notion of generalized predicativity, in particular in an attempt of making more precise the metaphor of inductively defined collections as collections *built up from below* and hence predicatively admissible. I explain the main differences between the constructive notion of generalized predicativity and classical predicativity and what is the relationship intercurring between general inductive types in type theory and Martin-Löf's meaning explanations, that can be regarded as an informal semantics for type theory. Indeed, it is claimed sometimes that meaning explanations justify the predicative status of general inductive constructions as those found in Intuitionistic Type Theory. In doing so, I am also led to consider the link with cognate topics, such as the status of Bar Induction and the meaning of Π^1_1-statements and the view of inductive definitions in Brouwerian intuitionism. Finally, I briefly review some conjectures about the limits of generalized predicative systems. I think that the main interest of these final sections lies in the fact that it represents an attempt to work out some insights that are often only hinted at in the literature. However, I do not claim that they provide a complete philosophical analysis of the notion of generalized predicativity. Rather, these considerations should be further refined (and, possibly, extended) in order to get a more comprehensive view of the topic.

Chapter 1

Martin-Löf Type Theories

Intuitionistic Type Theory or *Martin-Löf Type Theory* is a full-scale foundational system for constructive mathematics and, at the same time, an abstract framework for functional programming languages. The early version of Intuitionistic Type Theory was introduced by Per Martin-Löf in the seminal papers [97, 102]. I decided to make use of the plural 'Martin-Löf Type Theories', since in the literature one can find several variants of the theory. All these variants share some distinctive features, but in many cases they present subtle conceptual differences as well as different expressive strength and different meta-theoretical properties. In the following, I will survey the main alternative presentations, but I will adopt the intensional and predicative version of the theory, as presented in [97], as frame of reference. I will refer to it as **MLTT**. The choice is motivated also by the recent advances in *Homotopy Type Theory* [155], that has deeply influenced the current research in type theory and has increased the attention towards the subject. In fact, as we will see in Chapter 2, these developments are grounded on the intensional version of the theory.

1.1 Judgements and Types

In type theory, every object comes equipped with its *type*. In particular, this means that we will never consider an object a without specifying the type to which it belongs. The informal expression 'a is an object of type A' is formally expressed as '$a : A$'. The relation expressed by '$a : A$' must not be confused with the set-theoretical membership relation usually denoted

by '$a \in A$'. In a system of set-theory as ZFC (but also in its constructive counterparts), asserting $a \in A$ amounts to connect two objects which are both *sets*. The universe of mathematical objects is made of sets and the membership relation can be applied to any two of them. The language of set theory is essentially built on the primitive symbol $\in$ and some formalization of first-order logic. In contrast to this, the universe of mathematical objects in **MLTT** is typed and the specification of the logical system and the definition of type constructors are done simultaneously. Indeed, the system is grounded on the *propositions-as-types* principle and hence propositions and types are *identified*. It follows that when we are defining the rules for a given type A, then they are equally valid for the corresponding proposition and vice-versa. Then, the notation '$a : A$' can be equivalently taken as meaning that a is a proof of the proposition A. Moreover, the whole expression stands for a *judgement*, that is something different from a proposition[1]. To illustrate this distinction, let us consider the formation rule for the conjunction connective that in this context is denoted by $\times$:

$$\frac{\Gamma \vdash A \; type \qquad \Gamma \vdash B \; type}{\Gamma \vdash A \times B \; type} \times\text{-F}$$

We can see that the connective applies to propositions/types, but the premises as well as the conclusion of the rule are judgements. In general, logical constants can be applied to propositions, but not to judgements. Therefore, in a formal system like **MLTT** a derivation is a tree whose root is the judgement representing the conclusion which is derived from other judgements by means of the logical rules.
Further, since propositions are identified with types, proof-objects are explicitly recorded in the syntax: e.g. the introduction rule for $\times$ is given as follows

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash b : B}{\Gamma \vdash \langle a, b \rangle : A \times B} \times\text{-I}$$

In this sense, the propositions-as-types principle provides a rigorous formalization of the intuitionistic explanation of logical constants known as BHK.

[1]The fundamental philosophical distinction between judgements and propositions is discussed at length in various places by Martin-Löf, but the main reference is [100].

One of the main feature of **MLTT** is the presence of *dependent types*. A type $A(x)$ where $x : B$ is called dependent since it depends or varies on terms of another type, that may be dependent as well. Dependent types allow to extend the correspondence between types and propositions to the case of first-order quantifiers $\forall, \exists$ and to the equality relation. Set-theoretically, one can think of a dependent type as an indexed family of sets.
The basic forms of judgement of **MLTT** are the following:

$\mathcal{J}_1$) $\Gamma \vdash A \; type$ [A is a well-formed type in context Γ]

$\mathcal{J}_2$) $\Gamma \vdash A \equiv B \; type$ [A and B are *judgementally* equal types in context Γ]

$\mathcal{J}_3$) $\Gamma \vdash a : A$ [a is a term of type A in context Γ]

$\mathcal{J}_4$) $\Gamma \vdash a \equiv b : A$ [a and b are jdm. equal terms of type A in context Γ]

Here Γ stands for an arbitrary *context*. A context is a list of typed variables $x_1 : A_1, x_2 : A_2(x_1), \ldots, x_n : A_n(x_1, \ldots, x_{n-1})$ that may occur free in the judgement. Note that each declaration in the list may depend on the previous ones. A judgement is said to be *categorical* or *closed* if Γ is empty, otherwise it is said to be *hypothetical* or *open*. In some places, I will make use also of the notation A [Γ] and $a : A$ [Γ] for $\mathcal{J}_1$ and $\mathcal{J}_3$.
The notion of judgemental equality is a primitive notion of equality between formal expressions and is a judgement. We will see that there is another notion of equality in **MLTT** which behaves as a type/proposition and the relationship between these two notions is analyzed in Section 1.3.
We may also assume two auxiliary form of judgements Γ `ctxt` meaning that Γ is a well-formed context and $\Gamma \equiv \Delta$ `ctxt` meaning that Γ and Δ are judgementally equal contexts. However, these forms can be reduced to the others.
In the following we are going to specify a set of rules that represent the *structural core* of **MLTT**. It consists of the rules for context formation, variable assumption, judgemental equality, conversion, weakening and substitution.

- Context formation:

$$\vdash \cdot \; \mathsf{ctxt} \; \text{(Empty-C)} \qquad \frac{\Gamma \vdash A \; type}{\vdash \Gamma, x : A \; \mathsf{ctxt}} \; \text{(Ext-C)}$$

- Variable Assumption:

$$\frac{\vdash \Gamma, x : A, \Delta \ \mathsf{ctxt}}{\Gamma, x : A, \Delta \vdash x : A} \text{ (Var-Ass)}$$

- Judgemental Equality:

$$\frac{\Gamma \vdash A \ type}{\Gamma \vdash A \equiv A \ type} \text{ (Ty-Eq-refl)} \qquad \frac{\Gamma \vdash A \equiv B \ type}{\Gamma \vdash B \equiv A \ type} \text{ (Ty-Eq-sym)}$$

$$\frac{\Gamma \vdash A \equiv B \ type \qquad \Gamma \vdash B \equiv C \ type}{\Gamma \vdash A \equiv C \ type} \text{ (Ty-Eq-trans)}$$

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash a \equiv a : A} \text{ (Tm-Eq-refl)} \qquad \frac{\Gamma \vdash a \equiv b : A}{\Gamma \vdash b \equiv a : A} \text{ (Tm-Eq-sym)}$$

$$\frac{\Gamma \vdash a \equiv b : A \qquad \Gamma \vdash b \equiv c : A}{\Gamma \vdash a \equiv c : A} \text{ (Tm-Eq-trans)}$$

- Conversion:

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash A \equiv B \ type}{\Gamma \vdash a : B} \text{ (Conv)}$$

$$\frac{\Gamma \vdash a \equiv b : A \qquad \Gamma \vdash A \equiv B \ type}{\Gamma \vdash a \equiv b : B} \text{ (Conv-Eq)}$$

- Weakening:

$$\frac{\Gamma \vdash A \ type \qquad \Gamma, \Delta \ \mathsf{ctxt}}{\Gamma, \Delta \vdash A \ type} \text{ (W-Ty)}$$

$$\frac{\Gamma \vdash A \equiv B \ type \qquad \Gamma, \Delta \ \mathsf{ctxt}}{\Gamma, \Delta \vdash A \equiv B \ type} \text{ (W-Ty-Eq)}$$

$$\frac{\Gamma \vdash a : A \qquad \Gamma, \Delta \ \mathsf{ctxt}}{\Gamma, \Delta \vdash a : A} \text{ (W-Tm)}$$

$$\frac{\Gamma \vdash a \equiv b : A \qquad \Gamma, \Delta \ \mathsf{ctxt}}{\Gamma, \Delta \vdash a \equiv b : A}\ \text{(W-Tm-Eq)}$$

- Substitution: let us assume for simplicity that $\mathcal{J}$ ranges over the basic forms of judgement. Then we have

$$\frac{\Gamma, x : A, \Delta \vdash \mathcal{J} \qquad \Gamma \vdash a : A}{\Gamma, \Delta[a/x] \vdash \mathcal{J}[a/x]}\ \text{Subst}$$

  that for each value of $\mathcal{J}$ gives the corresponding substitution rule. Further, it is assumed that the usual conventions about renaming are defined in order to prevent the capture of free variables.

The calculus can be formulated in a way such that the weakening and substitution rules are *admissible*.

1.2 Type Constructors

The system **MLTT** is formulated in a natural-deduction style. The rules for types and terms constructors are given according to a standard pattern consisting of *formation* rules, *introduction* rules, *elimination* rules and *computation* rules. In the following, we present the rules for the most common constructors.

The rules for the ground types $\mathbf{0}$ and $\mathbf{1}$ are the following:

$$\frac{\Gamma\ \mathtt{ctxt}}{\Gamma \vdash \mathbf{0}\ \mathit{type}}\ \mathbf{0}\text{-F} \qquad \frac{\Gamma, x : \mathbf{0} \vdash C(x)\ \mathit{type} \qquad \Gamma \vdash a : \mathbf{0}}{\Gamma \vdash \mathsf{ind}_{\mathbf{0}}(a) : C(a)}\ \mathbf{0}\text{-E}$$

$$\frac{\Gamma\ \mathtt{ctxt}}{\Gamma \vdash \mathbf{1}\ \mathit{type}}\ \mathbf{1}\text{-F} \qquad \frac{\Gamma\ \mathtt{ctxt}}{\Gamma \vdash \star : \mathbf{1}}\ \mathbf{1}\text{-I}$$

$$\frac{\Gamma, x : \mathbf{1} \vdash C(x)\ \mathit{type} \qquad \Gamma \vdash c : C(\star) \qquad \Gamma \vdash a : \mathbf{1}}{\Gamma \vdash \mathsf{ind}_{\mathbf{1}}(c, a) : C(a)}\ \mathbf{1}\text{-E}$$

$$\frac{\Gamma, x : \mathbf{1} \vdash C(x)\ \mathit{type} \qquad \Gamma \vdash c : C(\star)}{\Gamma \vdash \mathsf{ind}_{\mathbf{1}}(c, \star) \equiv c : C(\star)}\ \mathbf{1}\text{-C}$$

Remark 1.2.1. As you can see, there is no introduction rule for the empty type **0** (and, accordingly, no computation rule). This is in conformity with the intuitionistic explanation of the meaning of the falsum $\bot$, that asserts that there is no canonical proof of $\bot$. Informally, if one 'forgets ' the proof-terms, then one can immediately recover the usual elimination rule for $\bot$ (*ex falso*). Analogously, **1** corresponds to the true constant $\top$. The same method of forgetting or deleting the proof-terms can be applied also to derive the rules for the other logical constants.

Another ground type is the type of Booleans Bool, whose rules are:

$$\frac{\Gamma \ \mathtt{ctxt}}{\Gamma \vdash \mathsf{Bool}\ \mathit{type}}\ \mathsf{Bool\text{-}F$$

$$\frac{\Gamma \ \mathtt{ctxt}}{\Gamma \vdash \mathsf{tt} : \mathsf{Bool}}\ \mathsf{Bool\text{-}I_1} \qquad \frac{\Gamma \ \mathtt{ctxt}}{\Gamma \vdash \mathsf{ff} : \mathsf{Bool}}\ \mathsf{Bool\text{-}I_2}$$

$$\frac{\begin{array}{c}\Gamma, x : \mathsf{Bool} \vdash C(x)\ \mathit{type} \\ \Gamma \vdash c_0 : C(\mathsf{tt}) \\ \Gamma \vdash c_1 : C(\mathsf{ff}) \\ \Gamma \vdash a : \mathsf{Bool}\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathsf{Bool}}(a, c_0, c_1) : C(a)}\ \mathsf{Bool\text{-}E}$$

$$\frac{\begin{array}{c}\Gamma, x : \mathsf{Bool} \vdash C(x)\ \mathit{type} \\ \Gamma \vdash c_0 : C(\mathsf{tt}) \\ \Gamma \vdash c_1 : C(\mathsf{ff})\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathsf{Bool}}(\mathsf{tt}, c_0, c_1) \equiv c_0 : C(\mathsf{tt})}\ \mathsf{Bool\text{-}C_1}$$

$$\frac{\begin{array}{c}\Gamma, x : \mathsf{Bool} \vdash C(x)\ \mathit{type} \\ \Gamma \vdash c_0 : C(\mathsf{tt}) \\ \Gamma \vdash c_1 : C(\mathsf{ff})\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathsf{Bool}}(\mathsf{ff}, c_0, c_1) \equiv c_1 : C(\mathsf{ff})}\ \mathsf{Bool\text{-}C_2}$$

Remark 1.2.2. The elimination rule for Bool gives exactly the $\mathsf{if}\ldots\mathsf{then}\ldots\mathsf{else}$ operator, which is familiar from programming.

Remark 1.2.3. All the examples of type constructors given so far are just particular instances of *finite* types. Indeed, we could formulate general rules for the these as well and show that the rules for $\mathbf{0}, \mathbf{1}$ and Bool can be easily derived from the general ones.

The most paradigmatic example of *inductive* type is the ground type of natural numbers $\mathbb{N}$, that is also the simplest example of infinite type. The rules for $\mathbb{N}$ are the following:

$$\frac{\Gamma\ \mathtt{ctxt}}{\Gamma \vdash \mathbb{N}}\ \mathbb{N}\text{-F} \qquad \frac{\Gamma\ \mathtt{ctxt}}{\Gamma \vdash 0 : \mathbb{N}}\ \mathbb{N}\text{-I}_1 \qquad \frac{\Gamma \vdash a : \mathbb{N}}{\Gamma \vdash \mathsf{succ}(a) : \mathbb{N}}\ \mathbb{N}\text{-I}_2$$

$$\frac{\begin{array}{c}\Gamma, x : \mathbb{N} \vdash C(x)\ \mathit{type} \\ \Gamma \vdash c : C(0) \\ \Gamma, y : \mathbb{N}, z : C(y) \vdash d(y,z) : C(\mathsf{succ}(y)) \\ \Gamma \vdash a : \mathbb{N}\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathbb{N}}(a, c, y.z.d(y,z)) : C(a)}\ \mathbb{N}\text{-E}$$

$$\frac{\begin{array}{c}\Gamma, x : \mathbb{N} \vdash C(x)\ \mathit{type} \\ \Gamma \vdash c : C(0) \\ \Gamma, y : \mathbb{N}, z : C(y) \vdash d(y,z) : C(\mathsf{succ}(y))\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathbb{N}}(0, c, y.z.d(y,z)) \equiv c : C(0)}\ \mathbb{N}\text{-C}_1$$

$$\frac{\begin{array}{c}\Gamma, x : \mathbb{N} \vdash C(x)\ \mathit{type} \\ \Gamma \vdash c : C(0) \\ \Gamma, y : \mathbb{N}, z : C(y) \vdash d(y,z) : C(\mathsf{succ}(y)) \\ \Gamma \vdash a : \mathbb{N}\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathbb{N}}(\mathsf{succ}(a), c, y.z.d(y,z)) \equiv d(a, \mathsf{ind}_{\mathbb{N}}(a, c, y.z.d(y,z)) : C(\mathsf{succ}(a))}\ \mathbb{N}\text{-C}_2$$

Notation. The notation $y.z.d(y,z)$ represents the fact that the variables y and z are bound in d.

Remark 1.2.4. The two introduction rules for $\mathbb{N}$ say that canonical elements of type $\mathbb{N}$ are obtained from 0 by applying the successor operation. If we perform the operation of deleting the proof-terms in the case of the elimination rule for $\mathbb{N}$, then we recover the familiar induction principle for natural numbers: we can infer that $C(a)$ holds from the assumptions that C holds for 0 and that, for any natural number x, if C holds for x, then C holds for the successor of x.

Remark 1.2.5. The elimination rule for $\mathbb{N}$ gives at the same time also a principle of definition by recursion. In other words, we can make use of the eliminator $\mathsf{ind}_{\mathbb{N}}$ to define new functions over $\mathbb{N}$. A standard example is the definition of $n+m$ $[n, m : \mathbb{N}]$. We can put $a+b \ :\equiv \ \mathsf{add}(a,b) \ :\equiv \ \mathsf{ind}_{\mathbb{N}}(b, a, x.y.\mathsf{succ}(y))$. Then

$$a + 0 \equiv \mathsf{add}(a, 0) \equiv a$$

and

$$a + \mathsf{succ}(m) \equiv \mathsf{add}(a, \mathsf{succ}(m)) \equiv \mathsf{succ}(\mathsf{add}(a, m))$$

So far we have considered just simple types, i.e types that are not dependent. In the following, *dependent product* Π and *dependent sum* Σ are introduced. Let us start from Π:

$$\frac{\Gamma \vdash A \ type \qquad \Gamma, x : A \vdash B(x) \ type}{\Gamma \vdash \Pi_{(x:A)} B(x) \ type} \ \Pi\text{-F}$$

$$\frac{\Gamma, x : A \vdash b(x) : B(x)}{\Gamma \vdash \lambda(x : A).b(x) : \Pi_{(x:A)} B(x)} \ \Pi\text{-I}$$

$$\frac{\Gamma \vdash f : \Pi_{(x:A)} B(x) \qquad \Gamma \vdash a : A}{\Gamma \vdash \mathsf{App}(f, a) : B(a)} \ \Pi\text{-E}$$

$$\frac{\Gamma, x : A \vdash b(x) : B(x) \qquad \Gamma \vdash a : A}{\Gamma \vdash \mathsf{App}(\lambda(x : A).b(x), a) \equiv b(a) : B(a)} \ \Pi\text{-C}$$

Notation. The notation $\mathsf{App}(f, a)$ denotes the *application* of f to a. An alternative notation, which I will employ later, is simply $f(a)$. Also I will write $\lambda x.b(x)$ instead of the full form $\lambda(x : A).b(x)$.

Remark 1.2.6. The introduction rule for Π states that a canonical element of this type has the form of a λ-abstraction. The term $b(x)$ $[x : A]$ can be regarded as a function that, for each $a : A$, yields a term $b(a) : B(a)$. Then $\lambda(x : A).b(x)$ is a 'name ' for this function ([99], p. 14). It follows, in particular, that in type theory there is a primitive notion of function. Moreover, talking of λ-abstraction and application immediately calls to mind the same operations from λ-calculus. In analogy with λ-calculus terminology, the rule Π-C can be viewed as a generalization of the β-rule to the dependent case. Finally, we can introduce two other rules familiar from λ-calculus: the ξ-rule and the η-rule. The first states that the canonical term constructor of Π respects judgemental equality[2] and is the following:

$$\frac{\Gamma, x : A \vdash b(x) \equiv b'(x) : B(x)}{\Gamma \vdash \lambda(x : A).b(x) \equiv \lambda(x : A).b'(x) : \Pi_{(x:A)}B(x)}\ \Pi\text{-}\xi$$

The second is a kind of *uniqueness* principle asserting that any dependent function is a λ-abstraction and is formulated as follows:

$$\frac{\Gamma \vdash f : \Pi_{(x:A)}B(x)}{\Gamma \vdash f \equiv \lambda(x : A).b(x) : \Pi_{(x:A)}B(x)}\ \Pi\text{-}\eta$$

Remark 1.2.7. $\Pi_{(x:A)}B(x)$ can be viewed as the type of dependent functions from A to B. It follows that we cannot inductively define all the elements of the types having this form (e.g., think of $\Pi_{(x:\mathbb{N})}\mathbb{N}$) and this explains the form of the elimination rule. In fact, borrowing some terminology from the literature on linear logic, we can see that Π is an example of *negative* type. A negative type is defined by its eliminators. An element of a negative type is characterized by its behaviour whenever the corresponding eliminator constant is applied to it. In the case of Π, this means that a dependent function is

[2]It is possible to assume that all the constructors of the system preserve this notion.

primarily defined by its behaviour on an arbitrary input (application). On the other hand, a *positive* type is completely defined by its constructors and hence it is inductively defined. The elimination rule then is an induction principle, following the pattern we have seen at work in the previous type constructors.
Quite interestingly, it is possible to formulate the rules for the dependent product in a positive way. The formation and the introduction rules are left unchanged, while the elimination rule and the computation rule are formulated as follows

$$\frac{\begin{array}{c}\Gamma, z : \Pi_{(x:A)} B(x) \vdash C(z)\ \mathit{type} \\ \Gamma \vdash f : \Pi_{(x:A)} B(x) \\ \Gamma, w : (x : A)B(x) \vdash d(w) : C(\lambda(x : A).w)\end{array}}{\Gamma \vdash \mathsf{funsplit}(f, d) : C(f)}\ \Pi\text{-E}_2$$

$$\frac{\begin{array}{c}\Gamma, z : \Pi_{(x:A)} B(x) \vdash C(z)\ \mathit{type} \\ \Gamma, x : A \vdash b(x) : B(x) \\ \Gamma, w : (x : A)B(x) \vdash d(w) : C(\lambda(x : A).w)\end{array}}{\Gamma \vdash \mathsf{funsplit}(\lambda(x : A).b(x), d) \equiv d(b) : C(\lambda(x : A).b(x))}\ \Pi\text{-C}_2$$

It must be stressed that these rules cannot be expressed in the usual formulation of type theory. The assumption $\Gamma, w : (x : A)B(x) \vdash d(w) : C(\lambda(x : A).w)$ amounts to a *higher-level* judgement of the form $(x : A \vdash w(x) : B(x)) \vdash d(w) : C(\lambda(x : A).w)$. These rules are usually assumed in versions of type theory based on a *logical framework* [101]. Similar rules have been investigated in a proof-theoretic context as higher-level inference rules [137]. With these rules at hand, the elimination rule for the dependent product fits the same pattern of the other elimination rules.
The relationship between the positive and the negative version of the Π-type has been thoroughly investigated in [54].

Remark 1.2.8. According to BHK, the meaning of the universal quantifier is explained as follows: a proof of $\forall x A(x)$ is a *method* that, for an arbitrary individual c, yields a proof of $A(c)$. Here method is just another name for function. Then, a proof of a universally quantified statement is a function. It is clear that there is a strict link with the rules for Π. In fact, the dependent

product corresponds to the universal quantifier. The usual rules for $\forall$ can be recovered by applying the usual strategy of deleting the proof-terms.

Remark 1.2.9. Think of $\Pi_{(x:\mathbb{N})}\mathbb{N}$: this is the type of functions from $\mathbb{N}$ to $\mathbb{N}$. Since $\mathbb{N}$ is a simple type, we can write it in the usual notation $\mathbb{N} \to \mathbb{N}$. In general, it may be the case that B does not depend on A. Then we put $A \to B \ :\equiv \ \Pi_{(x:A)}B$.
It is immediate to see that in this case the rules are exactly those for the arrow type in the simply typed λ-calculus. The correspondence with logic is also well-known: the type $A \to B$ corresponds to the implication connective.

The rules for Σ are the following:

$$\frac{\Gamma \vdash A \ type \qquad \Gamma, x : A \vdash B(x) \ type}{\Gamma \vdash \Sigma_{(x:A)}B(x) \ type}\ \Sigma\text{-F}$$

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash b : B(a)}{\Gamma \vdash \langle a, b\rangle : \Sigma_{(x:A)}B(x)}\ \Sigma\text{-I}$$

$$\frac{\begin{array}{c}\Gamma, z : \Sigma_{(x:A)}B(x) \vdash C(z) \ type \\ \Gamma, x : A, y : B(x) \vdash d(x, y) : C(\langle x, y\rangle) \\ \Gamma \vdash c : \Sigma_{(x:A)}B(x)\end{array}}{\Gamma \vdash \mathsf{ind}_\Sigma(c, x.y.d(x, y)) : C(c)}\ \Sigma\text{-E}$$

$$\frac{\begin{array}{c}\Gamma, z : \Sigma_{(x:A)}B(x) \vdash C(z) \ type \\ \Gamma, x : A, y : B(x) \vdash d(x, y) : C(\langle x, y\rangle) \\ \Gamma \vdash a : A \\ \Gamma \vdash b : B(a)\end{array}}{\Gamma \vdash \mathsf{ind}_\Sigma(\langle a, b\rangle, x.y.d(x, y)) \equiv d(a, b) : C(\langle a, b\rangle)}\ \Sigma\text{-C}$$

Remark 1.2.10. The canonical term constructor of a Σ-type is a *pairing* operator $\langle\cdot,\cdot\rangle$. The first element of the pair is a term belonging to A and the second, under the logical reading, is a proof-object witnessing that B holds for the first. The relation with the BHK explanation of the existential

quantifier is evident, since according to that a proof of an existentially quantified statement is a pair given by an individual a and a proof of $B(a)$. However, the relationship with the intuitionistic existential quantifier is not as straightforward as in the other cases examined so far. Actually, the Σ-elimination rule is *stronger* than the usual $\exists$-elimination. This additional strength is made explicit in the definition of *projections* by means of the Σ-elimination rule. The derived rules for projections can be taken as alternative elimination rules for Σ:

$$\frac{\Gamma \vdash c : \Sigma_{(x:A)} B(x)}{\Gamma \vdash \mathsf{fst}(c) : A}\ \Sigma\text{-}\mathsf{fst} \qquad \frac{\Gamma \vdash c : \Sigma_{(x:A)} B(x)}{\Gamma \vdash \mathsf{snd}(c) : B(\mathsf{fst}(c))}\ \Sigma\text{-}\mathsf{snd}$$

with the obvious computation rules $\mathsf{fst}(\langle a, b\rangle) \equiv a$ and $\mathsf{snd}(\langle a, b\rangle) \equiv b$.
The relation between the constructive notion of existence and its type-theoretical formalization is quite subtle: later we will see two other ways of representing the existential quantifier in a type-theoretic context, that are weaker than the identification of $\exists$ with Σ.
Finally, it should be noted that if it is the case that B does not depend on A, then we get the product type $A \times B \ :\equiv\ \Sigma_{(x:A)} B$, which corresponds to the conjunction connective in logic.

Remark 1.2.11. The strength of the rules for Σ has among its consequences the derivability of the type-theoretical *Axiom of Choice*:

$$(\ \Pi_{(x:A)} \Sigma_{(y:B(x))} R(x, y)\ ) \to (\ \Sigma_{(f:\Pi_{x:A} B(x))} \Pi_{(x:A)} R(x, f(x))\ )$$

The proof is well-known [102, 97, 99]. We start by assuming $g : \Pi_{(x:A)} \Sigma_{(y:B(x))} R(x, y)$ and $x : A$. Then by Π-E, $g(x) : \Sigma_{(y:B(x))} R(x, y)$. Then we proceed by applying the derived projection rules to this term and we get

$$\mathsf{fst}(g(x)) : B(x)$$

by Σ-fst, and

$$\mathsf{snd}(g(x)) : R(x, \mathsf{fst}(g(x)))$$

by Σ-snd. By applying first Π-I and then Π-C, we obtain

$$(\lambda x.\mathsf{fst}(g(x)))(x) \equiv \mathsf{fst}(g(x)) : B(x)$$

and by substitution

$$R(x, (\lambda x.\mathsf{fst}(g(x)))(x)) \equiv R(x, \mathsf{fst}(g(x)))$$

In this way by applying conversion, we can conclude

$$\mathsf{snd}(g(x)) : R(x, (\lambda x.\mathsf{fst}(g(x)))(x))$$

that followed by an application of Π-I yields

$$\lambda x.\mathsf{snd}(g(x)) : \Pi_{(x:A)} R(x, (\lambda x.\mathsf{fst}(g(x)))(x))$$

finally, by Σ-I, we get

$$\langle \lambda x.\mathsf{fst}(g(x)), \lambda x.\mathsf{snd}(g(x)) \rangle : \Sigma_{(f:\Pi_{x:A} B(x))} \Pi_{(x:A)} R(x, f(x))$$

and by discharging the assumption g the proof is concluded.

The *disjoint sum* or *coproduct* type $+$ is characterized by the following rules:

$$\frac{\Gamma \vdash A \; type \qquad \Gamma \vdash B \; type}{\Gamma \vdash A + B \; type} \; +\text{-F}$$

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash \mathsf{inl}(a) : A + B} \; +\text{-I}_1 \qquad \frac{\Gamma \vdash b : B}{\Gamma \vdash \mathsf{inr}(b) : A + B} \; +\text{-I}_2$$

$$\frac{\begin{array}{c}\Gamma, z : A + B \vdash C(z) \; type \\ \Gamma, x : A \vdash d_A(x) : C(\mathsf{inl}(x)) \\ \Gamma, y : B \vdash d_B(y) : C(\mathsf{inr}(y)) \\ \Gamma \vdash c : A + B\end{array}}{\Gamma \vdash \mathsf{ind}_{A+B}(c, x.d_A(x), y.d_B(y)) : C(c)} \; +\text{-E}$$

$$\frac{\begin{array}{c}\Gamma, z : A + B \vdash C(z) \; type \\ \Gamma, x : A \vdash d_A(x) : C(\mathsf{inl}(x)) \\ \Gamma, y : B \vdash d_B(y) : C(\mathsf{inr}(y)) \\ \Gamma \vdash a : A\end{array}}{\Gamma \vdash \mathsf{ind}_{A+B}(\mathsf{inl}(a), x.d_A(x), y.d_B(y)) \equiv d_A(a) : C(\mathsf{inl}(a))} \; +\text{-C}_1$$

$$\frac{\begin{array}{c}\Gamma, z : A + B \vdash C(z)\ type \\ \Gamma, x : A \vdash d_A(x) : C(\mathsf{inl}(x)) \\ \Gamma, y : B \vdash d_B(y) : C(\mathsf{inr}(y)) \\ \Gamma \vdash b : B\end{array}}{\Gamma \vdash \mathsf{ind}_{A+B}(\mathsf{inr}(b), x.d_A(x), y.d_B(y)) \equiv d_B(b) : C(\mathsf{inr}(b))}\ +\text{-C}_2$$

Remark 1.2.12. Logically, the sum type + corresponds to disjunction. However, as in the case of Σ, the elimination rule is stronger than the usual elimination rule for $\vee$, since explicit information about the witnesses is available.

1.2.1 Varieties of Inductive Definitions

What makes **MLTT** a powerful framework for the formalization of mathematics is the presence of inductively defined types as well as different generalizations thereof.
An inductive type can be regarded as the *free structure* generated by the iterated applications of a collection of constructors. These constructors are either constants or functions having the inductive type A as codomain. Then, the elimination rule is a *closure* principle asserting that any element of A is obtained by applying the constructors associated to A.
This informal description is approximately right, but some clarification is in order. Taken without any constraint, this characterization leads to paradoxical results. In order to avoid unpleasant consequences, we require that inductively defined types must satisfy *strict positivity* [28]. Roughly, this condition amounts to the fact that the inductive type being defined never occurs on the left of an arrow in the domain of its constructors. E.g. the type $(A \to A) \to A$ clearly violates this condition and in fact it can be shown to lead to paradoxes.
All the inductive types we have considered so far are strictly positive. From a categorical point of view, these types are characterized by *mapping out* universal properties. On the other hand, negative types satisfy *mapping in* universal properties, as one would expect. In particular, elimination rules express the associated universal property and computation rules the fact that the relevant equations are satisfied.
In **MLTT**, there are also constructors corresponding to *generalized* inductive definitions. The most important example is represented by $\mathcal{W}$-types. I will not consider this constructor for the moment, since it will be examined in the last

chapter. But I remark that $\mathcal{W}$-types increase in a substantial way the proof-theoretic strength of **MLTT** and, under some extensionality assumptions, they can be used to encode all the other inductively defined types. Finally, I point out that inductive types admit a well-known characterization in categorical terms as *initial algebras* for endofunctors. We recall the notion of F-algebra for an endofunctor $F : \mathcal{C} \to \mathcal{C}$:

Definition 1.2.13. An F-algebra is a pair (X, α) where X is an object in $\mathcal{C}$ and $\alpha : F(X) \to X$. A morphism of F-algebras is a morphism $f : X \to Y$ such that $f \circ \alpha = \beta \circ F(f)$.

These data form a category and then an initial F-algebra is the initial object in the category of F-algebras.

Example 1.2.14. In the case of $\mathbb{N}$, the relevant endofunctor is $F(X) :\equiv X + 1$. In the case of $+$, the endofunctor is just the constant endofunctor $F(X) :\equiv A + B$.

In general, we can introduce also *inductive families*, that are a kind of indexed inductive definition. The basic insight is that what we define inductively is not a single type, but a whole family of types simultaneously. An important constructor, which is an example of inductively defined family, is the propositional identity type, that I will present in the next section.
Among the possible generalizations of inductive definitions, we have also *inductive-inductive* definitions and *inductive-recursive* definitions. The former is a kind of definition consisting of a type A and a family B indexed by A that are both inductively defined and, moreover, they can refer to each other. The latter consists of an inductively defined type A and $B(x)[x : A]$ defined *by recursion* over A. Also, we assume that A can make use of B. The difference with inductive-inductive definitions is that in inductive-recursive types B is specified by a list of equations, when in the first case we give explicit constructors for the family as well. An important example of inductive-recursive definition is represented by *Tarski universes*.

1.2.2 Universes

It is well-known that the first version of intuitionistic type theory proposed by Martin-Löf [96] had the impredicative axiom $V : V$, where V was intended to be the type of *all* types. However, J.Y.Girard was able to show that the

system was contradictory [55]. This result is known as *Girard's Paradox* and highlights the problematic relationship between impredicativity and the propositions-as-types discipline.
After the discovery of the paradox, Martin-Löf formulated a predicative version of the theory with a type of *small* types $\mathcal{U}$, which allowed to define type-valued functions and thus makes the whole system more powerful. This type is called *universe* and has as objects all the small types introduced. Therefore, $\mathbf{0}, \mathbf{1}, \mathbb{N}$ are in $\mathcal{U}$ and, moreover, $\mathcal{U}$ is closed under the type constructors considered so far $\Pi, \Sigma, +$ as well as the others we are going to introduce later like $\mathrm{Id}, \mathcal{W}, \ldots$.
It is fundamental that $\mathcal{U}$ is a type of small types, since this allows to avoid Girard's paradox. Even if $\mathcal{U}$ itself is a type, it is not possible to conclude that $\mathcal{U} : \mathcal{U}$ holds.
The introduction of a universe is basically the result of a process of *reflection* (see [102]). At a certain moment in time, we can inspect all the rules that we have given and convince ourselves that they are correct; then we recognize that all the operations we have done with types can be done inside a universe $\mathcal{U}$. In this sense, we *reflect* at the object level the type system defined so far. Moreover, the process does not lead to postulate an axiom of the form $\mathcal{U} : \mathcal{U}$ and thus predicativity and consistency are retained. We can make a bit more precise the idea of inspecting the rules and seeing their (informal) soundness by means of the meaning explanations (see Chapter 5).
On the formal level, there are two main ways for introducing a universe. They are known in the literature as *a lá Russell* and *a lá Tarski*. In a Russell-style universe, there is no distinction between types viewed primarily as constructors of the system and types viewed as elements of the universe. This means that a term of a Russell-$\mathcal{U}$ is a type. On the other hand, if the universe is formulated in a Tarski-style, then it is a pair given by a type $\mathcal{U}$ and a decoding function T recursively defined over $\mathcal{U}$. In this case, the 'actual' types are distinguished from the elements of $\mathcal{U}$, that are taken to be *codes* or *names* for the former. Then, the decoding function sends a code to the corresponding type. This kind of universes are an example of inductive-recursive definition.

Remark 1.2.15. The name 'Russell universe' is motivated by the fact that the treatment of types is akin to that occurring in the ramified theory of types. The name 'Tarski universe', instead, is due to the analogy between the decoding function and the Tarskian truth-definition.

In the following, I introduce the rules for a Tarski-style universe $(\mathcal{U}, \mathrm{T})$:

$$\frac{\Gamma\ \mathsf{ctxt}}{\Gamma \vdash \mathcal{U}\ \mathit{type}}\,\mathcal{U}\text{-F} \qquad \frac{\Gamma \vdash a : \mathcal{U}}{\Gamma \vdash \mathrm{T}(a)\ \mathit{type}}\,\mathrm{T}\text{-F}$$

$$\frac{\Gamma\ \mathsf{ctxt}}{\Gamma \vdash \widehat{\mathbf{0}} : \mathcal{U}}\,\mathcal{U}\text{-I}_1 \qquad \frac{}{\Gamma \vdash \mathrm{T}(\widehat{\mathbf{0}}) \equiv \mathbf{0}\ \mathit{type}}\,\mathrm{T}\text{-C}_1$$

$$\frac{\Gamma\ \mathsf{ctxt}}{\Gamma \vdash \widehat{\mathbf{1}} : \mathcal{U}}\,\mathcal{U}\text{-I}_2 \qquad \frac{}{\Gamma \vdash \mathrm{T}(\widehat{\mathbf{1}}) \equiv \mathbf{1}\ \mathit{type}}\,\mathrm{T}\text{-C}_2$$

$$\frac{\Gamma\ \mathsf{ctxt}}{\Gamma \vdash \widehat{\mathbb{N}} : \mathcal{U}}\,\mathcal{U}\text{-I}_3 \qquad \frac{}{\Gamma \vdash \mathrm{T}(\widehat{\mathbb{N}}) \equiv \mathbb{N}\ \mathit{type}}\,\mathrm{T}\text{-C}_3$$

$$\frac{\Gamma \vdash a : \mathcal{U} \qquad \Gamma, x : \mathrm{T}(a) \vdash b(x) : \mathcal{U}}{\Gamma \vdash \widehat{\Pi}_{(x:a)} b(x) : \mathcal{U}}\,\mathcal{U}\text{-I}_4$$

$$\frac{\Gamma \vdash a : \mathcal{U} \qquad \Gamma, x : \mathrm{T}(a) \vdash b(x) : \mathcal{U}}{\Gamma \vdash \mathrm{T}(\widehat{\Pi}_{(x:a)} b(x)) \equiv \Pi_{(x:\mathrm{T}(a))} \mathrm{T}(b(x))\ \mathit{type}}\,\mathrm{T}\text{-C}_4$$

$$\frac{\Gamma \vdash a : \mathcal{U} \qquad \Gamma, x : \mathrm{T}(a) \vdash b(x) : \mathcal{U}}{\Gamma \vdash \widehat{\Sigma}_{(x:a)} b(x) : \mathcal{U}}\,\mathcal{U}\text{-I}_5$$

$$\frac{\Gamma \vdash a : \mathcal{U} \qquad \Gamma, x : \mathrm{T}(a) \vdash b(x) : \mathcal{U}}{\Gamma \vdash \mathrm{T}(\widehat{\Sigma}_{(x:a)} b(x)) \equiv \Sigma_{(x:\mathrm{T}(a))} \mathrm{T}(b(x))\ \mathit{type}}\,\mathrm{T}\text{-C}_5$$

$$\frac{\Gamma \vdash a : \mathcal{U} \qquad \Gamma \vdash b : \mathcal{U}}{\Gamma \vdash a \widehat{+} b : \mathcal{U}}\,\mathcal{U}\text{-I}_6$$

$$\frac{\Gamma \vdash a : \mathcal{U} \qquad \Gamma \vdash b : \mathcal{U}}{\Gamma \vdash \mathrm{T}(a \widehat{+} b) \equiv \mathrm{T}(a) + \mathrm{T}(b)\ \mathit{type}}\,\mathrm{T}\text{-C}_6$$

$$\frac{\Gamma \vdash a : \mathcal{U} \qquad \Gamma \vdash b : \mathrm{T}(a) \qquad \Gamma \vdash c : \mathrm{T}(a)}{\Gamma \vdash \widehat{\mathrm{Id}}_a(b,c) : \mathcal{U}}\,\mathcal{U}\text{-I}_7$$

$$\frac{\Gamma \vdash a : \mathcal{U} \qquad \Gamma \vdash b : \mathrm{T}(a) \qquad \Gamma \vdash c : \mathrm{T}(a)}{\Gamma \vdash \mathrm{T}(\widehat{\mathrm{Id}}_a(b,c)) \equiv \mathrm{Id}_{T(a)}(b,c)\ \mathit{type}}\,\text{T-C}_7$$

Remark 1.2.16. Observe that the introduction rules for $\mathcal{U}$ refers to the decoding function T in some places (e.g. Π, Σ). On the other hand, in the Russell-style formulation, the function T disappears and we have that if $A : \mathcal{U}$ $[\Gamma]$, then A *type* $[\Gamma]$.

Remark 1.2.17. The definition of $(\mathcal{U}, \mathrm{T})$ includes only formation and introduction rules. It is possible to add an elimination rule asserting that $\mathcal{U}$ is the least type generated by the above constructors, so as to arrive at a proper inductive definition. The elimination-rule should be of the following form:

$$\begin{gathered}
\Gamma \vdash t : \mathcal{U} \\
\Gamma, z : \mathcal{U} \vdash C(z)\ \mathit{type} \\
\Gamma \vdash t_1 : C(\widehat{\mathbf{0}}) \\
\Gamma \vdash t_2 : C(\widehat{\mathbf{1}}) \\
\Gamma \vdash t_3 : C(\widehat{\mathbb{N}}) \\
\Gamma, x : \mathcal{U}, y(z) : \mathcal{U}\ [z : \mathrm{T}(x)], u : C(x), v(z) : C(y(z)) \vdash t_4(x,y,u,v) : C(\widehat{\Pi}_{(z:x)} y(z)) \\
\Gamma, x : \mathcal{U}, y(z) : \mathcal{U}\ [z : \mathrm{T}(x)], u : C(x), v(z) : C(y(z)) \vdash t_5(x,y,u,v) : C(\widehat{\Sigma}_{(z:x)} y(z)) \\
\Gamma x : \mathcal{U}, y : \mathcal{U}, u : C(x), v : C(y) \vdash t_6(x,y,u,v) : C(x \widehat{+} y) \\
\frac{\Gamma, x : \mathcal{U}, y : \mathrm{T}(x), u : \mathrm{T}(x), v : C(x) \vdash t_7(x,y,u,v) : C(\widehat{\mathrm{Id}}_x(y,u))}{\Gamma \vdash \mathsf{ind}_{\mathcal{U}}(t, t_1, t_2, t_3, t_4, t_5, t_6, t_7) : C(t)}\,\mathcal{U}\text{-E}
\end{gathered}$$

Then there are seven computation rules, one for each constructor, that are formulated in the usual manner.
The main reason to avoid the inclusion of the elimination rule for $\mathcal{U}$ in the theory is that we want to preserve its *open-ended* nature. This means that, at some point, we may formulate valid rules for a new type-constructor

and include them in the universe. Therefore, we want to avoid any closure principle over the universe. Philosophically, this choice conforms to a constructive conception of the universe of mathematical objects as a 'growing' structure and such that the process is never completed. Another reason comes from Homotopy Type Theory: if $\mathcal{U}$ is univalent, that is, it satisfies the Univalence Axiom, then assuming an inductive elimination rule for it yields a contradiction.

The formulation of the rules for $\mathcal{U}$ gives expressive strength to the basic theory and allows to define type-valued functions. In the following, some examples are given witnessing possible applications of universes in type theory.

Example 1.2.18. The rules for $\mathcal{U}$ given above comprehend also identity types, even if they have not been introduced yet (see the next Section 1.3). However, in absence of Id-types, the equality over $\mathbb{N}$ can be defined by means of $\mathcal{U}$. We define a predicate $\mathrm{Eq}(m,n) : \mathcal{U}$ $[m,n:\mathbb{N}]$ by recursion:

$$\mathrm{Eq}(0,0) \equiv \widehat{\mathbf{1}} : \mathcal{U}$$

$$\mathrm{Eq}(\mathsf{succ}(m),0) \equiv \widehat{\mathbf{0}} : \mathcal{U}$$

$$\mathrm{Eq}(0,\mathsf{succ}(n)) \equiv \widehat{\mathbf{0}} : \mathcal{U}$$

$$\mathrm{Eq}(\mathsf{succ}(m),\mathsf{succ}(n)) \equiv \mathrm{Eq}(m,n) : \mathcal{U}$$

Moreover, it can be shown that Eq corresponds in a precise sense to $\mathrm{Id}_{\mathbb{N}}$.

Example 1.2.19. It is also useful to think of $\mathcal{U}$ as a type of truth-values. Then, a map $g : X \to \mathcal{U}$ can be regarded as an element of a (restricted) power-set operator $\mathcal{P}(X)$. The restriction amounts to the fact that $g(x)$ must be small in $\mathcal{U}$. Therefore, given $a : \mathcal{P}(X)$ and $x : X$, we can introduce the auxiliary notions

$$x \;\varepsilon\; a :\equiv \;\; \mathrm{T}(a(x))$$

representing a membership relation and

$$a \subseteq b :\equiv \;\; \Pi_{(x:X)}(x \;\varepsilon\; a \to x \;\varepsilon\; b)$$

representing inclusion.

Example 1.2.20. The combination of dependent sums with universes allows to encode many examples of mathematical structures in a very natural way. Informally, a certain structure may be given as a type equipped with a certain operation that satisfies some laws. E.g., a semigroup is given by a type endowed with an associative operation. Then we can define the type of semigroups (in $\mathcal{U}$):

$$\mathcal{SG}_{\mathcal{U}} \ :\equiv \ \Sigma_{X:\mathcal{U}}\Sigma_{*:X\times X\to X}\Pi_{x,y,z:X} x * (y * z) =_X (x * y) * z$$

where the associativity condition is written in the usual notation. Further, this type $\mathcal{SG}_{\mathcal{U}}$ is not itself a type inside $\mathcal{U}$.
This basic schema admits some variations (e.g. in **HoTT** we may require that the carrier is not an arbitrary type, but an h-set) and can be extended to many other cases.

The reflection process leading to the introduction of a universe can be iterated. Therefore, the first universe can be viewed as an element of the second and so on. Eventually we arrive at a cumulative hierarchy of universes

$$\mathcal{U}_0 : \mathcal{U}_1 : \mathcal{U}_2 \dots$$

that is externally indexed by natural numbers. The whole construction is predicatively admissible. It is also such from the point of view of ordinal proof-theory, since the theory with a hierarchy of universes and without $\mathcal{W}$-types has Γ_0 as its proof-theoretic ordinal [47].
It is possible to introduce even more complicated universe constructions, that give rise to stronger theories. Examples are the *superuniverse* first formalized by Palmgren [115] and the *Mahlo* universe introduced by Setzer [142].

1.3 Equality in Type Theory

Two different notions of equality occur in **MLTT**: *judgemental* equality and *propositional* equality. The concept of equality in type theory has received significant attention after the introduction of Homotopy Type Theory. As we will see, the homotopical interpretation stems from the rules for propositional equality in the intensional system.
Judgemental equality is primarily a syntactical notion, since it is a relation between *linguistic* expressions and not between the entities they stand for. The notions is axiomatically characterized by the rules considered in Section

1.1 and is preserved by the type-constructors rules. On the other hand, propositional equality represents equality between objects and is the type-theoretic counterpart of the notion of equality that we encounter in predicate logic. This notion is expressed in the system by the identity type Id_A associated to each type A [3], that is an example of inductively defined family and thus it satisfies a suitable universal property. It corresponds to the logical equality in the sense that it is a proposition that can be proved or disproved and also assumed as hypothesis. All these operations do not apply to judgemental equality.
It is possible to find different presentations of Id-types in the literature. The following is that originally proposed by Martin-Löf and is probably the most adopted:

$$\frac{\Gamma \vdash A\ \mathit{type} \qquad \Gamma \vdash a : A \qquad \Gamma \vdash b : A}{\Gamma \vdash \mathrm{Id}_A(a,b)}\ \text{Id-F}$$

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash \mathsf{refl}(a) : \mathrm{Id}_A(a,a)}\ \text{Id-I}$$

$$\frac{\begin{array}{c}\Gamma, x : A, y : A, q : \mathrm{Id}_A(x,y) \vdash C(x,y,q)\ \mathit{type} \\ \Gamma, z : A \vdash c(z) : C(z,z,\mathsf{refl}(z)) \\ \Gamma \vdash a : A \\ \Gamma \vdash b : A \\ \Gamma \vdash p : \mathrm{Id}_A(a,b)\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathrm{Id}}(a,b,p,z.c(z)) : C(a,b,p)}\ \text{Id-E}$$

$$\frac{\begin{array}{c}\Gamma, x : A, y : A, q : \mathrm{Id}_A(x,y) \vdash C(x,y,q)\ \mathit{type} \\ \Gamma, z : A \vdash c(z) : C(z,z,\mathsf{refl}(z)) \\ \Gamma \vdash a : A\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathrm{Id}}(a,a,\mathsf{refl}(a),z.c(z)) \equiv c(a) : C(a,a,\mathsf{refl}(a))}\ \text{Id-C}$$

Notation. I will make use also of the alternative notation $a =_A b$ for $\mathrm{Id}_A(a,b)$. When the relevant type is clear from the context, I will simply write $a = b$.

[3] Therefore, we do not have a 'global' notion of equality in type theory.

Remark 1.3.1. These rules state that Id_A is the least reflexive relation over A. The elimination rule fits the pattern of the other inductively defined types. An alternative formulation of the rules has been proposed by Paulin-Mohring [119] and appears under the name *based path induction* in [155]. The elimination rule is given relatively to a fixed term $a : A$:

$$\frac{\begin{array}{c}\Gamma, y : A, q : \mathrm{Id}_A(a, y) \vdash C(y, q)\ \mathit{type} \\ \Gamma \vdash c : C(a, \mathsf{refl}(a)) \\ \Gamma \vdash b : A \\ \Gamma \vdash p : \mathrm{Id}_A(a, b)\end{array}}{\Gamma \vdash \mathsf{ind}^2_{\mathrm{Id}}(p, c) : C(b, p)}\ \text{Id-E}_2$$

The computation rule is modified accordingly. The reader may consult [155] to see a proof that this formulation is equivalent to the previous one.

Among the consequences of Id-types, there is a derivation of the type-theoretic version of the *indiscernibility of identicals*, stating that a family of types respects propositional equality, and a proof of the fact that propositional equality is an equivalence relation.
The first property is also known as *transport* in homotopy type theory (see Lemma 2.3.1 in [155]).

Lemma 1.3.2. *Let A be a type and $x, y : A$. If $P(x)$ $[x : A]$ and $p : \mathrm{Id}_A(x, y)$, then there is map $p_* : P(x) \to P(y)$.*

Proof. Let $C(x, y, p) :\equiv P(x) \to P(y)$. Then there is a term $c :\equiv \lambda x.\mathsf{id}_{P(x)} : \Pi_{(x:A)} P(x) \to P(x)$. Then we can apply Id-E and we get

$$\mathsf{ind}_{\mathrm{Id}}(x, y, p, c) : P(x) \to P(y)$$

and we call this map p_*.

□

In particular, it follows that the following rule is admissible:

$$\frac{\begin{array}{c}\Gamma, x : A \vdash P(x)\ \mathit{type} \\ \Gamma \vdash a : A \\ \Gamma \vdash b : A \\ \Gamma \vdash p : \mathrm{Id}_A(a, b) \\ \Gamma \vdash t : P(a)\end{array}}{\Gamma \vdash p_*(t) : P(b)}$$

Notation. An alternative notation for p_* is $\mathsf{transport}^P(p, -)$, where the blank space stands for the element which is transported along p in the type family P.

The next lemma establishes that propositional equality is an equivalence relation. Reflexivity is already ensured by the introduction rule, therefore we need to prove only symmetry and transitivity.

Lemma 1.3.3. *Let A be a type and x, y, z elements of A. Then Id_A satisfies the following properties:*

1. *If $p : \mathrm{Id}_A(x, y)$, then there exists a term $p^{-1} : \mathrm{Id}_A(y, x)$.*
2. *If $p : \mathrm{Id}_A(x, y)$ and $q : \mathrm{Id}_A(y, z)$, then there exists a term $p \blacksquare q : \mathrm{Id}_A(x, z)$.*

Proof. Both items follow from applications of the Id-elimination rule. □

In the light of the remarks opening this section, a natural question would be to investigate the relationship between judgemental and propositional equality. Moreover, the way these notions are characterized determines the intensional or extensional character of the theory. Martin-Löf in [105] provided a conceptual analysis of the meaning of judgemental equality (within an intensional setting) and gave some basic results relating this notion with propositional equality. In that paper, judgemental equality is called *convertibility* and is viewed as the formal counterpart of the informal notion of *definitional equality*. According to Martin-Löf, definitional equality is exhaustively characterized by the following principles:

1. A definiens is always definitionally equal to its definiendum.
2. If we substitute two definitionally equal expressions for a variable occurring in a certain expression, then the expressions resulting from the substitution are definitionally equal.
3. Definitional equality is an equivalence relation.

The interesting consequence of this analysis is that the formalization of definitional equality does not correspond with the usual notion of convertibility as it appears in the literature on λ-calculus. In particular, some rules as the ξ-rule and the η-rule must be kept out and hence a weaker notion is obtained

[4]. Therefore, in the intensional setting, judgemental equality is an equivalence relation between expressions which is generated by computation rules and abbreviatory definitions and is closed under substitution and renaming of bound variables. We can take definitional equality and judgemental equality as being the same notion:

$$\textit{definitional} \approx \textit{judgemental}$$

The following propositions, which are taken from [105], explain the relation between these two notions in the intensional theory:

Proposition 1.3.4. *Let A be a type and a and b terms of type A. If $\Gamma \vdash a \equiv b : A$, then there exists a term p such that $\Gamma \vdash p : \mathrm{Id}_A(a, b)$. Moreover, $p \equiv \mathsf{refl}(a) \equiv \mathsf{refl}(b)$.*

Proof. $\Gamma \vdash \mathsf{refl}(a) : \mathrm{Id}_A(a, a)$ is an axiom and, since $\Gamma \vdash a \equiv b : A$ holds, by substitution we get $\Gamma \vdash \mathrm{Id}_A(a, a) \equiv \mathrm{Id}_A(a, b)$. Therefore, by conversion, $\Gamma \vdash \mathsf{refl}(a) : \mathrm{Id}_A(a, b)$. □

The converse of this proposition is not true in general. The best we can achieve is the following, that is much weaker, and depends on a 'global' feature of the system like the normalization property:

Proposition 1.3.5. *Let A be a closed type and a and b closed term in A. If $\vdash p : \mathrm{Id}_A(a, b)$, then $\vdash a \equiv b : A$.*

Proof. By normalization, the closed term p must reduce to a term of the form $\mathsf{refl}(c)$ for a certain c. Then $\mathrm{Id}_A(a, b) \equiv \mathrm{Id}_A(c, c)$ and hence, in particular, both a and b reduces to c, by confluence. Therefore, $a \equiv b : A$. □

The notions of normalization, normal form and confluence are explained in the next section.
The crucial point is that the two notions of equality cannot be conflated. On the other hand, the collapse of the distinction between judgemental equality and propositional equality is the hallmark of *extensional* type theory [98, 99]. In [99], the elimination rule for propositional equality is no more an induction principle, but is the so-called *reflection* rule:

[4]It is worth noting that this analysis no longer seems to be accepted by Martin-Löf in his later works, especially after the introduction of the logical framework [101]. For instance, here, the ξ-rule is included for functional abstraction at the LF-level. As for the status of the η-rules, a detailed analysis is contained in [75]

$$\frac{\Gamma \vdash p : \mathrm{Id}_A(a, b)}{\Gamma \vdash a \equiv b : A}$$

Then it becomes evident that judgemental and propositional equality are conflated. This change deeply affects the global properties of the system, but for the moment I just point out that, by assuming the reflection rule, many extensionality principles are derivable and the correspondence between judgemental and definitional equality is broken. In ([99], p.32), these notions are explicitly distinguished. Definitional equality is defined as "the equivalence relation generated by abbreviatory definitions, changes of bound variables and the principle of substituting equals for equals" (*ibid.*, p.16). What is missing are the computation rules as well as other principles (e.g. uniqueness) characterizing the notion of judgemental equality at the extensional level. Therefore, we have that in extensional type theory:

$$\textit{Definitional} \not\approx \textit{Judgemental}$$

1.4 Meta-theoretical Properties

In this section, I survey the main meta-theoretical properties of **MLTT**. Moreover, the discussion is meant to apply to the intensional version of the theory, i.e. without reflection rule or equivalent rules, since in the extensional case many meta-theoretical features are lost.
The computation rules associated to each type constructor can be viewed as *rewriting* or *reduction* rules, determining thus a rewriting system that enjoys good computational properties. Basically, we just give an asymmetric reading of the equations as reductions going from the left to the right. E.g. the equation $(\lambda(x : A).b(x))(a) \equiv b(a) : B(a)$ becomes

$$(\lambda(x : A).b(x))(a) \rightsquigarrow b(a) : B(a)$$

with the meaning that the term on the left *reduces* to the term on the right. The relation '$\rightsquigarrow$' is called reduction relation and is determined by the rewriting rules. In general, an expression a reduces to an expression b, if the latter is the result of iterated applications of the reductions rules to the former (or to sub-expressions of it). An expression to which a reduction rule can be applied is called *redex*. There is a subset formed by those expressions that do not contain any redex and hence cannot be further reduced: they are the expressions in *normal form*. Under the Curry-Howard correspondence, a

term in normal form corresponds to a derivation without *detours* and hence fully 'simplified'. A derivation in such form is called normal.
A basic property of the reduction relation is that it does not change the type of the expressions. Therefore, we can say that it preserves typing. This property is known as *subject reduction*

Proposition 1.4.1 (Subject Reduction)**.** *If* $\Gamma \vdash a : A$ *and* $a \rightsquigarrow b$*, then* $\Gamma \vdash b : A$*.*

An important question about a rewriting system is whether it is *confluent* or not. Confluence is also known as *Church-Rosser* property in the literature and asserts that if from a term a starts two reduction sequences, then they eventually converge.

Proposition 1.4.2 (Church-Rosser)**.** *Let* a, b, c *be terms of type* A *under a context* Γ*. If* $a \rightsquigarrow b$ *and* $a \rightsquigarrow c$*, then there exists a term* $d : A\ [\Gamma]$ *such that* $b \rightsquigarrow d$ *and* $c \rightsquigarrow d$ *in finitely many steps.*

Moreover, a corollary of this fact is that the normal form of a term is *unique*, if it exists.

Corollary 1.4.3 (Uniqueness of Normal Form)**.** *If* $a \rightsquigarrow b$ *and* $a \rightsquigarrow c$ *and both* b *and* c *are in normal form, then* $b \equiv c$*.*

In many cases, confluence can be derived as corollary of the *normalization* theorem. First, we must distinguish between two forms of normalization, *weak* and *strong*. A weak normalization theorem guarantees that *there is a terminating* reduction sequence starting from a given term a. On the other hand, a strong normalization theorem asserts that *every* reduction sequence starting from a given term a terminates. The strong form of the theorem is the following:

Theorem 1.4.4 (Strong Normalization)**.** *Let* $\Gamma \vdash a : A$ *be derivable. Then there exists a term* $b : A\ [\Gamma]$ *in normal form such that* $a \rightsquigarrow b$ *and the judgement* $\Gamma \vdash a \equiv b : A$ *is derivable. Furthermore, every reduction sequence starting from* a *terminates in finitely many steps.*

The significance of this theorem can hardly be underestimated, as it is evident also from its corollaries. Usually, *decidability* of judgemental equality and thus decidability of *type-checking* can be derived from normalization. Type-checking is the problem of whether a judgement $\Gamma \vdash a : A$ is derivable or not.

Proposition 1.4.5 (Decidability of judgemental equality)**.** *Let* $\Gamma \vdash a, b : A$. *Then it is decidable whether* $\Gamma \vdash a \equiv b : A$ *or not.*
Let $\Gamma \vdash A, B$ *type. Then it is decidable whether* $\Gamma \vdash A \equiv B$ *or not.*

The following proposition is usually grounded on the previous one:

Proposition 1.4.6 (Decidability of type-checking)**.** *It is decidable whether* $\Gamma \vdash a : A$ *or not.*

Remark 1.4.7. A type-checker is a program verifying that a given term is an element of a certain type. Therefore, it checks whether a proof is correct. Under this perspective, decidability of type-checking is a crucial ingredient for implementing the system into a proof-assistant and for formally verifying the correctness of a proof in type-theory. Furthermore, it is interesting to observe that this property was already recognized as fundamental in discussions about the foundations of constructive mathematics and, in particular, about the notion of intuitionistic proof. This fact is highlighted in [97], where the decidability of type-checking is presented as the proof of the fact that intuitionistic type theory satisfies the *adequacy condition* for a theory of constructions proposed by Kreisel in [80]. The condition amounts to requiring that it should be decidable whether a given expression is a proof of a given formula or not. It was intended as a formalization of the slogan 'we recognize a proof when we see one'. Moreover, the adequacy condition is taken by Kreisel as a criterion to distinguish between the intuitionistic explanation of logical constants and other explanations that can still be counted as constructive (e.g. Kleene's realizability).

Among the corollaries of normalization there is also *canonicity*. A term $a : A$ is canonical if it is of one of the forms determined by an introduction-rule. In other words, the outer leftmost constructor is given by an introduction rule. Canonical terms must not be confused with normal terms.

Example 1.4.8. A variable $x : A$ is a term in normal form, but it is not canonical. On the other hand, $\mathsf{succ}(\mathsf{add}(0, n))$ is canonical, but not in normal form. Finally, a term like $\mathsf{succ}(\mathsf{succ}(0))$ is both canonical and normal.

Therefore, the following facts hold:

Proposition 1.4.9 (Canonicity)**.** *Let* a *be a closed term of type* A. *Then there exists a canonical term* b *such that* $\vdash b : A$ *and* $\vdash a \equiv b : A$.

In particular, this means that in the case of $\mathbb{N}$, if $\vdash n : \mathbb{N}$, then $n \equiv \mathsf{succ}^k(0)$ for some external natural number k.

Proposition 1.4.10 (Consistency)**.** *There is no term inhabiting the type* $\mathbf{0}$ *in the empty context.*

Then by Gödel's theorem, the normalization argument cannot be entirely formalized and established in the system itself.

Remark 1.4.11. The strategy of evaluating a term to the canonical form determines a different reduction relation. In particular, this relation applies only to *closed* expressions. As we will see later, a canonical form theorem is still valid in extensional type theory, where, instead, the normalization property breaks up, and is sufficient to ensure the logical consistency of the calculus. Also the whole method of *meaning explanations* [98, 99] is based on this notion of reduction. The method must be compared with the *proof-theoretic semantics* program as envisioned by Dummett and Prawitz [40, 124, 125]: the *fundamental assumption*, namely that every closed derivation can be reduced to one whose last rule is an introduction rule, is there individuated as a genuine semantic property, contrary to normalization, and the definition of valid argument is based on canonical derivations.

1.5 Variants of MLTT

The aim of the present section is to present some main variants of the basic intensional system. Some of these variants shall be discussed in detail later, therefore what follows is meant to give just a general overview of the various systems that arise as modifications of the core system exposed in the previous sections.

Martin-Löf introduced an *extensional* version of type theory in [98]. The important innovation consisted in the fact that this system was endowed with an intended, informal semantics, whose aim was to provide a direct recognition of the validity of the rules without resorting to meta-theoretical constructions.

There are some equivalent ways for turning the intensional system into the extensional one. Martin-Löf [98, 99] simply modified the rules for propositional equality as follows:

$$\frac{\Gamma \vdash p : \mathrm{Id}_A(a,b)}{\Gamma \vdash a \equiv b : A}\ \text{Id-E} \qquad \frac{\Gamma \vdash p : \mathrm{Id}_A(a,b)}{\Gamma \vdash p \equiv \mathsf{refl}(a) : \mathrm{Id}_A(a,b)}\ \text{Id-C}$$

The rule Id-E is called *reflection rule*. It is immediately seen that such rule does not give an induction principle. The rule Id-C, instead, asserts that every proof of a propositional equality is judgementally equal to the canonical proof-term, namely reflexivity. These rules together with Proposition 1.3.4 entail the collapse of the distinction between judgemental and propositional equality.

Remark 1.5.1. Another way of making the theory extensional is to postulate the validity of some equations involving definitional equality. For example, Streicher [147] assumes an η-like principle for propositional equality that is equivalent to the assumption of the reflection rule.

The reflection rule entails many important consequences and moreover causes the failure of fundamental computational properties of the system. In extensional type theory, the following principle of *function extensionality* is derivable:

Proposition 1.5.2. *Let $f, g : \Pi_{(x:A)} B(x)$, then the following type is inhabited*

$$\mathsf{FunExt}(f,g) \ :\equiv\ \Pi_{(x:A)}(f(x) =_{B(x)} g(x)) \to f =_{\Pi_{(x:A)}B(x)} g$$

Proof. Let t be a term inhabiting $\Pi_{(x:A)} f(x) =_{B(x)} g(x)$. Then given $x : A$, $t(x) : f(x) =_{B(x)} g(x)$ and by equality reflection $f(x) \equiv g(x) : B(x)$. Then $\lambda x.f(x) \equiv \lambda x.g(x) : \Pi_{(x:A)} B(x)$ and by Π-η, $f \equiv g$. Therefore, $\mathsf{refl}(f) : f =_{\Pi_{(x:A)}B(x)} g$. □

Remark 1.5.3. It should be noted that rules like Π-η and the ξ-rule, which sometimes are not assumed in the intensional theory (e.g. η-rules are presented as *optional* principles in [155]), are always assumed in the extensional system.

Another relevant consequence of the extensional rules for propositional equality is that any two terms inhabiting an identity type are equal. This is known as *uniqueness of identity proofs* principle:

Proposition 1.5.4. *Let A be a type and $x, y : A$. If $p, q : \mathrm{Id}_A(x,y)$, then the following type is inhabited*

$$\mathrm{Id}_{\mathrm{Id}_A(x,y)}(p,q)$$

Proof. By Id-C, we get $p \equiv \mathsf{refl}(x) : \mathrm{Id}_A(x, y)$ and $q \equiv \mathsf{refl}(y) : \mathrm{Id}_A(x, y)$. Further, since by reflection $x \equiv y$, then $\mathsf{refl}(x) \equiv \mathsf{refl}(y)$ and hence $p \equiv \mathsf{refl}(x) \equiv \mathsf{refl}(y) \equiv q$. Therefore, $\mathsf{refl}(p) : \mathrm{Id}_{\mathrm{Id}_A(x,y)}(p, q)$. □

Remark 1.5.5. The uniqueness of identity proofs principle is logically equivalent to the Axiom K formulated by Streicher [147]. Even if they can be understood as extensional principles, however adding one of them to intensional type theory is not equivalent to assuming the reflection rule, that is stronger.

As anticipated, some fundamental properties are broken by moving to the extensional theory. In particular, normalization and decidability of type-checking do not hold anymore, making the implementation of extensional type theory more difficult.
The failure of normalization is shown by exhibiting a typable term that is not strongly normalizing. The standard argument requires a universe (for simplicity, I assume a Russell-style universe) and extensional propositional equality. Let us assume $A : \mathcal{U}$. Then if $z : \mathbf{0}$, by $\mathbf{0}$-E, we obtain $\mathsf{ind}_{\mathbf{0}}(z) : \mathrm{Id}_{\mathcal{U}}(A, A \to A)$. Then by reflection $A \equiv A \to A : \mathcal{U}$. If $x : A$, then by conversion $x : A \to A$ and by Π-E, $x(x) : A$. So by Π-I, $\lambda x.x(x) : A \to A$ and thus $\lambda x.x(x) : A$. Let $\omega :\equiv \lambda x.x(x)$, then $\omega(\omega) : A$, that is not normalizing w.r.t. the β-reduction relation.
The reflection rule provokes a loss of information. Informally, we may say that it forgets a proof-term inhabiting a propositional equality. Decidability of type-checking is entangled with decidability of judgemental equality, since we have the conversion rule in type theory. Then, since the reflection rule abolishes the distinction between judgemental and propositional equality, an algorithm deciding judgemental equality should decide also the inhabitation of identity types of arbitrary complexity. Therefore, we can encode mathematical statements that are known to be undecidable and thus derive the failure of judgemental equality decidability, that in turns implies that there is no terminating type-checking algorithm. For a rigorous proof of this fact, the reader may consult [66].
In any event, the consistency of the system is not at stake, since it follows as a corollary of canonicity, that is preserved.
Martin-Löf proposed another different presentation of intuitionistic type theory in [101]. This is known as the *logical framework* presentation of type theory. The idea is that a logical framework provides a setting to specify deductive systems. Basically, these systems are given by extending the core

LF by means of constants and equations. There exists an extensive literature on the subject, since logical frameworks can be used also to define other formal systems and not only versions of type theory. The interested reader may refer to [58, 120] for a general overview and to [113] for a presentation of type theory based on a logical framework. De Bruijn's AUTOMATH can be legitimately regarded as an ancestor of modern logical frameworks.
In LF-type theory two levels are distinguished: one is the level of types and the other is the level of sets. The theory of types consists of Π-types and a universe *Set*. The universe is equipped with the rule if $A : Set$, then $\mathsf{El}(A)$ *type*. The Π-type of LF is distinguished from the dependent products at the object-level and is written as $[x : A]B$, while abstraction is written as $[x]b : [x : A]B$. Moreover, β, η, ξ and α (renaming) rules are assumed on the LF- Π. Then a set is specified as a constant with type *Set* and the constructors as well as the eliminators are encoded as typed constants. The equations express the relevant computation rules. All the problems with binding and substitution are handled at the level of the logical framework. Finally, in LF-type theory usually a notion of intensional equality is adopted, that turns out to be decidable.
If we want to follow closely Martin-Löf's conception of logic, then the correspondence between types and propositions must be taken as an *identification*. This identification is taken to be a core element of type theory and is the correct formal counterpart of the BHK explanation of logical constants:

> "I will use the term set, but you should understand it in the sense of individual domain, or quantificational domain: in first-order predicate logic, you always quantify over a domain, and Curry-Howard is a correspondence, or isomorphism, between propositions and sets in that sense. The Curry-Howard correspondence was to me from the beginning the natural completion of the Brouwer-Heyting-Kolmogorov interpretation of intuitionistic first-order predicate logic, just drawing the full consequences, so to say, of the Brouwer-Heyting-Kolmogorov interpretation." ([104], p.249-250)

This belief together with the idea that a domain of quantification is always a type is what led Martin-Löf to postulate a type of all types V with the axiom $V : V$ in the manuscript [96]. On a more pragmatic level, it is useful to allow quantification over propositions since in this way many notions can

be expressed within the system in a uniform way and the system itself is characterized by a minimal set of rules. Now quantifying over propositions implies that there is a type of all propositions, but propositions are identified with types and hence there is a type of all types. This V is itself a type, therefore $V : V$. However, it is well-known that this system runs into Girard's paradox and thus is inconsistent. According to Martin-Löf, the true meaning of Girard's Paradox is the following:

> "Now the paradox that Girard discovered in 1971 (Girard 1972), in the first version of constructive type theory, then called intuitionistic type theory, showed that Curry-Howard is incompatible with impredicativity: if you combine the Curry-Howard identification of propositions and sets with impredicativity, that is, the possibility of quantifying over propositions, over classes, over relations, and so on, you end up with a contradiction. So one of them has to go: either Curry-Howard has to go or there is some problem with impredicativity, with which there had been problems from the very beginning: when the notion itself was introduced by Russell in 1906 (Russell 1906), it was precisely because it was a problematic notion. So it seems fair to say that Girard's paradox put an end to the optimism about impredicative intuitionism in the 1960s,[...] So, hopeful as impredicative intuitionism had seemed for some time, during the approximate decade from Spector's paper to Girard's paradox, it nevertheless ended in disillusion." (*ibid.*)

Therefore, if the identification of propositions with types is regarded as a fundamental and indispensable tenet of intuitionism, then Girard's paradox is the tombstone on any genuine version of impredicative intuitionism. On the other hand, one may have a different take on the problem: certainly propositions are related to types, since a proposition can be regarded as the type of its proofs, but we are not committed to think of this relationship as an identification. Then, it becomes possible to design meaningful and consistent intuitionistic impredicative systems. The whole issue therefore reduces to the very fundamental question of how to think of propositions in an intuitionistic setting.

A higher-order system conceived along these lines is the *Calculus of Constructions* [27]. In this system, there is an impredicative universe of propositions Prop that is closed under arbitrary dependent products

$$\frac{\Gamma, x : A \vdash B(x) : \mathsf{Prop}}{\Gamma \vdash \Pi_{(x:A)} B(x) : \mathsf{Prop}}$$

Note that there is no restriction over A, that can be Prop itself and thus we can define an element belonging the universe of propositions by quantifying over the universe itself. Moreover if $p : \mathsf{Prop}$, then $\mathsf{Prf}(p)$ *type*, i.e. a proposition can be thought of as the type of its proofs. Since it is a higher-order system, then the other logical notions as well as inductive data types can be encoded by means of implication and higher-order universal quantification in a standard way. In the case of the existential quantifier, we have a notion of existence weaker than that occurring in Martin-Löf type theory, since it does not entail the validity of any choice principle[146]. In general, we cannot recover the usual induction principles from the impredicative encoding of inductive types and this has motivated the introduction of a modified version of the system with primitive inductive types that is known as the *Calculus of Inductive Constructions* [28, 119]. The French proof-assistant *Coq* is based on this system.

It is also possible to introduce a primitive type of propositions and, at the same time, retain predicativity. This is the choice made in the *Minimalist Foundations* [92, 89], where four basic sorts are included: small propositions, propositions, sets and collections. I will discuss this system later in Chapter 4.

There exists also a third way of thinking about propositions in type theory. We might say that a type P is a proposition whenever for any $x, y : P$, $\mathrm{Id}_P(x, y)$ holds. These types are known as *h-propositions* or *mere* propositions in Homotopy Type Theory (**HoTT**) [155]. **HoTT** is another variant of intuitionistic type theory that, roughly speaking, arises from viewing types as *spaces* and terms as *points*. Moreover, this system includes an axiom called *Univalence* asserting that equivalent types are equal and that was first considered by Voevodsky. Let us stop here for the moment, since the next chapter will provide a more systematic introduction to **HoTT**. But one point must be raised: the presence of an axiom breaks canonicity, since there might be terms derived from applications of the axiom that are computationally not effective. The problem of providing a computational interpretation of *Univalence* has led to the introduction of another family of type theories called *cubical type theories*[108]. The system of type theory is extended with a primitive *interval* $\mathcal{I}$ with two fixed endpoints and the basic forms of judgement

include judgements depending on elements in $\mathcal{I}$. Identity types are replaced by *path* types, where a path in A is a map from $\mathcal{I}$ to A. Moreover, path types are not inductively characterized. Finally, at least in some version of cubical type theory, Univalence turns out to be provable[5].

1.6 Historical Notes

Martin-Löf has given different formulations of his theory. This being the case, it is useful to list chronologically the main presentations of Intuitionistic Type Theory given by Martin-Löf and briefly summarize the most distinctive features of each presentation:

- $ML71$([96]): this is the first version of Intuitionistic Type Theory with the strong impredicative axiom $V : V$, where V is a type of all types. The impredicative features of the system allow for the assumption of few primitive symbols, since the other notions can be encoded. A normalization argument is provided for this system, but, as we have seen above, it is inconsistent. All the subsequent presentations of type theory avoid impredicative principles like $V : V$ and gain expressive strength by introducing predicative universes.

- $ML72$ ([102]): judgemental equality is not included among the primitive forms of judgement and an untyped notion of conversion is assumed. This system contains rules for all the most familiar type-constructors, but not for identity types. A proof of normalization by the method of computability is given and the relationship with other formal theories is investigated.

- $ML73$ ([97]): this version includes judgemental equality as basic form of judgement and contains the first exposition of the rules for Id-types. Also the relation between the two notions of equality is discussed. A different proof of normalization is presented with respect to a modified reduction relation. In particular, the ξ-rule as well as the η-rules are thrown out.

[5] A proof of normalization (and its corollaries) for a variant of cubical type theory can be found in [145].

- $ML79$ ([98]): it is the first exposition of the extensional theory. The rules for propositional equality are crucially modified, since the elimination rule is now given by the reflection rule. Many properties of the intensional system are lost (e.g. non-normalizable terms are typable and judgemental equality is undecidable). The system is endowed with an intended semantics represented by the meaning explanations. However, canonicity is retained and thus also the logical consistency of the system.

- $ML86$ ([101]): first presentation based on the logical framework. There is a systematic use of the higher-order notation and the theory is again intensional. This presentation contains a theory of types (Set and the dependent product written as $(x : A)B(x)$ to distinguish it from the dependent product at the set-level) and a theory of sets. Sets are introduced by means of typed constants that make up the signature of the theory and the behaviour of these constants is governed by suitable definitional equations.

Martin-Löf Type Theory was introduced in a prosperous period for proof theory, constructive mathematics and the foundations of mathematics in general. In fact, the theory can be regarded as the synthesis of different trends that were emerging in those years. Since nowadays type theory is utterly important both as foundational system for constructive mathematics and as a framework for the implementation of mathematics in proof-assistants, it is instructive to give a panoramic view of the historical context in which the theory was worked out.
My claim is that Martin-Löf type theory can be regarded as the synthesis of three, partly overlapping, important strands in logic and foundations of mathematics:

1. The development of *general* proof-theory and thus the study of normalization results for various systems as well as the emergence of the proofs-as-programs paradigm.

2. The renewed interest in constructive mathematics, mainly in the computational and neutral form advocated by Bishop [15].

3. The pioneering investigations about mechanical formalization of mathematics and formal proof-checking (chiefly, De Bruijn's AUTOMATH project [36]).

In the following, I shall give a brief reconstruction of each of them.
(1). After Gentzen's seminal contributions, natural deduction was a neglected subject for a while. It was Prawitz's monograph *Natural Deduction: A Proof-Theoretical Study* [126] that revived the interest for this topic. Prawitz gave normalization results for a number of logical calculi and clarified the relationship between introduction and elimination rules already observed by Gentzen. It was in this context that he introduced the *inversion principle*, that underlies the definition of the reduction steps and thus the whole normalization procedure.
Around the same time, the correspondence between proofs and programs and between formulas and types begun to be intensively explored. It is well-known that the correspondence for a limited fragment of intuitionistic logic had already been pointed out by Curry in [34, 35] within the context of combinatory logic. However, it was only at the end of the sixties that the correspondence was extended to more expressive systems and investigated in a more systematic manner. The paper by Howard [68], which at first circulated only privately, represented a seminal contribution to this area and highlighted the depth of the correspondence.
The interest in the relation between natural deduction systems and typed calculi culminated in the systematic study of normalization procedures for several different theories. In particular, the work by Tait on Gödel's System **T** [152] paved the way for analogous results about more complex systems. Gödel had introduced [56] the system **T** to provide an interpretation of Heyting Arithmetic into a quantifier-free system of functionals of finite type. This system can be presented as an extension of simply-typed λ-calculus with a type of natural numbers. Moreover, the normalization theorem for this system implies the consistency of Heyting Arithmetic (and thus, by the double-negation translation, of Peano Arithmetic). In order to prove normalization for System **T**, Tait introduced the so-called *reducibility* or *computability* predicates. A set of 'reducible' terms is associated to each type symbol and such a set is defined by induction over the structure of the type. Moreover, these sets satisfy appropriate closure conditions. Then the key result consists in establishing that all the terms of the system are reducible and hence, by definition of reducibility, strongly normalizing.
The method introduced by Tait was extended to several different systems (see, for instance, the contributions by Girard, Prawitz and Martin-Löf to the volume [52]). A noteworthy case is Girard's proof of strong normalization for System **F**. System **F** is an extension of simply-typed λ-calculus with

rules for quantifying over type variables. Therefore, it is a second-order calculus. Tait's method cannot be extended to this system in a direct and straightforward way. Handling naively the universal quantification over types yields a circular, ill-founded definition. To overcome this difficulty, Girard introduced *reducibility candidates* [55] and by means of these was able to derive a strong normalization proof for his system. The argument had as consequence a positive answer to *Takeuti's Conjecture*, stating that second-order logic (and higher-order logic in general) enjoys cut-elimination. Before Girard's normalization proof, the only proofs of this fact were semantical [151, 122, 154].

From the point of view of the development of type theory, the paper [95] by Martin-Löf on iterated inductive definitions is worth mentioning. There, an intuitionistic theory of inductive definitions is investigated and proved to be normalising. More interesting from our point of view it is the fact that Martin-Löf provides a general schema for inductively defined predicates, that anticipates the representation of inductive types in Intuitionistic Type Theory and has been quite influential in the later literature about general schemes for inductive definitions.

Finally, it must be noted that all these impressive developments in proof theory were also subject of careful philosophical analysis. In particular, a leading motive was to extract genuinely semantical concepts from normalization arguments in order to work out a valuable alternative to model-theoretic semantics [6]. In particular, both Dummett's semantical argument rejecting classical logic in favour of intuitionistic logic [39] as well as his general program of working out an anti-realistic theory of meaning [41, 40] were partly inspired by developments in proof theory. The notion of harmony as well as the fundamental assumption (that is, technically, a corollary of normalization) are both concepts taken from proof theory and regarded by Dummett as fundamental in the articulation of a correct theory of meaning. Prawitz's program of proof-theoretic semantics [123, 124, 125] bears an even stronger relationship with technical work in proof theory. It is fair to say that the concept of proof-theoretic validity is a refinement of the computability method in normalization arguments. In fact, Prawitz considered the development of a proof-theoretic semantics and the associated alternative characterization of fundamental logical notions such as validity and consequence one of the main

[6] See [150] for a detailed reconstruction of the relationships between proof theory and theory of meaning.

aims of general proof theory.
(2). Errett Bishop's book *Foundations of Constructive Analysis* [15] represented a fresh start for constructive mathematics. This work developed constructively many fundamental mathematical topics (from basic real analysis to functional analysis) avoiding the classically unacceptable consequences of traditional intuitionist mathematics. Since the book is mainly devoted to the development of constructive analysis, it does not expose any formal system in which all these definitions and results might be formalized. Nevertheless, Bishop discusses many topics of foundational interest and outlines the philosophical principles underlying his conception of constructive mathematics [7]. Two distinctive features of Bishop's conception are the thesis according to which the meaning of any mathematical statement is ultimately reduced to *numerical* meaning and the idea that *axiomatic freedom* is what characterizes this new form of constructive mathematics. The first thesis is the main motivation behind Bishop's rejection of the theory of choice sequences and thus of the Brouwerian conception of the continuum. The problem with choice sequences is that we cannot make sense of statements about them in terms of numerical computation. Also it is precisely this view of the meaningfulness of mathematical statements that gives to Bishop's constructive mathematics a strong *computational* flavour. This aspect is manifest in some definitions and distinctions proposed by Bishop such as that between *operations* and *functions*:

> "An *operation* from a set A to a set B is a rule f which assigns an element $f(a)$ of B to each element a of A. The rule must afford an explicit, finite, mechanical reduction of the procedure for constructing $f(a)$ to the procedure for constructing a. The set A is called the *domain* of f. The most important case occurs when
>
> $$f(a_1) = f(a_2)$$
>
> whenever a_1 and a_2 are equal elements of A. Such an operation f is called a *function*." ([15], p.14)

Here *rule* must be understood as a primitive notion and corresponds to the notion of *method* occurring in the intuitionistic explanation of logical constants. A function then is an operation that respects the notion of equality on a given

[7] See [32] for a systematic account of Bishop's philosophical thought.

domain. Bishop's characterization of functions relies upon his notion of *set*. Asserting that a set exists means that it has been properly defined. In fact, a set is given by prescribing how to construct an element belonging to it and what we have to do to show that two elements in the set are equal. Bishop's sets are sometimes identified with *setoids* in the literature.

The other hallmark of Bishop's approach, namely axiomatic freedom, can be explained as follows. Dropping classical principles as well as anti-classical ones such as continuity principles or Church's Thesis leads to a form of constructive mathematics that is *neutral* and more *general*. By neutrality, we mean that it is *compatible* with classical mathematics as well as with other brands of constructivism. Then axiomatic freedom consists in the possibility of isolating a 'minimal' set of principles that are compatible with other foundational frameworks, that must be regarded as further specifications of the general setting. They are mathematical 'worlds' arising from specific assumptions, that must not be taken as absolute. Therefore, the mathematician can freely explore these different worlds, but without being committed to any of them. The remarkable fact is that this perspective brings to light distinctions (e.g. operation and function) that are lost in presence of stronger assumptions and thus paves the way for a more fine-grained analysis of mathematical concepts. In this sense, the principle D of [16], asserting that "meaningful distinctions deserve to be preserved", is satisfied.

The years between the end of the sixties and the beginning of the seventies saw an intense development of metamathematical investigations of intuitionistic systems. Techniques such as Kleene's realizability and variants thereof were refined and successfully applied to a rich variety of systems in order to establish their metamathematical properties and to obtain consistency results for principles usually assumed in some form of constructive mathematics. The new methods developed in proof theory in the study of normalization proofs were also fruitfully applied to these systems. The encyclopedic book [156] represents a detailed exposition of this trend.

The new form of constructive mathematics as advocated by Bishop also gave an impulse to the investigation of formal systems for constructive mathematics. Since Bishop did not present his mathematics in a well-defined framework, there were different attempts to define formal systems in which Bishop's mathematics could be formalized. In fact, new systems for constructive mathematics were introduced in those years, including Feferman's various systems of *explicit mathematics* [46], Myhill's constructive set theory[110] and Martin-Löf Type Theory itself. Martin-Löf, indeed, presents his system as a

suitable framework for formalizing Bishop's constructive mathematics [102]. The interest in formal systems for intuitionistic and, in general, constructive mathematics was also motivated by an attempt to clarify the basic notions of intuitionism and put them on more solid theoretical grounds. In this respect, a fundamental role is played by Kreisel, who was also one of the leading figures in the contemporary developments of proof theory. According to Kreisel, fundamental intuitionistic notions such as proof or method were to be carefully analyzed in order to arrive at a more faithful formalization of them. In particular, the whole problem of the foundations of intuitionism should be dealt with by avoiding any classical assumption:

> "Was the (logical) language of the current intuitionistic systems obtained by *uncritical* transfer from languages which were, tacitly, understood classically?" ([81], p.283)

To this end, Kreisel devoted his efforts to formalize an abstract theory of constructions [79], that should provide a framework for reasoning directly about proofs or constructions and rules. The theory was meant to provide a faithful formalization of the intuitionistic explanation of logical constants, with an amendment of the clause for implication. The idea was to avoid the alleged impredicativity of the proof-interpretation by rephrasing the condition as follows: a proof of $A \to B$ consists of a method which transforms any proof of A into a proof of B together with a proof that this method satisfies the desired property. Kreisel's attempt was unsuccessful and also the *second clause* explanation of the implication connective was not generally accepted. Nevertheless, it is enlightening to consider the way Kreisel framed the question of an adequate foundation for intuitionism:

> "Intuitionistic mathematics, as developed by Brouwer and Heyting, has two aspects. Its negative aspect, which is well-known, rejects the basic notions of set-theoretic (or classical) mathematics. The positive aspect is this: the notions of 'construction (constructive function)' and 'constructive proof (of equality between two constructions)' are regarded as sufficiently clear for at least a part of mathematics to be built up systematically from some evident assertions about these notions. This aspect has been largely disregarded, probably because of current preoccupation with formal and first-order aspects of mathematics. " ([79], p.198)

Kreisel's program was taken up by Scott in [140]. Admittedly, it is a partial attempt, but it shares many features with the intuitionistic theory of types as worked out by Martin-Löf. In particular, Scott includes dependent types and also transfinite types (following [153]). At the end of the paper, he also mentions the possibility of extending the system with a kind of reflection principles, that anticipate the role played by universes in Martin-Löf's system. An interesting point is that Scott's system does not assign any role to judgements and to definitional equality. Nevertheless, Scott himself comments on this point by remarking the fact that Kreisel emphasized the importance of the notions of judgement and definitional equality in theories dealing with proofs or constructions as main objects. In this perspective, Martin-Löf's theory fully satisfies Kreisel's conditions and represents the most elaborated attempt to design a system where proofs are taken as first-class citizens.
(3). The third trend concerning computer formalization and formal verification of mathematical proofs was still in an embryonic form, when intuitionistic type theory was introduced. Scott's paper on the theory of constructions credited De Bruijn for the introduction of the dependent product type in the context of his project AUTOMATH [36]. There, the notation $[x : A]B$ is introduced for both the term $\lambda(x : A)B$ and the type $\Pi_{x:A}B$. The groundbreaking role of AUTOMATH can hardly be underestimated: it was the first formal language built over the propositions-as-types principle with the aim of mechanically checking the correctness of proofs formalized in it. In De Bruijn's own words: "the rules [of the language] are such that a computer can be instructed to check whether texts written in the language are correct" ([36], p.2). It is noteworthy that Martin-Löf himself pointed out the relationship between intutionistic type theory (and, in particular, the decidability of type-checking) and the computer implementation of proofs formulated in this system. However, while entertaining this possibility, he had a more cautious attitude towards the potential usefulness of these applications:

> "What is doubtful at present is not whether computerization is possible, because we already know that, but rather whether these proof-theoretical computation procedures are at all useful in practice. So far, they do not seem to have found a single significant application." ([97], p. 117)

However, a few years later Martin-Löf emphasized the link of constructive mathematics as developed in a type-theoretical setting with programming

[98] to the point of asserting that constructive mathematics *is* computer programming, if the latter is understood in an appropriate theoretical way.

Chapter 2

Homotopy Type Theory and the Univalence Axiom

Homotopy Type Theory (**HoTT**) [155] has represented a truly groundbreaking contribution to the field of type theory and has brought to light the (unexpected) relationship between intensional type theory, on one side, and higher categories and abstract homotopy theory, on the other. A pioneering work in this direction is the *groupoid* model designed by Hofmann and Streicher to give an example of semantics in which the principle of uniqueness of identity proofs fails [67]. However, the first works in which such a perspective was more clearly articulated are the notes [159] and the paper [12]. These seminal contributions have paved the way for a whole new research area, that is still flourishing. The aim of the following chapter is to introduce the basics of **HoTT** and highlight the new concepts that have been introduced in this context such as the Univalence Axiom and the Higher Inductive Types (HIT).

2.1 The true nature of Identity

The extensional rules for propositional equality entail that any two proofs of an identity type are equal. Such rules are validated in the set-theoretic interpretation of type theory, where types are interpreted as sets. It is well-known from [141] (modulo some coherence issues [33, 64, 23]) that extensional type theory (without universes) is the internal language of *locally cartesian closed categories* (lccc). A category $\mathcal{C}$ is locally cartesian closed if each slice $\mathcal{C}/A$ is cartesian closed. Moreover, if $\mathcal{C}$ has a terminal object, then $\mathcal{C}$ itself is

cartesian closed. A dependent type $B(x)$ $[x : A]$ is interpreted as an arbitrary map in $\mathcal{C}$ $[\![B]\!] \xrightarrow{b} [\![A]\!]$ or, equivalently, as an object in $\mathcal{C}/A$. In this context, the interpretation of propositional equality as equalizer is somewhat forced and thus the reflection rule is valid: because we have the arrow

$$[\![\Gamma.\mathrm{Id}_A(a,b)]\!] \xrightarrow{e} [\![\Gamma]\!]$$

and two parallel arrows a and b going from $[\![\Gamma]\!]$ to $[\![\Gamma.A]\!]$. Further, there is a section p of e,

$$[\![\Gamma]\!] \xrightarrow{p} [\![\Gamma.\mathrm{Id}_A(a,b)]\!]$$

Then, since p is a section of e and e is mono, it follows that e is an isomorphism and hence $a = b$.

In the case of intensional type theory, there is a *distinguished* class of maps representing dependent types. They are called *display maps* or *fibrations*. Then identity types are interpreted as path space fibrations $PA \to A \times A$ and the diagonal $\delta_A : A \to A \times A$ factors as a map $r : A \to PA$ followed by the fibration $PA \to A$. The map r is an example of *acyclic cofibration* or *anodyne* map. In a more syntactic language, the path space of A can be instantiated by the total space of the identity type, which is $\Sigma_{(x,y:A)} x =_A y$ and the map from A to the total space is just $\lambda x.\langle x, x, \mathsf{refl}_x \rangle$. Very roughly, equipping type theory with rules for identity types amounts to assuming the existence of a *weak factorization system*. In fact, the main result of [53] asserts that the syntactic category of a type theory including the intensional identity types is endowed with a weak factorization system. Conversely, in [12] it is shown that any Quillen model category provides a setting for interpreting intensional type theory. These models do not validate the reflection rule and hence show that intensional identity types support a non-trivial higher-dimensional structure.

In the following, some basic facts about propositional equality are introduced that may help to understand why types are higher-groupoids and how this higher-dimensional structure is generated from the rules for Id-types. Let us collect what we know so far about (intensional) propositional equality: we know that it is an equivalence relation and that a proof p that x and y are equal in A induces a map p_* between instances of a type family over A at x and y, respectively. This second fact is known as transport, while the data provided by the fact that it is an equivalence relation can be called, respectively, *constant path* (reflexivity), *inverse path* (symmetry) and *concatenation of paths* (transitivity), where a path p is a term of $\mathrm{Id}_A(x, y)$ that can be regarded as a way of going from a certain point x to another point y. Then we may

expect that these operations on paths behave in a natural way: concatenation should be associative and should satisfy the unit laws and inverses should be 'true' inverses, that is, two-sided ones. Further, the operation of taking the inverse path should be idempotent. It is the case that all these facts hold as the following lemma shows:

Lemma 2.1.1. *Let A be a type and $x, y, z, w : A$. Further, let us suppose that the following equalities hold: $p : \mathrm{Id}_A(x, y), q : \mathrm{Id}_A(y, z)$ and $r : \mathrm{Id}_A(z, w)$. Then there are constructions* ass, u_1, u_2, linv, rinv, idem *such that the following judgements are derivable:*

a) $\mathsf{ass}(p, q, r) : (p \blacksquare q) \blacksquare r = p \blacksquare (q \blacksquare r)$;

b) $\mathsf{u}_1(p) : p = p \blacksquare \mathsf{refl}(y)$ *and* $\mathsf{u}_2(p) : p = \mathsf{refl}(x) \blacksquare p$;

c) $\mathsf{linv}(p) : p^{-1} \blacksquare p = \mathsf{refl}(y)$ *and* $\mathsf{rinv}(p) : p \blacksquare p^{-1} = \mathsf{refl}(x)$;

d) $\mathsf{idem}(p) : (p^{-1})^{-1} = p$.

Remark 2.1.2. The type of these equalities is left implicit, but they are all equalities between equalities and thus the type can be easily inferred. Further, all the proofs make use of the induction principle associated to the identity types, but there are different possibilities according to how this principle is applied which yield different proofs. The resulting proofs are not judgementally equal, but it can be shown that they are still propositionally equal.

Proof. a). We want to build an element $\mathsf{ass}(p, q, r)$ of type

$$C(x, y, p) \ :\equiv \ \Pi_{(z:A)}\Pi_{(q:x=z)}\Pi_{(w:A)}\Pi_{(r:z=w)}(p \blacksquare q) \blacksquare r = p \blacksquare (q \blacksquare r)$$

Then by Id-E, we know that it is enough to build an element of $C(x, x, \mathsf{refl}(x))$ that is

$$\Pi_{(z:A)}\Pi_{(q:x=z)}\Pi_{(w:A)}\Pi_{(r:z=w)}(\mathsf{refl}(x) \blacksquare q) \blacksquare r = \mathsf{refl}(x) \blacksquare (q \blacksquare r)$$

By Id-C, we get $\mathsf{refl}(x) \blacksquare q \equiv q$ and hence the left side is judgementally equal to $q \blacksquare r$. Similarly, on the right-hand side. Then we end up with $\mathsf{refl}(q \blacksquare r) : q \blacksquare r = q \blacksquare r$, that is our $\mathsf{ass}(\mathsf{refl}(x), q, r)$. Finally, by Id-E, we get $\mathsf{ass}(p, q, r) : C(x, y, p)$ as required.

b). Let $C(x, y, p) :\equiv (p = p \blacksquare \mathsf{refl}(y))$. Then $C(x, x, \mathsf{refl}(x))$ amounts to $(\mathsf{refl}(x) = \mathsf{refl}(x) \blacksquare \mathsf{refl}(x))$. Since the right-hand side is judgementally equal to $\mathsf{refl}(x)$, we can conclude

$$\mathsf{refl}(\mathsf{refl}(x)) : \mathsf{refl}(x) = \mathsf{refl}(x)$$

that is $\mathsf{u}_1(\mathsf{refl}(x))$ and hence by Id-E, we get $\mathsf{u}_1(p) : C(x, y, p)$. The judgement $\mathsf{u}_2 : p = \mathsf{refl}(x) \blacksquare p$ is established by a similar reasoning.

c). Let $C(x, y, p) :\equiv (p^{-1} \blacksquare p = \mathsf{refl}(y))$. Then $C(x, x, \mathsf{refl}(x))$ amounts to

$$\mathsf{refl}(x)^{-1} \blacksquare \mathsf{refl}(x) = \mathsf{refl}(x)$$

By the computation rule for identity,

$$\mathsf{refl}(x)^{-1} \blacksquare \mathsf{refl}(x) \equiv \mathsf{refl}(x) \blacksquare \mathsf{refl}(x) \equiv \mathsf{refl}(x)$$

and hence we can conclude $\mathsf{refl}(\mathsf{refl}(x)) : \mathsf{refl}(x) = \mathsf{refl}(x)$, where $\mathsf{refl}(\mathsf{refl}(x))$ is our $\mathsf{linv}(\mathsf{refl}(x))$ and thus by Id-E, we get $\mathsf{linv}(p) : C(x, y, p)$. The judgment $\mathsf{rinv}(p) : p \blacksquare p^{-1} = \mathsf{refl}(x)$ is derived by a similar reasoning.

d). Let $C(x, y, p) :\equiv ((p^{-1})^{-1} = p)$, then

$$C(x, x, \mathsf{refl}(x)) :\equiv ((\mathsf{refl}(x)^{-1})^{-1} = \mathsf{refl}(x))$$

But $\mathsf{refl}(x)^{-1} \equiv \mathsf{refl}(x)$ and also $\mathsf{refl}(x) \blacksquare \mathsf{refl}(x) = \mathsf{refl}(x)$, therefore we can conclude

$$\mathsf{idem}(\mathsf{refl}(x)) : \mathsf{refl}(x) = \mathsf{refl}(x)$$

where $\mathsf{idem}(\mathsf{refl}(x)) :\equiv \mathsf{refl}(\mathsf{refl}(x))$. Finally, by Id-E, we get $\mathsf{idem}(p) : C(x, y, p)$. □

In the light of the previous remarks, a type A can be conceived as a *space* and the identity type Id_A as the *path space* associated to A. Also there is no reason to stop here. In a proof-relevant and intensional setting, all these operations on paths are not mere properties, actually they define structures over paths. Since they are all propositional equalities, they will be inhabited by possibly multiple proof-terms and therefore they must satisfy other laws in the next dimension and so on. These operations can be interpreted also as saying: reflexivity corresponds to the identity morphism, while symmetry

and transitivity correspond, respectively, to the inverse morphism and to the composition of morphism. Then the interpretation of the conditions in Lemma 2.1.1 is straightforward. Under this reading, a type A is just a category in which every morphism has an inverse and this is exactly the definition of a *groupoid*. But since the identity types are proof-relevant, we can build morphisms between morphisms and so forth. Therefore, a type is not just a groupoid, but a higher-groupoid or ∞-groupoid. Since all the equations corresponding to the coherence conditions at each dimension are valid only propositionally, then the resulting structure is that of a *weak* ∞-groupoid.
It can be useful to visualize what's going on as follows: we start from two points

$$a, b : A$$

in a space A and then we build the space of paths associated to A that turns out to be inhabited by two paths p and q

$$p, q : \mathrm{Id}_A(a, b)$$

then we can iterate the construction and obtain

$$\alpha, \beta : \mathrm{Id}_{\mathrm{Id}_A(a,b)}(p, q)$$

where α, β are paths between paths, that is *homotopies*. Then we can still iterate the process and obtain new higher paths and so forth. In this way, the result is a tower of identity types and at each level of the tower some coherence conditions are satisfied up to higher homotopies. The fact that Id-types endow each type with such a rich and intricate structure is the key insight underlying Homotopy Type Theory.

2.2 The Homotopical Perspective

The homotopical perspective introduced in the previous section can be extended to other type-theoretical concepts as well. A type family $B : A \to \mathcal{U}$ can be thought of as a *fibration* with A being the *base* of the fibration. The application of the Σ-type to B gives the *total space* of the fibration represented by $\Sigma_{(x:A)}B(x)$. It follows that a dependent term can be regarded as a *section* of a given fibration.
Under this light, it becomes quite natural to regard a map $f : A \to B$ between types as a functor that preserves the relevant structure. This is essentially the content of the following lemma:

Lemma 2.2.1. *Let $f : A \to B$, $g : B \to C$, $x, y, z : A$ and $p : x = y, q : y = z$, then the following hold:*

a) $\mathsf{ap}_f : \Pi_{(x,y:A)}(x = y) \to (f(x) = f(y))$;

b) $\mathsf{ap}_{id_A}(p) = p$:

c) $\mathsf{ap}_g(\mathsf{ap}_f(p)) = \mathsf{ap}_{g\circ f}(p)$;

d) $\mathsf{ap}_f(p \blacksquare q) = \mathsf{ap}_f(p) \blacksquare \mathsf{ap}_f(q)$;

e) $\mathsf{ap}_f(p^{-1}) = \mathsf{ap}_f(p)^{-1}$

Proof. a). Let $C(x, y, p) :\equiv f(x) = f(y)$. Then we have to construct an element of $C(x, x, \mathsf{refl}(x))$, that is given by $\mathsf{refl}(f(x))$. Then by Id-E, we get $\mathsf{ap}_f(p) : C(x, y, p)$.
b). Let $C(x, y, p) :\equiv \mathsf{ap}_{id_A}(p) = p$. Then $C(x, x, \mathsf{refl}(x))$ is inhabited by $\mathsf{refl}(\mathsf{refl}(x))$ and then the result follows by Id-E.
c). Let $C(x, y, p) :\equiv \mathsf{ap}_g(\mathsf{ap}_f(p)) = \mathsf{ap}_{g\circ f}(p)$. Then,

$$C(x, x, \mathsf{refl}(x)) :\equiv \mathsf{ap}_g(\mathsf{ap}_f(\mathsf{refl}(x))) = \mathsf{ap}_{g\circ f}(\mathsf{refl}(x))$$

Then by a), this reduces to $\mathsf{refl}(g(f(x))) = \mathsf{refl}(g(f(x)))$, that is inhabited by $\mathsf{refl}(\mathsf{refl}(g(f(x))))$. Therefore, the result follows by Id-E.
d). Let $C(x, y, p) :\equiv \mathsf{ap}_f(p \blacksquare q) = \mathsf{ap}_f \blacksquare \mathsf{ap}_f(q)$. Then

$$C(x, x, \mathsf{refl}(x)) :\equiv \mathsf{ap}_f(\mathsf{refl}(x) \blacksquare q) = \mathsf{ap}_f(\mathsf{refl}(x)) \blacksquare \mathsf{ap}_f(q)$$

This amounts to $\mathsf{ap}_f(q) = \mathsf{ap}_f(q)$, which is inhabited by $\mathsf{refl}(\mathsf{ap}_f(q))$ and then the result follows by Id-E.
e). Let $C(x, y, p) :\equiv \mathsf{ap}_f(p^{-1}) = \mathsf{ap}_f(p)^{-1}$.
Then an element of $C(x, x, \mathsf{refl}(x))$ is given by $\mathsf{refl}(\mathsf{refl}(f(x)))$. Therefore, the result follows by Id-E. □

Transport allows us to compare terms that live in different types. In fact, if $f : \Pi_{(x:A)}B(x)$ is a dependent map and p is a path from x to y in A, then $f(x)$ and $f(y)$ are in different types, but we can transport $f(x)$ along p so as to obtain a new element in $B(y)$, that can be directly compared to $f(y)$. We can say even more, since this element turns out to be equal to $f(y)$:

Lemma 2.2.2. *Let $f : \Pi_{(x:A)}B(x)$ and $p : x = y$ in A, then the following holds:*

$$\mathsf{apd}_f(p) : p_*(f(x)) = f(y).$$

Proof. Let $C(x, y, p) :\equiv p_*(f(x)) = f(y)$. Then it is enough to construct a term of $C(x, x, \mathsf{refl}(x))$, that is $\mathsf{refl}_*(f(x)) = f(x)$. Since $\mathsf{refl}_*(f(x)) \equiv f(x)$, then $\mathsf{apd}_f(\mathsf{refl}(x)) :\equiv \mathsf{refl}_{f(x)} : f(x) = f(x)$. Therefore, we get $\mathsf{apd}_f(p) : p_*(f(x)) = f(y)$. □

Therefore, transport allows a generalization to the dependent case of the fact that maps between types are 'continuous', i.e. they respect propositional equality. Further, it holds that the composition of transports is the transport of the concatenation:

Lemma 2.2.3. *Let $t : P(x)$ $[x : A]$ and $p : x =_A y, q : y =_A z$. Then the following holds:*

$$\mathsf{trp}(p, q) : q_*(p_*(t)) = (p \blacksquare q)_*(t)$$

Proof. Let $C(x, y, p) :\equiv q_*(p_*(t)) = (p \blacksquare q)_*(t)$. Then it is enough to construct an element of $q_*(\mathsf{refl}(x)_*(t)) = (\mathsf{refl}(x) \blacksquare q)(t)$. This reduces to $q_*(t) = q_*(t)$ which is inhabited by $\mathsf{trp}(\mathsf{refl}(x), q) :\equiv \mathsf{refl}(q_*(t))$. Then we conclude by Id-E. □

Furthermore, we can show that the dependent case and the non-dependent one are related, since the former can be obtained by concatenating the latter with an appropriate term. This means that we can solve the equation

$$\mathsf{apd}_f(p) = ? \blacksquare \mathsf{ap}_f(p).$$

Lemma 2.2.4. *Let B be a type and $x, y : A$. If $p : x =_A y$ and $b : B$, then*

$$\mathsf{transpconst}^B_p(b) : p_*(b) =_B b.$$

Proof. Let $C(x, y, p) :\equiv p_*(b) = b$. Then $\mathsf{refl}(x)_*(b) = b$ and we know that $\mathsf{refl}(x)_*(b) \equiv b$, therefore it boils down to $b = b$, which is obviously inhabited by $\mathsf{refl}(b)$. Then the result follows by Id-E. □

The term $\mathsf{transpconst}$ is what we need to connect apd with ap:

Lemma 2.2.5. *Let $f : A \to B$, $x, y : A$ and $p : x =_A y$. Then the following type is inhabited*

$$\mathsf{apd}_f(p) = \mathsf{transpconst}^B_p(f(x)) \blacksquare \mathsf{ap}_f(p).$$

Proof. The proof consists again in a simple application of Id-E rule. □

2.3 Homotopy Levels & HIT

The fruitfulness of the homotopical interpretation of type theory is also highlighted by the fact that many concepts and constructions from abstract homotopy theory and higher category theory can be directly formalized and applied in type theory. Better, type theory can be regarded as a synthetic language to handle these higher-dimensional structures. The classification of types into the hierarchy of *homotopy* levels (h-levels) is motivated by similar definitions appearing in those fields.
In higher category theory, it is usual to define a n-truncated object starting from $n = -2$. Similarly, we could define a predicate ntype over types by recursion by requiring that

$$(-2)\mathsf{type}(A) \;:\equiv\; \mathsf{isContr}(A)$$

and

$$(\mathsf{n}+1)\mathsf{type}(A) \;:\equiv\; \Pi_{(x,y:A)}(\mathsf{n})\mathsf{type}(\mathrm{Id}_A(x,y))$$

However, making rigorous this definition in type theory requires the definition of an inductive type $\mathbb{N}_{-2}$ with the constructors $-2 : \mathbb{N}_{-2}$ and $\mathsf{sc} : \mathbb{N}_{-2} \to \mathbb{N}_{-2}$. Then `ntype` is defined as a predicate with type $\mathbb{N}_{-2} \to \mathcal{U} \to \mathcal{U}$. Of course, this does not pose any significant problem, but is more convenient to give an inductive definition by making use of the type $\mathbb{N}$. Then we introduce a predicate $\mathsf{hlevel} : \mathbb{N} \to \mathcal{U} \to \mathcal{U}$ that assigns to any type is homotopy level. The two alternatives are clearly related, since there is a mapping $\mathsf{ntype} \mapsto \mathsf{hlevel}_{n+2}$. Therefore, we arrive at the following definition

Definition 2.3.1. The predicate $\mathsf{hlevel} : \mathbb{N} \to \mathcal{U} \to \mathcal{U}$ is recursively defined as follows:

$$\mathsf{hlevel}_0(A) \;:\equiv\; \mathsf{isContr}(A)$$

$$\mathsf{hlevel}_{n+1}(A) \;:\equiv\; \Pi_{(x,y:A)}\mathsf{hlevel}_n(\mathrm{Id}_A(x,y))$$

A type A is said to be of h-level n if the type $\mathsf{hlevel}_n(A)$ is inhabited.

Making sense of this definition requires to explain what is the predicate $\mathsf{isContr}(-)$ that occurs in the base case:

Definition 2.3.2. A type A is said to be *contractible* if the following type is inhabited

$$\mathsf{isContr}(A) \;:\equiv\; \Sigma_{(x:A)}\Pi_{(y:A)}\mathrm{Id}_A(x,y)$$

Hence, a contractible type is a type inhabited by an element x such that any other element of the type is equal to x.
In standard formulations of first-order logic, no distinction is made between multiple proofs of the same proposition. We care about provability, but not about proofs. This fact can be formalized in type theory by assuming that a type is a proposition if any two elements in it are equal. The resulting notion of proposition is different from that deriving from the identifications of propositions with types and thus we can give a different characterization of logical notions in this context[1].

Definition 2.3.3. A type A is said to be a *h-proposition* or a *mere* proposition if the following type is inhabited

$$\mathsf{isProp}(A) \ :\equiv \ \Pi_{(x,y:A)}\mathrm{Id}_A(x,y)$$

Lemma 2.3.4. *The following are logically equivalent:*

a) A is contractible;

b) A is a h-proposition and there exists $a : A$.

Proof. If A is contractible, then for any $x, y : A$, $x = a = y$ and thus A is a h-proposition. Conversely, if A is a h-proposition and $a : A$, then for any other $x : A$, $x = a$ holds and hence A is contractible. □

Contractible types and h-propositions are the levels 0 and 1, respectively, of the homotopy hierarchy. At level two, we find those types whose identity types are h-propositions. They are called *h-sets*, since they represent the type-theoretic counterpart of the usual sets in set theory.

Definition 2.3.5. A type A is said to be a *h-set* if the following type is inhabited

$$\mathsf{isSet}(A) \ :\equiv \ \Pi_{(x,y:A)}\mathsf{isProp}(\mathrm{Id}_A(x,y))$$

Therefore, a h-set is a type in which any two proofs of an equality between elements of A are equal. H-sets can be equivalently characterized as those types satisfying the Axiom K:

Lemma 2.3.6. *Let A be a set. Then the following are equivalent:*

[1]The resulting logic must not be confused with the way logic is dealt with in type theories including a primitive type of propositions.

a) A is a h-set;

b) A satisfies Axiom K:

$$\Pi_{x:A}\Pi_{p:x=_A x} p = \mathsf{refl}(x).$$

Proof. If A is a h-set, then $x = x$ is a h-proposition and thus the Axiom K is immediately satisfied. On the the other hand, if A satisfies K, then given $p, q : x = y$, we get that $p \blacksquare q^{-1} = \mathsf{refl}(x)$ and thus $p = \mathsf{refl}(x) \blacksquare q$. But this implies $p = q$ and thus that A is a h-set. □

Most set-theoretic, traditional mathematics takes place at these first levels of the homotopical hierarchy, but the perspective motivated by abstract homotopy theory and higher category theory suggests that we should not stop here. For example, we can give a simple definition of h-groupoids as those types whose identity types are h-sets:

Definition 2.3.7. A type is said to be a *h-groupoid* if the following type is inhabited

$$\mathsf{isGpd}(A) \;:\equiv\; \Pi_{(x,y:A)}\mathsf{isSet}(\mathrm{Id}_A(x, y))$$

Two important general facts about the h-levels are the following: the hierarchy of h-levels is cumulative and the type $\mathsf{hlevel}_n(A)$ for A type is a mere proposition.

Theorem 2.3.8 (Thm 7.1.7 [155])**.** *If the type* $\mathsf{hlevel}_n(A)$ *is inhabited, then the type* $\mathsf{hlevel}_{n+1}(A)$ *is also inhabited.*

Theorem 2.3.9 (Thm 7.1.10 [155])**.** *Let A be a type. The type* $\mathsf{hlevel}_n(A)$ *is a h-proposition.*[2]

As we have seen, an inductive type is specified first by defining clauses telling us how to construct its elements and then by giving the corresponding induction principle. In the light of the homotopical interpretation, the constructors may be called *point* constructors. A useful generalization consists in admitting types whose definition includes *path* constructors other than point constructors. The corresponding induction principle should take in account simultaneously both kinds of constructors. These types are called *higher inductive types*.

[2]The proof of this theorem requires function extensionality.

For instance, many *colimits* constructions can be formalized in **HoTT** as higher inductive types. A simple example is the notion of coequalizer. Given two parallel morphisms f, g from A to B, then the coequalizer of f and g is an object $\mathsf{Coeq}(f, g)$ with a morphism $h : B \to \mathsf{Coeq}(f, g)$ such that $h \circ f = h \circ g$. Moreover, given another object C with a morphisms $k : B \to C$, there is a unique morphism $j : \mathsf{Coeq}(f, g) \to C$ such that $k = j \circ h$. An attempt to formalize this notion naturally leads to an example of HIT. Namely, we start from the formation rule

$$\frac{\Gamma \vdash f : A \to B \qquad \Gamma \vdash g : A \to B}{\Gamma \vdash \mathsf{Coeq}(f, g)\ \mathit{type}}\ \mathsf{Coeq\text{-}F$$

Then we have to define the constructors corresponding to the morphism h above with the accompanying equation:

$$\frac{\Gamma \vdash b : B}{\Gamma \vdash \mathsf{in}(b) : \mathsf{Coeq}(f, g)}\ \mathsf{Coeq\text{-}I_1}$$
$$\frac{\Gamma \vdash a : A}{\Gamma \vdash \mathsf{eq}(a) : \mathsf{in}(f(a)) = \mathsf{in}(g(a))}\ \mathsf{Coeq\text{-}I_2}$$

Finally, we formulate the elimination rule that gives the correspondent induction principle and thus the universal property of a coequalizer:

$$\Gamma, z : \mathsf{Coeq}(f, g) \vdash C(z)\ \mathit{type}$$
$$\Gamma, y : B \vdash d_0 : C(\mathsf{in}(y))$$
$$\Gamma, x : A \vdash d_1 : \mathsf{eq}(x)_*(d_0(f(x))) = d_0(g(x))$$
$$\frac{\Gamma \vdash p : \mathsf{Coeq}(f, g)}{\Gamma \vdash \mathsf{ind}_{\mathsf{Coeq}}(p, d_0, d_1) : C(p)}\ \mathsf{Coeq}(f, g)\text{-E}$$

The computation rules can be quickly inferred from the previous rules.
Another interesting example of HIT is the *interval* type $\mathcal{I}$ that is characterized by the following rules:

$$\frac{\Gamma\ \mathsf{ctxt}}{\Gamma \vdash \mathcal{I}\ \mathit{type}}\ \mathcal{I}\text{-F}$$

$$\frac{\Gamma \ \mathsf{ctxt}}{\Gamma \vdash 0 : \mathcal{I}}\,\mathcal{I}\text{-I}_1 \qquad \frac{\Gamma \ \mathsf{ctxt}}{\Gamma \vdash 1 : \mathcal{I}}\,\mathcal{I}\text{-I}_2 \qquad \frac{\Gamma \ \mathsf{ctxt}}{\Gamma \vdash \mathsf{seg} : 0 =_{\mathcal{I}} 1}\,\mathcal{I}\text{-I}_3$$

$$\frac{\begin{array}{c}\Gamma, x : \mathcal{I} \vdash C(x)\ \mathit{type} \\ \Gamma \vdash c_0 : C(0) \\ \Gamma \vdash c_1 : C(1) \\ \Gamma \vdash c_{\mathsf{seg}} : \mathsf{seg}_\star(c_0) =_{C(1)} c_1 \\ \Gamma \vdash i : \mathcal{I}\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathcal{I}}(i, c_0, c_1, c_{\mathsf{seg}}) : C(i)}\,\mathcal{I}\text{-E}$$

It can be shown that the interval type is contractible and implies the axiom of function extensionality (see Section 6.3 [155]).
The homotopical interpretation of intensional type theory has not only paved the way for a new understanding of type theory and its semantics, but has also brought to light the fact that this system is a natural setting for doing homotopy theory synthetically. Indeed, many results from classical homotopy theory can be developed in a type-theoretical way and thus HoTT can be regarded as a language to reason directly about homotopical structures. In this direction, higher inductive types play a fundamental role. One of the first examples given in [155] is the type-theoretic counterpart of the classical result stating that $\pi_1(\mathcal{S}^1) \cong \mathbb{Z}$, i.e. the fundamental group of the circle is isomorphic to $\mathbb{Z}$. The circle provides another example of HIT:

$$\frac{\Gamma \ \mathsf{ctxt}}{\Gamma \vdash \mathcal{S}^1\ \mathit{type}}\,\mathcal{S}^1\text{-F}$$

$$\frac{\Gamma \ \mathsf{ctxt}}{\Gamma \vdash \mathsf{base} : \mathcal{S}^1}\,\mathcal{S}^1\text{-I}_1 \qquad \frac{\Gamma \ \mathsf{ctxt}}{\Gamma \vdash \mathsf{loop} : \mathsf{base} =_{\mathcal{S}^1} \mathsf{base}}\,\mathcal{S}^1\text{-I}_2$$

$$\frac{\begin{array}{c}\Gamma, z : \mathcal{S}^1 \vdash C(z)\ \mathit{type} \\ \Gamma \vdash c_b : C(\mathsf{base}) \\ \Gamma \vdash c_\ell : c_b =_{C(\mathsf{loop})} c_b \\ \Gamma \vdash d : \mathcal{S}^1\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathcal{S}^1}(d, c_b, c_\ell) : C(d)}\,\mathcal{S}^1\text{-E}$$

As I mentioned above, it is possible to define within **HoTT** a logic of h-propositions which has quite different properties from the propositions-as-

types logic. It is straightforward to show that the types $\mathbf{0}$ and $\mathbf{1}$ are h-propositions and that the type constructors $\times$ and Π preserve h-propositions in the following sense: if A and B are h-propositions, then $A \times B$ is a h-proposition and if $B(x)$ $[x : A]$ is a h-proposition for every $x : A$, where A is a type with *arbitrary* h-level, then $\Pi_{(x:A)}B(x)$ is a h-proposition. In particular, it follows that $\to$ and $\neg$ also preserve h-propositions. However, this remark cannot be generalized to all type constructors. In particular, those constructors which are stronger under the Curry-Howard reading than in the standard first-order presentation do not preserve h-propositions. Namely, they are Σ and $+$[3]. Therefore, we need to find a way to express existential quantification and disjunction at the level of h-propositions. The solution is provided by another example of HIT that is known as *propositional truncation*. Informally, the propositional truncation of a type A corresponds to erase all the information contained in the elements of A and to equate all of them. The only information that is preserved is that A is inhabited and thus provable, but we do not distinguish between the proof-terms of A. The rules for propositional truncation are the following:

$$\frac{\Gamma \vdash A\ type}{\Gamma \vdash ||\ A\ ||\ type}\ ||-||\text{-F}$$

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash |a| : ||\ A\ ||}\ ||-||\text{-I}_1 \qquad \frac{\Gamma \vdash x : ||\ A\ || \qquad \Gamma \vdash y : ||\ A\ ||}{\Gamma \vdash \mathsf{sq}(x,y) : x =_{||\ A\ ||} y}\ ||-||\text{-I}_2$$

$$\frac{\begin{array}{c}\Gamma, z : ||\ A\ || \vdash C(z)\ type \\ \Gamma \vdash c_0 : C(|a|) \\ \Gamma, x : ||\ A\ ||, y : ||\ A\ ||, d : C(x), e : C(y) \vdash c_1 : \mathsf{sq}(x,y)_*(d) =_{C(y)} e \\ \Gamma \vdash t : ||\ A\ ||\end{array}}{\Gamma \vdash \mathsf{ind}_{||\ ||}(t, c_0, c_1) : C(t)}\ ||-||\text{-E}$$

In most cases, it is sufficient to consider just the recursion principle, that is the non-dependent version of the elimination rule

[3]A simple counterexample is the type $\mathbf{1} + \mathbf{1}$ that is not a h-proposition.

$$\frac{\Gamma, x : A \vdash b : B \qquad \Gamma \vdash p : \mathsf{isProp}(B) \qquad \Gamma \vdash t : \| A \|}{\Gamma \vdash \mathsf{ind}_{\| \ \|}(t, x.b, p) : B}$$

The recursion principle says that if there is a map from A to B and B is a h-proposition, then there is a map from the propositional truncation of A to B.
In the light of the previous remarks, we can just take $\| A + B \|$ as the disjunction connective for h-propositions and $\| \Sigma_{(x:A)} B(x) \|$ as the existential quantifier.

Remark 2.3.10. The logic of h-propositions allows to recover classical logic *locally* in the context of HoTT. The law of excluded middle or, equivalently, the double negation law can be consistently added to the logic of h-propositions, contrary to what happens if we postulate one of them for all types under the assumption of Univalence. Further, the truncation operation can be generalized to the other h-levels as well and is an example of *modality* [132]. Finally, a distinctive feature of the logic of h-propositions is that the type-theoretic AC is not derivable once the existential quantifier is not identified with Σ, but the axiom of *unique choice* AC! is. This principle can be phrased as saying that any function (i.e. functional relation) gives rise to an operation (i.e. a type-theoretic function).

The remark that h-propositions are closed under some type constructors can be extended to h-sets. Actually, in the case of h-sets it is possible to prove that they are preserved by $+$ and that if A is a h-set and $B(x)$ $[x : A]$ is a h-set for any $x : A$, then $\Sigma_{(x:A)} B(x)$ is a h-set. The fact that **HoTT** provides a nice setting for developing set-theoretical notion is confirmed by the possibility of defining *effective* set quotients, that is a notoriously delicate matter in type theory. These can be defined as higher inductive types by means of the following rules: the formation rule says that if A is a h-set and $R(x, y)$ $[x : A, y : A]$ is a h-prop for all $x, y : A$ and is also an equivalence relation, then we can form the type A/R with the following point and path constructors

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash [a] : A/R}\ \mathcal{Q}\text{-I}_1 \qquad \frac{\begin{array}{c}\Gamma \vdash a : A \\ \Gamma \vdash b : A \\ \Gamma \vdash r : R(a, b)\end{array}}{\Gamma \vdash \mathsf{eq}(a, b) : [a] =_{A/R} [b]}\ \mathcal{Q}\text{-I}_2$$

$$\frac{\begin{array}{c}\Gamma \vdash x : A/R \\ \Gamma \vdash y : A/R \\ \Gamma \vdash p : x =_{A/R} y \\ \Gamma \vdash q : x =_{A/R} y\end{array}}{\Gamma \vdash \mathsf{hset}(p,q) : p =_{\mathrm{Id}_{A/R}} q}\ \mathcal{Q}\text{-I}_3$$

and with the following elimination rule

$$\frac{\begin{array}{c}\Gamma, z : A/R \vdash C(z)\ \mathit{type} \\ \Gamma, z : A/R \vdash f : \mathsf{isSet}(C(z)) \\ \Gamma, x : A \vdash c_0 : C([x]) \\ \Gamma, x, y : A, r : R(x,y), u : C([x]), v : C([y]) \vdash c_1 : \mathsf{eq}(x,y)_*(u) =_{C([y])} v \\ \Gamma \vdash t : A/R\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathcal{Q}}(t, c_0, c_1) : C(t)}\ \mathcal{Q}\text{-E}$$

In general, we get an equivalence between $R(a,b)$ and $[a] =_{A/R} [b]$, meaning that set quotients are effective. We can also define more general homotopy quotients by taking R to be an arbitrary prop-valued relation and without requiring that A be a h-set.

Remark 2.3.11. The first type-theoretic definitions of quotient types were given long before the concept of HIT was introduced. For instance, Hofmann gave a definition of quotients in [66] in a type theory extended with a proof-irrelevant type of propositions. Further, Maietti [87] proved that if we assume effectiveness for arbitrary equivalence relations, then the law of excluded middle follows.

2.4 Homotopies & Equivalences

In general, within an intensional setting, we cannot conclude that two maps are equal if they are *pointwise* equal. We could do so under the assumption of the function extensionality axiom. However, in many cases we can establish that two maps f, g are pointwise equal and thus for each point x, we have an identification between the values of f and g at x. Since type-theoretic functions can be regarded as functors, it is natural to consider these identifications as natural transformations. In **HoTT**, they are called *homotopies* and the type of the homotopies between two functions is defined as follows:

Definition 2.4.1. Let $f, g : \Pi_{(x:A)} B(x)$. Then the type of homotopies from f to g is

$$(f \sim g) \;:\equiv\; \Pi_{(x:A)}(f(x) =_{B(x)} g(x)) \;.$$

In general, there could be multiple terms inhabiting the type $(f \sim g)$, that in turn give rise to higher homotopies.
The idea that these homotopies behave as natural transformations is substantiated by the following lemma:

Lemma 2.4.2. *Let f, g be two terms in $A \to B$ and let $p : x =_A y$. If $h : f \sim g$, then the following type is inhabited*

$$h(x) \blacksquare \mathsf{ap}_g(p) = \mathsf{ap}_f(p) \blacksquare h(y)$$

Proof. It is sufficient to apply the Id-E rule and thus to assume that $p \equiv \mathsf{refl}(x)$. Since

$$C(x, x, \mathsf{refl}(x)) \;:\equiv\; h(x) \blacksquare \mathsf{ap}_g(\mathsf{refl}(x)) = \mathsf{ap}_f(\mathsf{refl}(x)) \blacksquare h(x)$$

That implies $h(x) \blacksquare \mathsf{refl}(g(x)) = \mathsf{refl}(f(x)) \blacksquare h(x)$ and thus $h(x) = h(x)$ that is trivially inhabited. Therefore, by Id-E, we can conclude that $C(x, y, p)$ is inhabited. □

The lemma can be generalized to the dependent case where $f, g : \Pi_{(x:A)} B(x)$ and thus we have to consider apd_f and apd_g whose types contain an application of transport. Then the result follows by applying Id-E.
The notion of homotopy can be applied in order to define an appropriate notion of *sameness* between types. The naive attempt amounts to define in the language of type theory the familiar notion of isomorphism as a map with a two-sided inverse. Given a map $f : A \to B$, we say that f is an isomorphism is the following type is inhabited

$$\Sigma_{g:B\to A} \left((f \circ g \sim \mathsf{id}_B) \times (g \circ f \sim \mathsf{id}_A)\right)$$

However, there are good reasons to discard this definition as the correct one, since, in particular, it does not work properly at higher dimensions. We call this type $\mathsf{qinv}(f)$, the type of *quasi-inverses* of f, following [155]. In general, $\mathsf{qinv}(f)$ is not always a proposition and has the undesired consequence that the Univalence Axiom turns out to be false if formulated according this notion of isomorphism. Then, this notion is replaced by that of *equivalence*, that does not suffer from the same problems.

Definition 2.4.3. Let $f : A \to B$. The map f is an *equivalence* if the following type is inhabited

$$\mathsf{iseqv}(f) \;:\equiv\; (\Sigma_{(g:B\to A)}\; f \circ g \sim \mathsf{id}_B) \times (\Sigma_{(h:B\to A)}\; h \circ f \sim \mathsf{id}_A)$$

The type of equivalences from a type A to a type B is defined as follows:

$$(A \simeq B) \;:\equiv\; \Sigma_{(f:A\to B)}\; \mathsf{iseqv}(f)\ .$$

There are back and forth maps between $\mathsf{qinv}(f)$ and $\mathsf{iseqv}(f)$, but the latter satisfies the additional property of being a h-proposition (see Ch.4 [155])[4]. Therefore an equivalence f consists of the following data: a section of f given by g, a retraction of f given by h and two homotopies H, K. If function extensionality holds, then we can define the notion of equivalence directly in terms of equality, since homotopies and equalities are not distinct anymore. Finally, a basic closure property of equivalences, that is known as '2-out-of-3-property' in the literature, is the following:

Proposition 2.4.4 (2-out-of-3)**.** *Let $f : A \to B$, $g : B \to C$ be equivalences. Then if any two of f, g and $g \circ f$ are equivalences, then the so is the third.*

Proof. If f and g are equivalences, then $g \circ f$ is an equivalence. The quasi-inverses of f and g are f^{-1} and g^{-1}. Therefore, given $x : A$

$$f^{-1}(g^{-1}(g(f(x)))) = f^{-1}(f(x)) = x$$

and, on the other hand, given $z : C$

$$g(f(f^{-1}(g^{-1}(z)))) = g(g^{-1}(z)) = z$$

Hence $f^{-1} \circ g^{-1} = (g \circ f)^{-1}$ is a quasi-inverse to $g \circ f$.
If $g \circ f$ and g are equivalences, then f is an equivalence. The quasi-inverses of $g \circ f$ and g are $(g \circ f)^{-1}$ and g^{-1}. Then, given $x : A$,

$$f^{-1}(g^{-1}(g(f(x)))) = x$$

and given $y : B$,

$$f(f^{-1}(g^{-1}(g(y)))) = g^{-1}(g(f(f^{-1}(g^{-1}(g(y)))))) = g^{-1}(g(y)) = y$$

[4]Note that this proof uses function extensionality.

Therefore, $(g \circ f)^{-1} \circ g$ is a quasi-inverse to f.
If $g \circ f$ and f are equivalences, then g is an equivalence. The quasi-inverses of $g \circ f$ and f are $(g \circ f)^{-1}$ and f^{-1}. Then, given $y : B$,

$$f(f^{-1}(g^{-1}(g(y)))) = f(f^{-1}(g^{-1}(g(f(f^{-1}(y)))))) = f(f^{-1}(y)) = y$$

and, conversely, given $z : C$

$$g(f(f^{-1}(g^{-1}(z)))) = z$$

Therefore, $f \circ (g \circ f)^{-1}$ is a quasi-inverse to g. □

2.5 The Univalence Axiom

The distinctive feature of **HoTT** is that it includes an axiom, Univalence, that provides a specification of the identity type of a universe $\mathcal{U}$. Thus the axiom determines a notion that otherwise would be left unspecified. In doing so, a notion of sameness for types is obtained: roughly speaking, equivalent types are equal.
Let us observe that, for any two types $A, B : \mathcal{U}$, a map

$$\mathsf{ideqv}_{A,B} : (A =_{\mathcal{U}} B) \to (A \simeq B)$$

can always be constructed (by induction over equality). Then the Univalence Axiom amounts to the following:

(UA). For any two types $A, B : \mathcal{U}$, the map $\mathsf{ideqv}_{A,B}$ is an equivalence.

This implies that $(A =_{\mathcal{U}} B) \simeq (A \simeq B)$. The quasi-inverse to ideqv is $\mathsf{ua} : (A \simeq B) \to (A =_{\mathcal{U}} B)$. Moreover, it must be noted the axiom refers to a specific universe $\mathcal{U}$.

Definition 2.5.1. A universe $\mathcal{U}$ is said to be *univalent* if it satisfies UA.

Remark 2.5.2. The Univalence Axiom has been introduced by Vladimir Voevodsky, who has also established its consistency by the construction of a model in simplicial sets [73]. However, the axiom was foreshadowed by Hofmann and Streicher in [67] under the name of *Universe Extensionality*.

Univalence has many non-trivial consequences. First, UA can be regarded as a general extensionality principle stating that there is no more-fine grained way of distinguishing types than equivalence. Therefore, two types are regarded as equal if they exhibit the same behaviour in every context. This reading is substantiated by the following facts that can all be deduced from UA:

Proposition 2.5.3 (Propositional Extensionality)**.** *If $A, B : \mathcal{U}$ are h-propositions and $\mathcal{U}$ is univalent, then the following type is inhabited*

$$(A = B) \simeq (A \leftrightarrow B)$$

Proposition 2.5.4 (Function Extensionality)**.** *Let $A : \mathcal{U}, B : A \to \mathcal{U}$ and $f, g : \Pi_{(x:A)} B(x)$. If $\mathcal{U}$ is univalent, then the following type is inhabited*

$$\Pi_{(x:A)}(f(x) = g(x)) \to (f = g)$$

Proposition 2.5.5 (Transport along equivalences)**.** *Let $A, B : \mathcal{U}$ and $P : \mathcal{U} \to \mathcal{U}$. If $\mathcal{U}$ is univalent and $A \simeq B$, then the following type is inhabited*

$$P(A) \simeq P(B)$$

Further, classical principles such as LEM or double negation elimination must be regarded as strictly logical principles in the sense that they can be assumed only for h-propositions, since they are inconsistent if assumed for every type in a univalent universe $\mathcal{U}$ (see Theorem 3.2.2 in [155]).
Postulating the uniqueness of identity proofs also for the universe is inconsistent with UA. Indeed, Univalence allows to derive that the hierarchy of h-levels is not degenerate, in the sense that not all types are h-sets (see Example 3.1.9 in [155]).
Finally, under the assumption of UA, the following fundamental fact can be derived:

Theorem 2.5.6 (Grothendieck correspondence for types)**.** *Let $A : \mathcal{U}$ and $\mathcal{U}$ univalent, then the following type is inhabited*

$$(\Sigma_{(X:\mathcal{U})}\ X \to A) \simeq (A \to \mathcal{U})$$

This theorem establishes a precise correspondence between family of types and fibrations. On the left side, we have maps into A, whereas on the other side we have families indexed over A. A proof can be found in Section 4.8 of [155].

However, UA has also the consequence of breaking canonicity and hence the good computational properties of type theory. In fact, we may derive a closed term of type A containing ua by an application of the axiom, but we cannot conclude that such a term reduces to a canonical term of type A, since the computational behaviour of the new constant ua is unknown. This problem has motivated the investigations into computational interpretations of Univalence.

Chapter 3

Invariance under Equivalence

The aim of the present chapter is to clarify the meaning of the Univalence Axiom and analyse its connection with the *principle of invariance* which states that any definable property should be invariant under equivalence. Such a principle is often invoked as a hallmark of structuralist conceptions of mathematics. As we have seen in the previous chapter, Univalence trivially implies the principle of invariance, since it enables transport along equivalences. Transport along equivalences can be regarded as a generalization of plain transport, namely the indiscernibility of identicals principle. In particular, the notion of identity is *expanded* to the notion of equivalence. In many areas of mathematics the relevant notion of *sameness* is not captured by the identity relation, which is too limited and this leads to consider notions such as isomorphism or equivalence. In this sense, the principle of invariance appears to be dictated by the mathematical practice itself, at least for what concerns many areas of modern mathematics. A similar motivation for introducing Univalence has been given by Awodey in [10, 11]: there, Univalence is justified as the internalization of the principle of invariance. The aim of the chapter is to further work out Awodey's argument, by showing that the principle of invariance is satisfied by Martin-Löf Type Theory without universes, which is what is actually assumed by Awodey. Further, this fact is established by a syntactic model construction which closely resembles parametricity interpretations [131] and can be viewed as a concrete instantiation of an inverse diagram model in the sense of [71, 72].

3.1 The Principle of Invariance

In [93], M.Makkai formulated the following *Principle of Isomorphism*:

> 'All grammatically correct properties of objects of a fixed category are to be invariant under isomorphism'

Then, if P is a definable property , $P(A)$ holds and $A \cong B$, then $P(B)$ holds as well.
Standard set-theoretic foundations obviously violate this principle: the membership relation allows for the formulation of statements which are not invariant under bijection of sets for instance. Further, the same holds also for the language of categories: statements which make reference to the equality of objects provide examples of properties not invariant under equivalence of categories.
It should be noted that the principle of isomorphism is quite strong. It is not a structural principle prescribing that the relevant properties in a given mathematical domain are the structural ones, which can be adopted as a heuristic principle also within non-invariant frameworks. The principle is meant to put a constraint on the meaningful expressions of a given language; then, it implies that certain notions and expressions must be ruled out as meaningless. Therefore, it provides a *syntactic* criterion about the definability of expressions.
Nevertheless, it would be misleading to conceive the principle simply as a way of restricting the class of meaningful expressions of a given language. In fact, it is meant to give a full foundational status to an idea which is often invoked in the practice of structural mathematics: two isomorphic objects are indistinguishable from a structural point of view and have the same structural properties. In particular, if a property P is established about an object G and we know that G is the 'same ' as H, where the relevant notion of sameness is dictated by the kinds of objects we are considering, then we can conclude that P holds of H as well. Then, this allows for the possibility of transporting theorems along the relevant notion of sameness. In this way, the same statement can be seen to be valid for a rich variety of objects without taking into account their concrete realizations.
In many cases, the plain notion of identity does not correspond to the notion of sameness which is backed up by practice. For example, a group theorist will consider two groups $\mathcal{G}$ and $\mathcal{H}$ to be the same whenever they are isomorphic.

She will be justified in doing so because all the relevant group-theoretic properties can be transported across isomorphisms: e.g., if $\mathcal{G}$ is abelian, then $\mathcal{H}$ is abelian as well; if $\mathcal{G}$ is cyclic, then $\mathcal{H}$ is cyclic as well and so on. However, in the standard set-theoretic presentation, the conclusion that $\mathcal{G}$ and $\mathcal{H}$ are equal is false: it is sufficient to consider the groups $\mathcal{G} :\equiv \mathbb{Z}$ and $\mathcal{H} :\equiv 2\mathbb{Z}$, then $1 \in \mathbb{Z}$, but $1 \notin 2\mathbb{Z}$. The property used to exhibit a counterexample relies on the set-theoretic membership and is not really meaningful from an algebraic perspective: it appears to be the byproduct of the language chosen to represent the algebraic structures. It can be regarded as a spurious result of the formalization process without any intrinsic mathematical meaning. It is these kinds of considerations which lead to endorse the claim that the relevant notion of sameness is provided by group-isomorphism. The principle of isomorphism as advocated by Makkai goes one step further by asking that such informal principle be syntactically implemented in such a way to get a language where 'spurious' properties are held to be meaningless and all the definable constructs are isomorphism-invariant.

The important lesson from the previous chapter is that the notion of isomorphism is adequate for set-level structures, but it cannot be assumed as providing a general notion of sameness for all structures. Indeed, there are examples of higher structures where the notion of equivalence gives the adequate sameness criterion. Then, as we have already remarked, Univalence implies that all definable properties of **HoTT** are invariant under equivalences and hence not only satisfies the Makkai's principle, but generalizes it to the notion of homotopy equivalence.

In [10], Univalence is discussed in relation to the informal principle of structuralism according to which 'isomorphic objects are equal'. As we have seen, Univalence actually implies this principle for a large class of mathematical structures. Further, Awodey observes that for the language of type theory the meta-theoretical principle of invariance is satisfied and hence any definable type-theoretical property turns out to be equivalence-invariant. For the language of simple type theory, this was already observed by Lindenbaum and Tarski in [5][1]. Awodey's argument assumes that the same result can be extended to dependent type theory, but the crucial point is that such invariance property is broken when universes enter into the picture. The introduction of a universe $\mathcal{U}$ allows, among other things, to take into account

[1]Apparently the same result was obtained in some unpublished work by Carnap. I thank Steve Awodey for having pointed out to me this fact.

identities between types in $\mathcal{U}$. Let $P(X)$ be a property involving an identity between types: $P(X) :\equiv \mathrm{Id}_{\mathcal{U}}(A, X)$. Then, $P(A)$ is just $\mathrm{Id}_{\mathcal{U}}(A, A)$, which is an axiom since it is inhabited by reflexivity. Therefore, the validity of the principle of invariance would imply that equivalent types are equal, because it boils down to $A \simeq B \to \mathrm{Id}_{\mathcal{U}}(A, B)$. Hence, invariance implies the general form of the principle of structuralism.
It is enough to consider non-univalent models of type theory to show that such a statement is not derivable in general, while it is immediately seen to be valid if the universe $\mathcal{U}$ is assumed to be univalent. Then, Awodey concludes, Univalence can be regarded as an internalization of the principle of invariance and is fundamental to restore this property in presence of a universe.

Remark 3.1.1. In the last section of [10], Awodey makes an interesting historical parallel with the introduction of the axiom of extensionality for propositional functions by Russell in the second edition of *Principia*(1925). Indeed, the original system did not contain any extensionality principle and Russell was led to consider the introduction of an extensionality axiom on the basis of a remark by Wittgenstein according to which 'a function can only occur in a proposition through its values'. This remark suggested the adoption of an extensional point of view about propositional functions since we can observe only their external behaviour within the system and two propositional functions having the same behavior cannot be practically distinguished by means of any definable property. Therefore, the axiom of extensionality allows to declare them equal, a conclusion which cannot be reached within the intensional system. Univalence can be regarded as a general extensionality principle and identifies exactly those types which are not distinguishable in plain intensional type theory. Therefore, Awodey's argument can be understood as a vast generalization of Russell-Wittgenstein's argument for extensionality in the system of *Principia*.

The argument makes the assumption that the principle of invariance is already satisfied by Martin-Löf Type Theory without universes: the insight being that such a principle can be established by inductive reasoning over the construction of P, because it is a statement about all the properties which are actually definable in the system. However, the naive argument runs into obvious difficulties. In the construction of P type dependencies might occur and, in general, terms and types cannot be defined independently within a dependently typed system. Therefore, we cannot establish the claim by direct induction over the construction of P. A more elaborated model construction

is required to show the correctness of Awodey's claim and it will be worked out in the next sections.

Remark 3.1.2. One might think that the consistency of Univalence, first established by Voevodsky with the construction of the simplicial set model [73], already implies the principle of invariance. Actually what follows from the consistency of the Univalence Axiom is just that is not possible to define a non-equivalence invariant property in the language of type theory. The statement of the principle of invariance is the positive statement for which all definable type-theoretic properties are provably equivalence-invariant.

3.2 The Proof-Relevant Relations Model

The main insight behind the interpretation I am going to define is to interpret a type by a triple consisting of a pair of types and a relation over them: e.g. the type Bool is interpreted as follows

$$[\![\mathsf{Bool}]\!] \;:\equiv\; x : \mathsf{Bool}, y : \mathsf{Bool} \vdash \mathrm{R}_{\mathsf{bool}}(x, y)\ \mathit{type}$$

In order to define this interpretation, I will assume as meta-theory an extension of **MLTT** with *relational* type constructors. In particular, for a relational type constructor such as $\mathrm{R}_{\mathsf{bool}}$ we will have two canonical term constructors which are

$$\rho_{\mathsf{ff}} : \mathrm{R}_{\mathsf{bool}}(\mathsf{ff}, \mathsf{ff})$$
$$\rho_{\mathsf{tt}} : \mathrm{R}_{\mathsf{bool}}(\mathsf{tt}, \mathsf{tt})$$

Clearly, the relational type constructors will be equipped with the corresponding eliminators. Considering the example of $\mathrm{R}_{\mathsf{bool}}$ again, it should be noted that the definition of the relational type allows us to see immediately that the relation is defined only when the arguments are *(true,true)* or *(false, false)*. Therefore, the cases $\mathrm{R}_{\mathsf{bool}}(\mathsf{tt}, \mathsf{ff})$ and $\mathrm{R}_{\mathsf{bool}}(\mathsf{ff}, \mathsf{tt})$ are ruled out. It is sufficient to analyze a value of type $\mathrm{R}_{\mathsf{bool}}$ to reach this conclusion.
Intuitively, two functions f, g from A to B are related when they take related inputs to related outputs. Therefore, given $f_0 : A_0 \to B_0$ and $f_1 : A_1 \to B_1$,

$$\mathrm{R}_{A\to B}(f_0, f_1) :\equiv \Pi_{(x_0:A_0)}\Pi_{(x_1:A_1)}\Pi_{(x_r:A_r(x_0,x_1))}\ B_r(x_r, f_0(x_0), f_1(x_1))$$

The same pattern must be extended to contexts: a context is not interpreted as a telescope of variables, but for each variable x_i in Γ we get a triple (x_{i0}, x_{i1}, x_{ir}), where the last component is a proof that the first two variables are related.
The previous explanation shows what is the general pattern followed by the interpretation. However, this construction must be further refined in order to get a proof of the principle of invariance: the idea is to require that the relations interpreting the type constructors be *equivalences* by employing a suitable definition of equivalence.
The pattern described is reminiscent of the *logical relations* method to obtain *parametricity* meta-theorems. The notion of parametricity was introduced by Reynolds in the seminal paper [131] in order to characterize the *uniform* behaviour of polymorphic programs in System **F**. Reynolds designed a relational model to establish this result and then he showed that these programs are *invariant* w.r.t. different implementations: e.g. any closed term of type $\forall A.A \to A$ in **F** is equivalent to the identity map. Among the well-known applications of parametricity, there are the so-called 'free theorems' [160].
Parametricity captures a *representation independence* principle: if two implementations of an abstract type structure are related in a suitably defined sense, then they are observationally equivalent. In this respect, representation independence is quite close to equivalence-invariance and in fact it has been exploited to get automatic transport of theorems across related structures. Therefore, it is not surprising that a relational model construction is what we need to show that **MLTT** satisfies the principle of invariance.
Further, it should be noted that by 'relations' in the present context I mean *type-valued* relations or *correspondences*, which are more general than the standard prop-valued relations. Then, the model is a *proof-relevant* relations model.
Such approach has been inspired by the abstract *inverse diagram models* considered in [71, 72]. In particular, there two constructions are introduced of which the present ones can be regarded as the syntactical counterparts. I will review the relation with these models at the end of the chapter.
The relational type constructors which are needed for interpreting **MLTT**-syntax are given by the following definitions.

Definition 3.2.1. The type $\mathrm{R}_{\mathbf{1}}$ is defined as follows:

Formation: $\Gamma, x : \mathbf{1}, y : \mathbf{1} \vdash \mathrm{R}_{\mathbf{1}}(x, y)$ *type*

Introduction: $\Gamma \vdash \rho_* : \mathrm{R}_{\mathbf{1}}(*, *)$ provided $\Gamma \vdash * : \mathbf{1}$

Elimination: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathbf{1}}}(a_r, c) : C(a_0, a_1, a_r)$ provided
$\Gamma, x_0, x_1 : \mathbf{1}, x_r : \mathrm{R}_{\mathbf{1}}(x_0, x_1) \vdash C(x_0, x_1, x_r)$ *type* and
$\Gamma \vdash a_r : \mathrm{R}_{\mathbf{1}}(a_0, a_1)$ and
$\Gamma \vdash c : C(*, *, \rho_*)$

Computation: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathbf{1}}}(\rho_*, c) \equiv c : C(*, *, \rho_*)$

Definition 3.2.2. The type $\mathrm{R}_{\mathbf{0}}$ is defined as follows:

Formation: $\Gamma, x : \mathbf{0}, y : \mathbf{0} \vdash \mathrm{R}_{\mathbf{0}}(x, y)$ *type*

Elimination: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathbf{0}}}(a_r) : C(a_0, a_1, a_r)$ provided
$\Gamma, x_0, x_1 : \mathbf{0}, x_r : \mathrm{R}_{\mathbf{0}}(x_0, x_1) \vdash C(x_0, x_1, x_r)$ *type* and
$\Gamma \vdash a_r : \mathrm{R}_{\mathbf{0}}(a_0, a_1)$

Definition 3.2.3. The type $\mathrm{R}_{\mathsf{bool}}$ is defined as follows:

Formation: $\Gamma, x : \mathsf{Bool}, y : \mathsf{Bool} \vdash \mathrm{R}_{\mathsf{bool}}(x, y)$ *type*

Introduction: $\Gamma \vdash \rho_{\mathsf{tt}} : \mathrm{R}_{\mathsf{bool}}(\mathsf{tt}, \mathsf{tt})$ provided $\Gamma \vdash \mathsf{tt} : \mathsf{Bool}$ and
$\Gamma \vdash \rho_{\mathsf{ff}} : \mathrm{R}_{\mathsf{bool}}(\mathsf{ff}, \mathsf{ff})$ provided $\Gamma \vdash \mathsf{ff} : \mathsf{Bool}$

Elimination: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathsf{bool}}}(c_r, t_0, t_1) : C(c_0, c_1, c_r)$ provided
$\Gamma, x_0 : \mathsf{Bool}, x_1 : \mathsf{Bool}, x_r : \mathrm{R}_{\mathsf{bool}}(x_0, x_1) \vdash C(x_0, x_1, x_r)$ *type* and
$\Gamma \vdash t_0 : C(\mathsf{ff}, \mathsf{ff}, \rho_{\mathsf{ff}})$ and
$\Gamma \vdash t_1 : C(\mathsf{tt}, \mathsf{tt}, \rho_{\mathsf{tt}})$ and
$\Gamma \vdash c_r : \mathrm{R}_{\mathsf{bool}}(c_0, c_1)$

Computation: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathsf{bool}}}(\rho_{\mathsf{ff}}, t_0, t_1) \equiv t_0 : C(\mathsf{ff}, \mathsf{ff}, \rho_{\mathsf{ff}})$ and
$\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathsf{bool}}}(\rho_{\mathsf{tt}}, t_0, t_1) \equiv t_1 : C(\mathsf{tt}, \mathsf{tt}, \rho_{\mathsf{tt}})$

Definition 3.2.4. The type $\mathrm{R}_{\mathbb{N}}$ is defined as follows:

Formation: $\Gamma, x : \mathbb{N}, y : \mathbb{N} \vdash \mathrm{R}_{\mathbb{N}}(x, y)$ *type*

Introduction: $\Gamma \vdash \rho_0 : \mathrm{R}_{\mathbb{N}}(0,0)$ provided $\Gamma \vdash 0 : \mathbb{N}$ and

$\Gamma \vdash \rho_{\mathsf{s}(a)} : \mathrm{R}_{\mathbb{N}}(\mathsf{s}(a_0), \mathsf{s}(a_1))$ provided $\Gamma \vdash a_0 : \mathbb{N}$ and $\Gamma \vdash a_1 : \mathbb{N}$ and $\Gamma \vdash a_r : \mathrm{R}_{\mathbb{N}}(a_0, a_1)$

Elimination: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathbb{N}}}(m_r, c_0, c_s) : C(m_0, m_1, m_r)$ provided

$\Gamma, x_0, x_1 : \mathbb{N}, x_r : \mathrm{R}_{\mathbb{N}}(x_0, x_1) \vdash C(x_0, x_1, x_r)$ *type* and

$\Gamma \vdash c_0 : C(0, 0, \rho_0)$ and

$\Gamma, z_0, z_1 : \mathbb{N}, z_r : \mathrm{R}_{\mathbb{N}}(z_0, z_1), y : C(z_0, z_1, z_r) \vdash c_s : C(\mathsf{s}(z_0), \mathsf{s}(z_1), \rho_{s(z)})$ and

$\Gamma \vdash m_r : \mathrm{R}_{\mathbb{N}}(m_0, m_1)$

Computation: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathbb{N}}}(\rho_0, c_0, c_s) \equiv c_0 : C(0, 0, \rho_0)$ and

$\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathbb{N}}}(\rho_{s(m)}, c_0, c_s) \equiv c_s\ [\cdots] : C(\mathsf{s}(m_0), \mathsf{s}(m_1), \rho_{s(m)})$

where $c_s[\cdots]$ stands for

$$c_s\ [m_0/z_0, m_1/z_1, m_r/z_r, y/\mathsf{ind}_{\mathrm{R}_{\mathbb{N}}}(m_r, c_0, c_s)]$$

Definition 3.2.5. The type R_{Π} is defined as follows:

Formation:

$\Gamma, z_0 : \Pi_{(x_0:A_0)} B_0(x_0), z_1 : \Pi_{(x_1:A_1)} B_1(x_1) \vdash \mathrm{R}_{\Pi(A,B)}(z_0, z_1)$ *type*

provided

$\Gamma \vdash A_0$ *type* and $\Gamma \vdash A_1$ *type* and

$\Gamma, x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1)$ and

$\Gamma, x_0 : A_0 \vdash B_0(x_0)$ *type* and

$\Gamma, x_1 : A_1 \vdash B_1(x_1)$ *type* and

$\Gamma, x_r : A_r(x_0, x_1), y_0 : B_0(x_0), y_1 : B_1(x_1) \vdash B_r(x_r, y_0, y_1)$

Introduction:

$\Gamma \vdash \rho_{\lambda(x.b)} : \mathrm{R}_{\Pi(A,B)}(\lambda x_0.b_0, \lambda x_1.b_1)$ provided

$\Gamma, x_0 : A_0 \vdash b_0 : B_0(x_0)$ and

$\Gamma, x_1 : A_1 \vdash b_1 : B_1(x_1)$ and

$\Gamma, x_r : A_r(x_0, x_1) \vdash b_r : B_r(x_r, b_0, b_1)$

Elimination:

$\Gamma \vdash \rho_{f(a)} : B_r(a_r, f_0(a_0), f_1(a_1))$

provided

$\Gamma \vdash a_0 : A_0$ and $\Gamma \vdash a_1 : A_1$ and

$\Gamma \vdash a_r : A_r(a_0, a_1)$ and

$\Gamma \vdash f_0 : \Pi_{(x_0:A_0)} B_0(x_0)$ and

$\Gamma \vdash f_1 : \Pi_{(x_1:A_1)} B_1(x_1)$ and

$\Gamma \vdash f_r : \mathrm{R}_{\Pi(A,B)}(f_0, f_1)$

Computation: $\Gamma \vdash \rho_{\lambda(b(a))} \equiv b_r : B_r(a_r, b_0(a_0), b_1(a_1))$

Definition 3.2.6. The type R_Σ is defined as follows:

Formation:

$\Gamma, z_0 : \Sigma_{(x_0:A_0)} B_0(x_0), z_1 : \Sigma_{(x_1:A_1)} B_1(x_1) \vdash \mathrm{R}_{\Sigma(A,B)}(z_0, z_1)$ *type*

provided

$\Gamma \vdash A_0$ *type* and $\Gamma \vdash A_1$ *type* and

$\Gamma, x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1)$ and

$\Gamma, x_0 : A_0 \vdash B_0(x_0)$ *type* and

$\Gamma, x_1 : A_1 \vdash B_1(x_1)$ *type* and

$\Gamma, x_r : A_r(x_0, x_1), y_0 : B_0(x_0), y_1 : B_1(x_1) \vdash B_r(x_r, y_0, y_1)$

Introduction: $\Gamma \vdash \rho_{(a,b)} : \mathrm{R}_{\Sigma(A,B)}((a_0, b_0), (a_1, b_1))$ provided

$\Gamma \vdash a_0 : A_0$ and $\Gamma \vdash a_1 : A_1$ and

$\Gamma \vdash a_r : A_r(a_0, a_1)$ and

$\Gamma \vdash b_0 : B_0(a_0)$ and $\Gamma \vdash b_1 : B_1(a_1)$ and

$\Gamma \vdash b_r : B_r(a_r, b_0, b_1)$

Elimination: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\Sigma(A,B)}}(d_r, c) : C(d_0, d_1, d_r)$ provided

$\Gamma, \Delta \vdash C(z_0, z_1, z_r)$ *type* where

$$\Delta :\equiv z_0 : \Sigma_{(x_0:A_0)} B_0(x_0), z_1 : \Sigma_{(x_1:A_1)} B_1(x_1), z_r : \mathrm{R}_{\Sigma(A,B)}(z_0, z_1)$$

and

$\Gamma, \Theta \vdash c : C((x_0, y_0), (x_1, y_1), \rho_{(x,y)})$ where

$\Theta :\equiv x_0 : A_0, x_1 : A_1, x_r : A_r(x_0, x_1), y_0 : B_0(x_0), y_1 : B_1(x_1), y_r : B_r(x_r, y_0, y_1)$

and

$\Gamma \vdash d_r : \mathrm{R}_{\Sigma(A,B)}(d_0, d_1)$

Computation:

$\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\Sigma(A,B)}}(\rho_{(a,b)}, c) \equiv c : C((a_0, b_0), (a_1, b_1), \rho_{(a,b)})$

Definition 3.2.7. The type R_+ is defined as follows:

Formation: $\Gamma, z_0 : A_0 + B_0, z_1 : A_1 + B_1 \vdash \mathrm{R}_{A+B}(z_0, z_1)$ *type* provided

$\Gamma \vdash A_0$ *type* and $\Gamma \vdash A_1$ *type* and

$\Gamma, x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1)$ and

$\Gamma \vdash B_0$ *type* and $\Gamma \vdash B_1$ *type* and

$\Gamma, x_0 : B_0, x_1 : B_1 \vdash B_r(x_0, x_1)$

Introduction:

$\Gamma \vdash \rho_{\mathsf{inl(a)}} : \mathrm{R}_{A+B}(\mathsf{inl}(a_0), \mathsf{inl}(a_1))$

provided

$\Gamma \vdash a_0 : A_0$ and $\Gamma \vdash a_1 : A_1$ and

$\Gamma \vdash a_r : A_r(a_0, a_1)$

and for the other

$\Gamma \vdash \rho_{\mathsf{inr(b)}} : \mathrm{R}_{A+B}(\mathsf{inr}(b_0), \mathsf{inr}(b_1))$

provided

$\Gamma \vdash b_0 : B_0$ and $\Gamma \vdash b_1 : B_1$ and

$\Gamma \vdash b_r : B_r(b_0, b_1)$

Elimination: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{A+B}}(e_r, c_A, c_B) : C(e_0, e_1, e_r)$

provided

$\Gamma, z_0 : A_0 + B_0, z_1 : A_1 + B_1, z_r : \mathrm{R}_{A+B}(z_0, z_1) \vdash C(z_0, z_1, z_r)$ *type*

and

$\Gamma, x_0 : A_0, x_1 : A_1, x_r : A_r(x_0, x_1) \vdash c_A : C(\mathsf{inl}(x_0), \mathsf{inl}(x_1), \rho_{\mathsf{inl(x)}})$ and

$\Gamma, y_0 : B_0, y_1 : B_1, y_r : B_r(y_0, y_1) \vdash c_B : C(\mathsf{inr}(y_0), \mathsf{inr}(y_1), \rho_{\mathsf{inr(y)}})$ and

$\Gamma \vdash e_r : \mathrm{R}_{A+B}(e_0, e_1)$

Computation:

$\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{A+B}}(\rho_{\mathsf{inl(a)}}, c_A, c_B) \equiv c_A : C(\mathsf{inl}(a_0), \mathsf{inl}(a_1), \rho_{\mathsf{inl(a)}})$ and

$\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{A+B}}(\rho_{\mathsf{inr(b)}}, c_A, c_B) \equiv c_B : C(\mathsf{inr}(b_0), \mathsf{inr}(b_1), \rho_{\mathsf{inr(b)}})$

Definition 3.2.8. The type R_{Id} is defined as follows:

Formation:

$\Gamma, z_0 : \mathrm{Id}_{A_0}(a_0, b_0), z_1 : \mathrm{Id}_{A_1}(a_1, b_1) \vdash \mathrm{R}_{\mathrm{Id_A}}(z_0, z_1)$ *type*

provided

$\Gamma, x_0 : A_0, x_1 : A_1 \vdash x_r : A_r(x_0, x_1)$

$\Gamma \vdash a_0 : A_0$ and $\Gamma \vdash a_1 : A_1$ and

$\Gamma \vdash a_r : A_r(a_0, a_1)$ and

$\Gamma \vdash b_0 : A_0$ and $\Gamma \vdash b_1 : A_1$ and

$\Gamma \vdash b_r : A_r(b_0, b_1)$

Introduction: $\Gamma \vdash \rho_{\mathsf{refl}(a)} : \mathrm{R}_{\mathrm{Id_A}}(\mathsf{refl}(a_0), \mathsf{refl}(a_1))$

provided

$\Gamma \vdash a_0 : A_0$ and $\Gamma \vdash a_1 : A_1$ and

$\Gamma \vdash a_r : A_r(a_0, a_1)$

Elimination: $\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathrm{Id_A}}}(a_r, b_r, p_r, c) : C(p_0, p_1, p_r)$

provided

$\Gamma, \Delta \vdash C(z_0, z_1, z_r)$ *type* where

$$\Delta \ :\equiv \ z_0 : \mathrm{Id}_{A_0}(x_0, y_0), z_1 : \mathrm{Id}_{A_1}(x_1, y_1), z_r : \mathrm{R}_{\mathrm{Id_A}}(z_0, z_1)$$

and

$\Gamma, x_0 : A_0, x_1 : A_1, x_r : A_r(x_0, x_1) \vdash c : C(\mathsf{refl}(x_0), \mathsf{refl}(x_1), \rho_{\mathsf{refl(x)}})$ and

$\Gamma \vdash a_0 : A_0$ and $\Gamma \vdash a_1 : A_1$ and

$\Gamma \vdash a_r : A_r(a_0, a_1)$ and

$\Gamma \vdash b_0 : A_0$ and $\Gamma \vdash b_1 : A_1$ and

$\Gamma \vdash b_r : A_r(b_0, b_1)$ and

$\Gamma \vdash p_0 : \mathrm{Id}_{A_0}(a_0, b_0)$ and $\Gamma \vdash p_1 : \mathrm{Id}_{A_1}(a_1, b_1)$ and

$\Gamma \vdash p_r : \mathrm{R}_{\mathrm{Id_A}}(p_0, p_1)$

Computation:

$\Gamma \vdash \mathsf{ind}_{\mathrm{R}_{\mathrm{Id_A}}}(a_r, a_r, \rho_{\mathsf{refl}(a)}, c) \equiv c : C(\mathsf{refl}(a_0), \mathsf{refl}(a_1), \rho_{\mathsf{refl}(a)})$

For the reasons pointed out in [148], in defining an interpretation of a dependent type theory it is convenient to start from raw expressions (including possibly non-well typed expressions) and then single out the well-formed ones by means of the typing rules. To this end, I first introduce the pre-syntax of **MLTT** and then I define a partial interpretation function sending **MLTT**-raw expressions to raw expressions in an extension of **MLTT**, which I call $\mathbf{MLTT}_{rel}$, that is obtained by adding to the syntax the relational constructors previously defined. Finally, I show that the interpretation is correct, in the sense that if a judgement $\mathcal{J}$ is derivable in **MLTT** then it is valid w.r.t. to the partial interpretation.

Definition 3.2.9. [Pre-Syntax of **MLTT**] The pre-syntax is given by the following grammar:

$$\Gamma \ ::= \ [\cdot] \mid \ \Gamma, x : A \text{ (provided x is not in } \Gamma)$$

$$A, B \ ::= \ \Pi_{(x:A)} B(x) \mid \ \Sigma_{(x:A)} B(x) \mid \mathbf{0} \mid \mathbf{1} \mid \mathsf{Bool} \mid \mathbb{N} \mid \mathrm{Id}_A(x, y) \mid A + B$$

$$\begin{aligned} a, b, c, d ::= \ & x \mid \lambda x : A.b \mid \langle a, b \rangle \mid * \mid \mathsf{tt} \mid \mathsf{ff} \mid 0 \mid \mathsf{s}(a) \mid \mathsf{refl}(a) \mid \\ & \mathsf{inl}(a) \mid \mathsf{inr}(b) \mid c(a) \mid \mathsf{ind}_\Sigma(c, x.y.d) \mid \mathsf{ind}_{\mathbf{0}}(c) \mid \\ & \mathsf{ind}_{\mathbf{1}}(a, c) \mid \mathsf{ind}_{\mathsf{Bool}}(c, a, b) \mid \mathsf{ind}_{\mathbb{N}}(a, c, x.y.d) \mid \\ & \mathsf{ind}_{\mathrm{Id}}(z.c, a, b, d) \mid \mathsf{ind}_+(c, x.a, y.b) \end{aligned}$$

Accordingly, there are the following *pre-judgement* forms:

Definition 3.2.10. The pre-judgements of **MLTT** are the following:

1. $\Gamma \vdash A$
2. $\Gamma \vdash a : A$
3. $\Gamma \vdash A \equiv B$
4. $\Gamma \vdash a \equiv b : A$

Definition 3.2.11. The partial interpretation map $[\![-]\!]$ is meant to associate to each expression in **MLTT**-raw syntax an expression in the raw syntax of the extended theory $\mathbf{MLTT}_{rel}$ and is defined as follows:

Contexts:

- $[\![[\cdot]]\!] :\equiv [\cdot]$
- $[\![\Gamma, x : A]\!] :\equiv [\![\Gamma]\!], x_0 : A_0, x_1 : A_1, x_r : A_r(x_0, x_1)$

A variable x is translated as a triple x_0, x_1, x_r and the raw-judgement forms are interpreted as follows:

- $[\![\Gamma \vdash A]\!] :\equiv [\![\Gamma]\!], x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1)$ *type*
- $[\![\Gamma \vdash a : A]\!] :\equiv [\![\Gamma]\!] \vdash a_r : A_r(a_0, a_1)$, provided
  $[\![\Gamma]\!] \vdash a_0 : A_0$ and $[\![\Gamma]\!] \vdash a_1 : A_1$
- $[\![\Gamma \vdash A \equiv B]\!] :\equiv [\![\Gamma]\!], x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1) \equiv B_r(x_0, x_1)$,
  provided
  $[\![\Gamma]\!] \vdash A_0 \equiv B_0$ and $[\![\Gamma]\!] \vdash A_1 \equiv B_1$

- $[\![\Gamma \vdash a \equiv b : A]\!] :\equiv [\![\Gamma]\!] \vdash a_r \equiv b_r : A_r(a_0, a_1)$

  provided

  $[\![\Gamma]\!] \vdash a_0 \equiv b_0 : A_0$ and $[\![\Gamma]\!] \vdash a_1 \equiv b_1 : A_1$

The following clauses extend the interpretation to **MLTT**-raw terms and types constructors:

$$
\begin{aligned}
[\![\mathbf{1}]\!] &:\equiv \mathrm{R}_{\mathbf{1}}(x_0, x_1)\ [x_0, x_1 : \mathbf{1}] \\
[\![*]\!] &:\equiv \rho_* \\
[\![\mathsf{ind}_{\mathbf{1}}(a, c)]\!] &:\equiv \mathsf{ind}_{\mathrm{R}_{\mathbf{1}}}([\![a]\!], [\![c]\!])
\end{aligned}
$$

$$
\begin{aligned}
[\![\mathbf{0}]\!] &:\equiv \mathrm{R}_{\mathbf{0}}(x_0, x_1)\ [x_0, x_1 : \mathbf{0}] \\
[\![\mathsf{ind}_{\mathbf{0}}(a)]\!] &:\equiv \mathsf{ind}_{\mathrm{R}_{\mathbf{0}}}([\![a]\!])
\end{aligned}
$$

$$
\begin{aligned}
[\![\mathsf{Bool}]\!] &:\equiv \mathrm{R}_{\mathsf{bool}}(x_0, x_1)\ [x_0, x_1 : \mathsf{Bool}] \\
[\![\mathsf{tt}]\!] &:\equiv \rho_{\mathsf{tt}} \\
[\![\mathsf{ff}]\!] &:\equiv \rho_{\mathsf{ff}} \\
[\![\mathsf{ind}_{\mathsf{Bool}}(c, a, b)]\!] &:\equiv \mathsf{ind}_{\mathrm{R}_{\mathsf{bool}}}([\![c]\!], [\![a]\!], [\![b]\!])
\end{aligned}
$$

$$
\begin{aligned}
[\![\mathbb{N}]\!] &:\equiv \mathrm{R}_{\mathbb{N}}(x_0, x_1)\ [x_0, x_1 : \mathbb{N}] \\
[\![0]\!] &:\equiv \rho_0 \\
[\![\mathsf{s}(a)]\!] &:\equiv \rho_{\mathsf{s}(a)} \\
[\![\mathsf{ind}_{\mathbb{N}}(a, c, d)]\!] &:\equiv \mathsf{ind}_{\mathrm{R}_{\mathbb{N}}}([\![a]\!], [\![c]\!], [\![d]\!])
\end{aligned}
$$

$$
\begin{aligned}
[\![\Pi_{(x:A)}B(x)]\!] &:\equiv \mathrm{R}_{\Pi(A,B)}(z_0, z_1)\ [z_0 : \Pi_{(x_0:A_0)}B_0(x_0), z_1 : \Pi_{(x_1:A_1)}B_1(x_1)] \\
[\![\lambda x : A.b]\!] &:\equiv \rho_{\lambda(b)} \\
[\![f(a)]\!] &:\equiv \rho_{f(a)}
\end{aligned}
$$

$$
\begin{aligned}
[\![\Sigma_{(x:A)}B(x)]\!] &:\equiv \mathrm{R}_{\Sigma(A,B)}(z_0, z_1)\ [z_0 : \Sigma_{(x_0:A_0)}B_0(x_0), z_1 : \Sigma_{(x_1:A_1)}B_1(x_1)] \\
[\![(a, b)]\!] &:\equiv \rho_{(a,b)} \\
[\![\mathsf{ind}_{\Sigma(A,B)}(c, d)]\!] &:\equiv \mathsf{ind}_{\mathrm{R}_{\Sigma(A,B)}}([\![c]\!], [\![d]\!])
\end{aligned}
$$

$$
\begin{aligned}
[\![A + B]\!] &:\equiv \mathrm{R}_{A+B}(z_0, z_1)\ [z_0 : A_0 + B_0, z_1 : A_1 + B_1] \\
[\![\mathsf{inl}(a)]\!] &:\equiv \rho_{\mathsf{inl(a)}} \\
[\![\mathsf{inr}(b)]\!] &:\equiv \rho_{\mathsf{inr(b)}} \\
[\![\mathsf{ind}_{A+B}(c, a, b)]\!] &:\equiv \mathsf{ind}_{\mathrm{R}_{A+B}}([\![c]\!], [\![a]\!], [\![b]\!])
\end{aligned}
$$

$$
\begin{aligned}
[\![\mathrm{Id}_A(a, b)]\!] &:\equiv \mathrm{R}_{\mathrm{Id}_A}(z_0, z_1)\ [z_0 : \mathrm{Id}_{A_0}(a_0, b_0), z_1 : \mathrm{Id}_{A_1}(a_1, b_1)] \\
[\![\mathsf{refl}(a)]\!] &:\equiv \rho_{\mathsf{refl(a)}} \\
[\![\mathsf{ind}_{\mathrm{Id}_A}(a, b, d, c)]\!] &:\equiv \mathsf{ind}_{\mathrm{R}_{\mathrm{Id}_A}}([\![a]\!], [\![b]\!], [\![d]\!], [\![c]\!])
\end{aligned}
$$

The following lemma is meant to express the fact that the interpretation just defined preserves substitution:

Lemma 3.2.12 (Substitution)**.** *Let A be a pre-type, x a free variable in A and a a pre-term, then*

$$[\![\Gamma \vdash A[a/x]]\!] :\equiv [\![\Gamma]\!] \vdash [\![A]\!][[\![a]\!]/[\![x]\!]]$$

and let a, b be pre-terms and x a variable free in a, then

$$[\![\Gamma \vdash a[b/x]]\!] :\equiv [\![\Gamma]\!] \vdash [\![a]\!][[\![b]\!]/[\![x]\!]]$$

Proof. The proof is by structural induction over the derivation of the raw expressions. □

Theorem 3.2.13 (Validity)**.** *The following statements hold:*

1. *If $\Gamma \vdash A$ is derivable in* **MLTT***, then $[\![\Gamma \vdash A]\!]$ is well-defined;*
2. *If $\Gamma \vdash a : A$ is derivable in* **MLTT***, then $[\![\Gamma \vdash a : A]\!]$ is well-defined;*
3. *If $\Gamma \vdash A \equiv B$ is derivable in* **MLTT***, then $[\![\Gamma \vdash A \equiv B]\!]$ is well-defined;*
4. *If $\Gamma \vdash a \equiv b : A$ is derivable in* **MLTT***, then $[\![\Gamma \vdash a \equiv b : A]\!]$ is well-defined.*

Therefore, if $\mathcal{J}$ is a derivable judgement, then it is valid w.r.t. the partial interpretation $[\![-]\!]$.

Proof. The proof is by structural induction over the derivations.
Conversion Rule:

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash A \equiv B}{\Gamma \vdash a : B}$$

By induction hypothesis,

$$[\![\Gamma \vdash a : A]\!]$$

yields

$$[\![\Gamma]\!] \vdash a_r : A_r(a_0, a_1)$$

with $[\![\Gamma]\!] \vdash a_0 : A_0$ and $[\![\Gamma]\!] \vdash a_1 : A_1$. The other premise $[\![\Gamma \vdash A \equiv B]\!]$ yields

$$[\![\Gamma]\!], x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1) \equiv B_r(x_0, x_1)$$

with $[\![\Gamma]\!] \vdash A_0 \equiv B_0$ and $[\![\Gamma]\!] \vdash A_1 \equiv B_1$. Therefore, we immediately get

$$[\![\Gamma]\!] \vdash a_0 : B_0 \text{ and } [\![\Gamma]\!] \vdash a_1 : B_1$$

and thus we conclude

$$[\![\Gamma]\!] \vdash a_r : B_r(a_0, a_1)$$

which amounts to the interpretation of the conclusion.
Bool-Formation:

$$[\![\Gamma \vdash \mathsf{Bool}\ \mathit{type}]\!]$$

is interpreted as $[\![\Gamma]\!], x_0, x_1 : \mathsf{Bool} \vdash \mathrm{R}_{\mathsf{bool}}(x_0, x_1)$ by definition of the interpretation.
Bool-Introduction: the interpretation of the introduction rules is also immediate by definition of the interpretation and amounts to

$$[\![\Gamma \vdash \mathsf{tt} : \mathsf{Bool}]\!] :\equiv [\![\Gamma]\!] \vdash \rho_{\mathsf{tt}} : \mathrm{R}_{\mathsf{bool}}(\mathsf{tt}, \mathsf{tt})$$

and

$$[\![\Gamma \vdash \mathsf{ff} : \mathsf{Bool}]\!] :\equiv [\![\Gamma]\!] \vdash \rho_{\mathsf{ff}} : \mathrm{R}_{\mathsf{bool}}(\mathsf{ff}, \mathsf{ff})$$

Bool-Elimination:

$$\frac{\Gamma, x : \mathsf{Bool} \vdash C(x)\ \mathit{type} \qquad \Gamma \vdash a : C(\mathsf{tt}) \qquad \Gamma \vdash b : C(\mathsf{ff}) \qquad \Gamma \vdash c : \mathsf{Bool}}{\Gamma \vdash \mathsf{ind}_{\mathsf{Bool}}(c, a, b) : C(c)}$$

By induction hypothesis, the premises are interpreted as follows: $[\![\Gamma, x : \mathsf{Bool} \vdash C(x)\ type]\!]$ yields

$$[\![\Gamma]\!], x_0 : \mathsf{Bool} \vdash C_0(x_0)\ type \text{ and } [\![\Gamma]\!], x_1 : \mathsf{Bool} \vdash C_1(x_1)\ type$$

and

$$[\![\Gamma]\!], x_0, x_1 : \mathsf{Bool}, x_r : \mathrm{R}_{\mathsf{bool}}(x_0, x_1), u_0 : C_0(x_0), u_1 : C_1(x_1) \vdash C_r(-)\ type$$

where $C_r(-)$ stands for $C_r(x_0, x_1, x_r, u_0, u_1)$.
The interpretation of the second premise $[\![\Gamma \vdash a : C(\mathsf{tt})]\!]$ is the following triple:

$$[\![\Gamma]\!] \vdash a_r : C_r(\mathsf{tt}, \mathsf{tt}, \rho_{\mathsf{tt}}, a_0, a_1)$$

with $[\![\Gamma]\!] \vdash a_0 : C_0(\mathsf{tt})$ and $[\![\Gamma]\!] \vdash a_1 : C_1(\mathsf{tt})$.
The interpretation of the premise $\Gamma \vdash b : C(\mathsf{ff})$ is completely similar. Finally, the last premise is interpreted as follows: $[\![\Gamma \vdash c : \mathsf{Bool}]\!]$ yields

$$[\![\Gamma]\!] \vdash c_r : \mathrm{R}_{\mathsf{bool}}(c_0, c_1)$$

Then, we immediately get

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\mathsf{Bool}}(c_0, a_0, b_0) : C_0(c_0)$$

and

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\mathsf{Bool}}(c_1, a_1, b_1) : C_1(c_1)$$

Therefore, by the elimination rule for $\mathrm{R}_{\mathsf{bool}}$, we conclude

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\mathrm{R}_{\mathsf{bool}}}(c_r, a_r, b_r) : C_r(c_0, c_1, c_r, \mathsf{ind}_{\mathsf{Bool}}(-_0), \mathsf{ind}_{\mathsf{Bool}}(-_1))$$

where $\mathsf{ind}_{\mathsf{Bool}}(-_0)$ and $\mathsf{ind}_{\mathsf{Bool}}(-_1)$ are abbreviations for $\mathsf{ind}_{\mathsf{Bool}}(c_0, a_0, b_0)$ and $\mathsf{ind}_{\mathsf{Bool}}(c_1, a_1, b_1)$ respectively and which amounts to the interpretation of the conclusion.
Bool-Computation: the computation rules are immediately seen to be valid by considering the interpretation of the other rules.
The other ground types $\mathbf{0}, \mathbf{1}$ and $\mathbb{N}$ are interpreted in a similar manner.
Π-**Formation**:

$$\frac{\Gamma \vdash A\ type \qquad \Gamma, x : A \vdash B(x)\ type}{\Gamma \vdash \Pi_{(x:A)} B(x)\ type}$$

By induction hypothesis, $[\![\Gamma \vdash A\ type]\!]$ yields

$$[\![\Gamma]\!], x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1)\ type$$

with $[\![\Gamma]\!] \vdash A_0\ type$ and $[\![\Gamma]\!] \vdash A_1\ type$, while other premise is interpreted as follows: $[\![\Gamma, x : A \vdash B(x)\ type]\!]$ yields

$$[\![\Gamma]\!], x_0 : A_0, x_1 : A_1, x_r : A_r(x_0, x_1), y_0 : B_0(x_0), y_1 : B_1(x_1) \vdash B_r(x_r, y_0, y_1)$$

with $[\![\Gamma]\!]x_0 : A_0 \vdash B_0(x_0)\ type$ and $[\![\Gamma]\!]x_1 : A_1 \vdash B_1(x_1)\ type$. Then, by $\mathrm{R}_{\Pi(A,B)}$-formation rule, we get

$$[\![\Gamma]\!], z_0 : \Pi_{(x_0:A_0)} B_0(x_0), z_1 : \Pi_{(x_1:A_1)} B_1(x_1) \vdash \mathrm{R}_{\Pi(A,B)}(z_0, z_1)\ type$$

which amounts to the interpretation of the conclusion.
Π-Introduction:

$$\frac{\Gamma, x : A \vdash b : B(x)\ type}{\Gamma \vdash \lambda x.b : \Pi_{(x:A)} B(x)}$$

By induction hypothesis, $[\![\Gamma, x : A \vdash b : B(x)\ type]\!]$ yields

$$[\![\Gamma]\!], x_0 : A_0, x_1 : A_1, x_r : A_r(x_0, x_1) \vdash b_r : B_r(x_r, b_0, b_1)$$

with $[\![\Gamma]\!], x_0 : A_0 \vdash b_0 : B_0(x_0)$ and $[\![\Gamma]\!], x_1 : A_1 \vdash b_1 : B_1(x_1)$.
Then, we immediately get

$$[\![\Gamma]\!] \vdash \lambda x_0.b_0 : \Pi_{(x_0:A_0)} B_0(x_0) \text{ and } [\![\Gamma]\!] \vdash \lambda x_1.b_1 : \Pi_{(x_1:A_1)} B_1(x_1)$$

and hence by $\mathrm{R}_{\Pi(A,B)}$-introduction, we conclude

$$[\![\Gamma]\!] \vdash \rho_{\lambda(b)} : \mathrm{R}_{\Pi(A,B)}(\lambda x_0.b_0, \lambda x_1.b_1)$$

which amounts to the interpretation of the conclusion.
Π-Elimination:

By induction hypothesis, $[\![\Gamma \vdash f : \Pi_{(x:A)}B(x)]\!]$ yields the triple:

$$[\![\Gamma]\!] \vdash f_0 : \Pi_{(x_0:A_0)}B_0(x_0) \text{ and } [\![\Gamma]\!] \vdash f_1 : \Pi_{(x_1:A_1)}B_1(x_1)$$

and

$$[\![\Gamma]\!] \vdash f_r : \mathrm{R}_{\Pi(A,B)}(f_0, f_1)$$

while $[\![\Gamma \vdash a : A]\!]$ yields

$$[\![\Gamma]\!] \vdash a_r : A_r(a_0, a_1)$$

with $[\![\Gamma]\!] \vdash a_0 : A_0$ and $[\![\Gamma]\!] \vdash a_1 : A_1$. Then, we can immediately obtain

$$[\![\Gamma]\!] \vdash f_0(a_0) : B_0(a_0) \text{ and } [\![\Gamma]\!] \vdash f_1(a_1) : B_1(a_1)$$

and thus by $\mathrm{R}_{\Pi(A,B)}$-elimination we conclude

$$[\![\Gamma]\!] \vdash \rho_{f(a)} : B_r(a_r, f_0(a_0), f_1(a_1))$$

which amounts to the interpretation of the conclusion.
Π-Computation: it follows from the previous cases.
Σ-Formation:

$$\frac{\Gamma \vdash A \ type \qquad \Gamma, x : A \vdash B(x) \ type}{\Gamma \vdash \Sigma_{(x:A)}B(x) \ type}$$

By induction hypothesis, the premises are interpreted as in the case of Π-formation. Then, we immediately get

$$[\![\Gamma]\!], z_0 : \Sigma_{(x_0:A_0)}B_0(x_0), z_1 : \Sigma_{(x_1:A_1)}B_1(x_1) \vdash \mathrm{R}_{\Sigma(A,B)}(z_0, z_1) \ type$$

Σ-Introduction:

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash b : B(a)}{\Gamma \vdash (a, b) : \Sigma_{(x:A)}B(x)}$$

By induction hypothesis, $[\![\Gamma \vdash a : A]\!]$ yields

$$[\![\Gamma]\!] \vdash a_r : A_r(a_0, a_1)$$

with $[\![\Gamma]\!] \vdash a_0 : A_0$ and $[\![\Gamma]\!] \vdash a_1 : A_1$. And $[\![\Gamma \vdash b : B(a)]\!]$ yields

$$[\![\Gamma]\!] \vdash b_r : B_r(a_r, b_0, b_1)$$

with $[\![\Gamma]\!] \vdash b_0 : B_0(a_0)$ and $[\![\Gamma]\!] \vdash b_1 : B_1(a_1)$. Therefore, we get

$$[\![\Gamma]\!] \vdash (a_0, b_0) : \Sigma_{(x_0:A_0)} B_0(x_0) \text{ and } [\![\Gamma]\!] \vdash (a_1, b_1) : \Sigma_{(x_1:A_1)} B_1(x_1)$$

and hence

$$[\![\Gamma]\!] \vdash \rho_{(a,b)} : \mathrm{R}_{\Sigma(A,B)}((a_0, b_0), (a_1, b_1))$$

which amounts to the interpretation of the conclusion.
Σ-Elimination:

$$\frac{\begin{array}{c}\Gamma, z : \Sigma_{(x:A)} B(x) \vdash C(z)\ \mathit{type} \\ \Gamma, x : A, y : B(x) \vdash d((x, y)) : C((x, y)) \\ \Gamma \vdash c : \Sigma_{(x:A)} B(x)\end{array}}{\Gamma \vdash \mathsf{ind}_{\Sigma(A,B)}(c, d) : C(c)}$$

By induction hypothesis, $[\![\Gamma, z : \Sigma_{(x:A)} B(x) \vdash C(z)\ \mathit{type}]\!]$ yields the triple:

$$[\![\Gamma]\!], z_0 : \Sigma_{(x_0:A_0)} B_0(x_0) \vdash C_0(z_0)\ \mathit{type}$$

and

$$[\![\Gamma]\!], z_1 : \Sigma_{(x_1:A_1)} B_1(x_1) \vdash C_1(z_1)\ \mathit{type}$$

and

$$[\![\Gamma]\!], \Delta, u_0 : C_0(x_0), u_1 : C_1(x_1) \vdash C_r(z_0, z_1, z_r, u_0, u_1)$$

where $\Delta \ :\equiv\ z_0 : \Sigma_{(x_0:A_0)} B_0(x_0), z_1 : \Sigma_{(x_1:A_1)} B_1(x_1), z_r : \mathrm{R}_{\Sigma(A,B)}(z_0, z_1)$.
The second premise is interpreted as follows: we have

$$[\![\Gamma]\!], x_0 : A_0, y_0 : B_0(x_0) \vdash d_0 : C_0((x_0, y_0))$$

and

$$[\![\Gamma]\!], x_1 : A_1, y_1 : B_1(x_1) \vdash d_1 : C_1((x_1, y_1))$$

and

$$[\![\Gamma]\!], \Theta \vdash d_r : C_r((x_0, y_0), (x_1, y_1), \rho_{(x,y)}, d_0, d_1)$$

where
$\Theta :\equiv x_0 : A_0, x_1 : A_1, x_r : A_r(x_0, x_1), y_0 : B_0(x_0), y_1 : B_1(x_1), y_r : B_r(x_r, y_0, y_1)$.

Finally, $[\![\Gamma \vdash c : \Sigma_{(x:A)} B(x)]\!]$ yields

$$[\![\Gamma]\!] \vdash c_r : \mathrm{R}_{\Sigma(A,B)}(c_0, c_1)$$

with $[\![\Gamma]\!] \vdash c_0 : \Sigma_{(x_0:A_0)} B_0(x_0)$ and $[\![\Gamma]\!] \vdash c_1 : \Sigma_{(x_1:A_1)} B_1(x_1)$.
Then, we immediately get

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\Sigma(A,B)}(c_0, d_0) : C_0(c_0) \text{ and } [\![\Gamma]\!] \vdash \mathsf{ind}_{\Sigma(A,B)}(c_1, d_1) : C_1(c_1)$$

and hence by $\mathrm{R}_{\Sigma(A,B)}$-elimination, we conclude

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\mathrm{R}_{\Sigma(A,B)}}(c_r, d_r) : C_r(c_0, c_1, c_r, \mathsf{ind}_{\Sigma(A,B)}(-_0), \mathsf{ind}_{\Sigma(A,B)}(-_1))$$

where $\mathsf{ind}_{\Sigma(A,B)}(-_0)$ and $\mathsf{ind}_{\Sigma(A,B)}(-_1)$ stand, respectively, for $\mathsf{ind}_{\Sigma(A,B)}(c_0, d_0)$ and $\mathsf{ind}_{\Sigma(A,B)}(c_1, d_1)$ and which amounts to the interpretation of the conclusion.
Σ-Computation: it follows from the previous cases.
$+$-Formation:

$$\frac{\Gamma \vdash A \; type \qquad \Gamma \vdash B \; type}{\Gamma \vdash A + B \; type}$$

By induction hypothesis, $[\![\Gamma \vdash A \; type]\!]$ yields

$$[\![\Gamma]\!], x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1) \; type$$

with $[\![\Gamma]\!] \vdash A_0 \; type$ and $[\![\Gamma]\!] \vdash A_1 \; type$ and the other premise is interpreted in a similar manner. Then, we can form $[\![\Gamma]\!] \vdash A_0 + B_0 \; type$ and $[\![\Gamma]\!] \vdash A_1 + B_1 \; type$ and thus we conclude

$$[\![\Gamma]\!], z_0 : A_0 + B_0, z_1 : A_1 + B_1 \vdash \mathrm{R}_{A+B}(z_0, z_1) \; type$$

which amounts to the interpretation of the conclusion.
$+$-Introduction:

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash \mathsf{inl}(a) : A + B} \qquad \frac{\Gamma \vdash b : B}{\Gamma \vdash \mathsf{inr}(b) : A + B}$$

By induction hypothesis, $[\![\Gamma \vdash a : A]\!]$ yields

$$[\![\Gamma]\!] \vdash a_r : A_r(a_0, a_1)$$

with $[\![\Gamma]\!] \vdash a_0 : A_0$ and $[\![\Gamma]\!] \vdash a_1 : A_1$.
Then, we immediately get $[\![\Gamma]\!] \vdash \mathsf{inl}(a_0) : A_0 + B_0$ and $[\![\Gamma]\!] \vdash \mathsf{inl}(a_1) : A_1 + B_1$ and by R_{A+B}-introduction

$$[\![\Gamma]\!] \vdash \rho_{\mathsf{inl(a)}} : \mathrm{R}_{A+B}(\mathsf{inl}(a_0), \mathsf{inl}(a_1))$$

The other introduction rule is validated in a similar manner.
+-Elimination:

$$\frac{\begin{array}{c}\Gamma, z : A + B \vdash C(z)\ \mathit{type} \\ \Gamma, x : A \vdash a : C(\mathsf{inl}(x)) \\ \Gamma, y : B \vdash b : C(\mathsf{inr}(y)) \\ \Gamma \vdash c : A + B\end{array}}{\Gamma \vdash \mathsf{ind}_{A+B}(c, a, b) : C(c)}$$

By induction hypothesis, the first premise is interpreted by the following triple:

$$[\![\Gamma]\!], z_0 : A_0 + B_0 \vdash C_0(z_0)\ \mathit{type} \text{ and } [\![\Gamma]\!], z_1 : A_1 + B_1 \vdash C_1(z_1)\ \mathit{type}$$

and

$$[\![\Gamma]\!], \Delta, u_0 : C_0(z_0), u_1 : C_1(z_1) \vdash C_r(z_0, z_1, z_r, u_0, u_1)\ \mathit{type}$$

where $\Delta \ :\equiv\ z_0 : A_0 + B_0, z_1 : A_1 + B_1, z_r : \mathrm{R}_{A+B}(z_0, z_1)$.
The interpretation of the second premise $[\![\Gamma, x : A \vdash a : C(\mathsf{inl}(x))]\!]$ yields

$$[\![\Gamma]\!], x_0 : A_0 \vdash a_0 : C_0(\mathsf{inl}(x_0)) \text{ and } [\![\Gamma]\!], x_1 : A_1 \vdash a_1 : C_1(\mathsf{inl}(x_1))$$

and

$$[\![\Gamma]\!], x_0 : A_0, x_1 : A_1, x_r : A_r(x_0, x_1) \vdash a_r : C_r(\mathsf{inl}(x_0), \mathsf{inl}(x_1), \rho_{\mathsf{inl(x)}}, a_0, a_1)$$

The third premise is interpreted in a similar way. Finally, the last premise is interpreted by $[\![\Gamma]\!] \vdash c_r : \mathrm{R}_{A+B}(c_0, c_1)$ with $[\![\Gamma]\!] \vdash c_0 : A_0 + B_0$ and $[\![\Gamma]\!] \vdash c_1 : A_1 + B_1$. Therefore, we immediately get

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{A+B}(c_0, a_0, b_0) : C_0(c_0)$$

and

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{A+B}(c_1, a_1, b_1) : C_1(c_1)$$

Then, we conclude by R_{A+B}-elimination

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\mathrm{R}_{A+B}}(c_r, a_r, b_r) : C_r(c_0, c_1, c_r, \mathsf{ind}_{A+B}(-_0), \mathsf{ind}_{A+B}(-_1))$$

which amounts to the interpretation of the conclusion.
+-Computation: the two computation rules follow from the previous cases.
Id-Formation:

$$\frac{\Gamma \vdash A\ type \qquad \Gamma \vdash a : A \qquad \Gamma \vdash b : A}{\Gamma \vdash \mathrm{Id}_A(a, b)\ type}$$

By induction hypothesis, $[\![\Gamma \vdash A\ type]\!]$ yields

$$[\![\Gamma]\!], x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1)\ type$$

with $[\![\Gamma]\!] \vdash A_0\ type$ and $[\![\Gamma]\!] \vdash A_1\ type$. While $[\![\Gamma \vdash a : A]\!]$ yields

$$[\![\Gamma]\!] \vdash a_r : A_r(a_0, a_1)$$

with $[\![\Gamma]\!] \vdash a_0 : A_0$ and $[\![\Gamma]\!] \vdash a_1 : A_1$. The other premise $\Gamma \vdash b : A$ is interpreted in a similar way. Therefore, we get immediately

$$[\![\Gamma]\!] \vdash \mathrm{Id}_{A_0}(a_0, b_0)\ type \text{ and } [\![\Gamma]\!] \vdash \mathrm{Id}_{A_1}(a_1, b_1)\ type$$

and thus we conclude

$$[\![\Gamma]\!], z_0 : \mathrm{Id}_{A_0}(a_0, b_0), z_1 : \mathrm{Id}_{A_1}(a_1, b_1) \vdash \mathrm{R}_{\mathrm{Id_A}}(z_0, z_1)\ type$$

which amounts to the interpretation of the conclusion.
Id-Introduction:

$$\frac{\Gamma \vdash a : A}{\Gamma \vdash \mathsf{refl}(a) : \mathrm{Id}_A(a_0, a_1)}$$

Let us assume the premise as valid by induction hypothesis, then we can form

$$[\![\Gamma]\!] \vdash \mathsf{refl}(a_0) : \mathrm{Id}_{A_0}(a_0, a_0) \text{ and } [\![\Gamma]\!] \vdash \mathsf{refl}(a_1) : \mathrm{Id}_{A_1}(a_1, a_1)$$

and hence we conclude by $\mathrm{R}_{\mathrm{Id_A}}$-intro

$$[\![\Gamma]\!] \vdash \rho_{\mathsf{refl(a)}} : \mathrm{R}_{\mathrm{Id_A}}(\mathsf{refl}(a_0), \mathsf{refl}(a_1))$$

Id-Elimination:

$$\frac{\begin{array}{c}\Gamma, x, y : A, p : \mathrm{Id}_A(x,y) \vdash C(x,y,q)\ \mathit{type} \\ \Gamma, z : A \vdash c : C(z,z,\mathsf{refl}(z)) \\ \Gamma \vdash a : A \\ \Gamma \vdash b : A \\ \Gamma \vdash p : \mathrm{Id}_A(a,b)\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathrm{Id}_{\mathrm{A}}}(a,b,p,c) : C(a,b,p)}$$

By induction hypothesis, the first premise yields the following triple:

$$[\![\Gamma]\!], x_0, y_0 : A_0, q_0 : \mathrm{Id}_{A_0}(x_0,y_0) \vdash C_0(x_0,y_0,q_0)\ \mathit{type}$$

and

$$[\![\Gamma]\!], x_1, y_1 : A_1, q_1 : \mathrm{Id}_{A_1}(x_1,y_1) \vdash C_1(x_1,y_1,q_1)\ \mathit{type}$$

and

$$[\![\Gamma]\!], \Delta, u_0 : C_0(x_0,y_0,q_0), u_1 : C_1(x_1,y_1,q_1) \vdash C_r(x_r,y_r,q_r,u_0,u_1)$$

where $\Delta :\equiv x_0 : A_0, x_1 : A_1, x_r : A_r(x_0,x_1), y_0 : A_0, y_1 : A_1, y_r : A_r(y_0,y_1), q_0 : \mathrm{Id}_{A_0}(x_0,y_0), q_1 : \mathrm{Id}_{A_1}(x_1,y_1), q_r : \mathrm{R}_{\mathrm{Id}_{\mathrm{A}}}(q_0,q_1)$.
The interpretation of the second premise $[\![\Gamma, z : A \vdash c : C(z,z,\mathsf{refl}(z))]\!]$ yields

$$[\![\Gamma]\!], z_0 : A_0 \vdash c_0 : C_0(z_0,z_0,\mathsf{refl}(z_0))$$

and

$$[\![\Gamma]\!], z_1 : A_1 \vdash c_1 : C_1(z_1,z_1,\mathsf{refl}(z_1))$$

and

$$[\![\Gamma]\!], z_0 : A_0, z_1 : A_1, z_r : A_r(z_0,z_1) \vdash c_r : C_r(z_r,z_r,\rho_{\mathsf{refl(z)}},c_0,c_1)$$

The interpretation of the other three premises is straightforward and I will not spell it out. Then, we immediately get

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\mathrm{Id}_{\mathrm{A}}}(a_0,b_0,p_0,c_0) : C_0(c_0)$$

and

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\mathrm{Id}_{\mathrm{A}}}(a_1,b_1,p_1,c_1) : C_1(c_1)$$

and hence by $\mathrm{R}_{\mathrm{Id}_{\mathrm{A}}}$-elimination we conclude

$$[\![\Gamma]\!] \vdash \mathsf{ind}_{\mathrm{R}_{\mathrm{Id}_{\mathrm{A}}}}(a_r,b_r,p_r,c_r) : C_r(p_0,p_1,p_r,\mathsf{ind}_{\mathrm{Id}_{\mathrm{A}}}(-_0),\mathsf{ind}_{\mathrm{Id}_{\mathrm{A}}}(-_1))$$

which amounts to the interpretation of the conclusion.
Id-Computation: it follows from the previous cases. □

3.3 Invariance under Equivalence

The key insight to get a proof of the Principle of Invariance under Equivalence for dependent type theory is to refine the construction presented in the previous section by requiring that the type-valued relation be an *equivalence* in the sense of [155]. Intuitively, the idea is to define a sub-model Rel^{equ} of the relation model Rel. In this way, a proof of the validity of this refined interpretation automatically entails the equivalence-invariance of all definable properties of the system.
First, we should try to define a notion of equivalence which is adequate for general type-valued relations or correspondences. In the ordinary case of prop-valued relations, the natural assumption is that such relations must be total and one-to-one: given a relation $x : A, y : B \vdash R(x, y) : \mathsf{Prop}$, the two conditions are defined as follows

$$\textbf{(R-total)}\ \Pi_{(x:A)}\Sigma_{(y:B)}R(x,y) \times \Pi_{(y:B)}\Sigma_{(x:A)}R(x,y)$$

$$\textbf{(R one-to-one)}\ \Pi_{(x:A)}\Pi_{(y,z:B)}R(x,y) \times R(x,z) \to y = z$$

$$\textbf{(R one-to-one)}\ \Pi_{(y:B)}\Pi_{(x,z:A)}R(x,y) \times R(z,y) \to x = z$$

Luckily, the generalisation to arbitrary type-valued relations can be already found in the standard reference [155]:

Definition 3.3.1. A correspondence $R : A \to B \to \mathcal{U}$ is said to be one-to-one if the following type is inhabited:

$$\Pi_{(x:A)}\mathsf{isContr}(\Sigma_{(y:B)}R(x,y)) \times \Pi_{(y:B)}\mathsf{isContr}(\Sigma_{(x:A)}R(x,y))$$

and recall that $\mathsf{isContr}(A) :\equiv \Sigma_{(a:A)}\Pi_{(x:A)}\mathrm{Id}_A(a,x)$. Then, the type of one-to-one correspondences between A and B, $\mathsf{1to1}(A,B)$, is defined as:

$$\Sigma_{(R:A\to B\to\mathcal{U})}(\Pi_{(x:A)}\mathsf{isContr}(\Sigma_{(y:B)}R(x,y)) \times \Pi_{(y:B)}\mathsf{isContr}(\Sigma_{(x:A)}R(x,y)))$$

Remark 3.3.2. The type $\mathsf{1to1}(A,B)$ necessarily lives into a universe higher than $\mathcal{U}$.

The crucial fact is that this type is equivalent to the type of equivalences as defined in Def.2.4.3.

Proposition 3.3.3. *The type* $\mathsf{1to1}(A, B)$ *is equivalent to the type* $A \simeq B$.

Proof. Let us assume that the type $\mathsf{1to1}(A, B)$ is inhabited. We have to build an equivalence f from A to B. The assumption that $\mathsf{1to1}(A, B)$ is inhabited means that there is $R : A \to B \to \mathcal{U}$ such that

$$(h, k) : \Pi_{(x:A)}\mathsf{isContr}(\Sigma_{(y:B)}R(x, y)) \times \Pi_{(y:B)}\mathsf{isContr}(\Sigma_{(x:A)}R(x, y))$$

Now $\Pi_{(x:A)}\mathsf{isContr}(\Sigma_{(y:B)}R(x, y)) :\equiv \Pi_{(x:A)}\exists!_{(y:B)}R(x, y)$. Then by the validity of AC!, we get a map $f : A \to B$ such that $\Pi_{(x:A)}R(x, f(x))$ holds. By applying the same reasoning to the second conjunct, we obtain a map $g : B \to A$ such that $\Pi_{(y:B)}R(g(y), y)$.
For $a : A$, $h(a) : \mathsf{isContr}(\Sigma_{(y:B)}R(a, y))$ with center of contraction (b, q) and for $b : B$, $k(b) : \mathsf{isContr}(\Sigma_{(x:A)}R(x, b))$ with center (a, p).
Then, $k(f(a)) : \mathsf{isContr}(\Sigma_{(x:A)}R(x, f(a)))$ with center (a', p'). It follows that $g(f(a)) = a'$, but $(a, p) = (a', p')$ and hence $\mathsf{pr}_1(a, p) = \mathsf{pr}_1(a', p')$. Then we conclude $g(f(a)) = a' = a$. The same reasoning can be applied in the other direction for $b : B$ so as to get $f(g(b)) = b$. Therefore, we have that f is an equivalence and $f^{-1} :\equiv g$.
On the other hand, let us suppose that f is an equivalence from A to B. We can define $\mathrm{R}_f :\equiv \lambda x.\lambda y.f(x) = y$. We know from Lemma 3.11.8 in [155] that the type $\Sigma_{(x:A)}a = x$ is contractible for any type A and any $a : A$. Then, in particular, $\Sigma_{(y:B)}f(x) = y$ is contractible with center $(f(x), \mathsf{refl}_{f(x)})$. Also, $\Sigma_{(x:A)}f(x) = y$ is contractible with center $(f^{-1}(y), q)$, where $q : y = f(f^{-1}(y))$. Therefore we can conclude that R_f is a one-to-one correspondence between A and B. □

As anticipated, the idea is to refine the interpretation of a (closed) type A as follows:

$$\begin{aligned} &\vdash A_0 \ \mathit{type} \\ &\vdash A_1 \ \mathit{type} \\ &x_0 : A_0, x_1 : A_1 \vdash A_\varepsilon(x_0, x_1) \end{aligned}$$

where $A_\varepsilon(x_0, x_1)$ corresponds to $A_r(x_0, x_1)$ plus a proof that A_r is one-to-one and thus is an equivalence.

The relational type constructors are defined as in the previous section. However, we must supply additional data witnessing that such correspondences are one-to-one. In the case of base types, by analyzing a value of their relational counterparts it is immediate to recover the arguments of these correspondences. In particular, it follows that they correspond to the identities which trivially satisfy the definition of equivalences. Then, in the light of Proposition 3.3.3, we can conclude that such correspondences are one-to-one and such information is encoded by means of the following proof-terms:

$$\varepsilon_{\mathrm{R}_{\mathbf{0}}} : \mathsf{1to1}(\mathrm{R}_{\mathbf{0}}) \qquad \varepsilon_{\mathrm{R}_{\mathbf{1}}} : \mathsf{1to1}(\mathrm{R}_{\mathbf{1}})$$

$$\varepsilon_{\mathrm{R}_{\mathsf{bool}}} : \mathsf{1to1}(\mathrm{R}_{\mathsf{bool}}) \qquad \varepsilon_{\mathrm{R}_{\mathbb{N}}} : \mathsf{1to1}(\mathrm{R}_{\mathbb{N}})$$

For the other type constructors, it is enough to make appeal to the fact that they are congruent w.r.t. equivalences in the following sense:

Lemma 3.3.4. *1. If* $f : A_0 \simeq A_1$ *and* $g : \Pi_{(x:A_0)} B_0(x) \simeq B_1(f(x))$*, then*

$$\Pi_{(x_0:A_0)} B_0(x_0) \simeq \Pi_{(x_1:A_1)} B_1(x_1)$$

is inhabited.

2. If $f : A_0 \simeq A_1$ *and* $g : \Pi_{(x:A_0)} B_0(x) \simeq B_1(f(x))$*, then*

$$\Sigma_{(x_0:A_0)} B_0(x_0) \simeq \Sigma_{(x_1:A_1)} B_1(x_1)$$

is inhabited.

3. If $f : A_0 \simeq A_1$ *and* $g : B_0 \simeq B_1$*, then*

$$A_0 + B_0 \simeq A_1 + B_1$$

is inhabited.

4. If $f : A_0 \simeq A_1$*,* $p : \mathrm{Id}_{A_1}(f(x_0), x_1)$ *and* $q : \mathrm{Id}_{A_1}(f(y_0), y_1)$*, then*

$$\mathrm{Id}_{A_0}(x_0, y_0) \simeq \mathrm{Id}_{A_1}(x_1, y_1)$$

is inhabited.

Proof. The proof of these statements can be found in the **Coq HoTT** library [13]. It should be noted that the proof in the case of Π-types makes essential use of function extensionality. □

By extracting the relevant one-to-one correspondences from the previous equivalences, we can encode the fact that the relational type constructors are one-to-one by means of the following proof-terms:

$$\varepsilon_{\mathrm{R}_\Pi} : \mathsf{1to1}(\mathrm{R}_{\Pi(A,B)}) \qquad \varepsilon_{\mathrm{R}_\Sigma} : \mathsf{1to1}(\mathrm{R}_{\Sigma(A,B)})$$
$$\varepsilon_{\mathrm{R}_+} : \mathsf{1to1}(\mathrm{R}_{+(A,B)}) \qquad \varepsilon_{\mathrm{R}_{\mathrm{Id}}} : \mathsf{1to1}(\mathrm{R}_{\mathrm{Id}_A})$$

Now the partial interpretation $[\![-]\!]$ can be refined to a partial interpretation $[\![-]\!]^{\mathrm{e}}$ which is defined as follows:

Definition 3.3.5. The interpretation for contexts is given by

- $[\![[-]]\!]^{\mathrm{e}} :\equiv [-]$
- $[\![\Gamma, x : A]\!]^{\mathrm{e}} :\equiv [\![\Gamma]\!]^{\mathrm{e}}, x_0 : A_0, x_1 : A_1, x_\varepsilon : A_\varepsilon(x_0, x_1)$

A variable x is send to a triple x_0, x_1, x_ε and the judgements are interpreted as follows:

- $[\![\Gamma \vdash A]\!]^{\mathrm{e}} :\equiv [\![\Gamma]\!]^{\mathrm{e}}, x_0 : A_0, x_1 : A_1 \vdash A_\varepsilon(x_0, x_1)$ *type*
- $[\![\Gamma \vdash a : A]\!]^{\mathrm{e}} :\equiv [\![\Gamma]\!]^{\mathrm{e}} \vdash a_\varepsilon : A_\varepsilon(a_0, a_1)$, provided
  $[\![\Gamma]\!]^{\mathrm{e}} \vdash a_0 : A_0$ and $[\![\Gamma]\!]^{\mathrm{e}} \vdash a_1 : A_1$
- $[\![\Gamma \vdash A \equiv B]\!]^{\mathrm{e}} :\equiv [\![\Gamma]\!]^{\mathrm{e}}, x_0 : A_0, x_1 : A_1 \vdash A_\varepsilon(x_0, x_1) \equiv B_\varepsilon(x_0, x_1)$,
  provided
  $[\![\Gamma]\!]^{\mathrm{e}} \vdash A_0 \equiv B_0$ and $[\![\Gamma]\!]^{\mathrm{e}} \vdash A_1 \equiv B_1$
- $[\![\Gamma \vdash a \equiv b : A]\!]^{\mathrm{e}} :\equiv [\![\Gamma]\!]^{\mathrm{e}} \vdash a_\varepsilon \equiv b_\varepsilon : A_\varepsilon(a_0, a_1)$
  provided
  $[\![\Gamma]\!]^{\mathrm{e}} \vdash a_0 \equiv b_0 : A_0$ and $[\![\Gamma]\!]^{\mathrm{e}} \vdash a_1 \equiv b_1 : A_1$

The raw-type symbols of **MLTT** are interpreted as pairs, where the second component encodes the fact that the correspondence is one-to-one:

$$
\begin{aligned}
[\![\mathbf{0}]\!]^{\mathrm{e}} &:\equiv (\mathrm{R}_{\mathbf{0}}, \varepsilon_{\mathrm{R}_{\mathbf{0}}}) \\
[\![\mathbf{1}]\!]^{\mathrm{e}} &:\equiv (\mathrm{R}_{\mathbf{1}}, \varepsilon_{\mathrm{R}_{\mathbf{1}}}) \\
[\![\mathsf{Bool}]\!]^{\mathrm{e}} &:\equiv (\mathrm{R}_{\mathsf{bool}}, \varepsilon_{\mathrm{R}_{\mathsf{bool}}}) \\
[\![\mathbb{N}]\!]^{\mathrm{e}} &:\equiv (\mathrm{R}_{\mathbb{N}}, \varepsilon_{\mathrm{R}_{\mathbb{N}}}) \\
[\![\Pi_{(x:A)} B(x)]\!]^{\mathrm{e}} &:\equiv (\mathrm{R}_{\Pi(A,B)}, \varepsilon_{\mathrm{R}_{\Pi}}) \\
[\![\Sigma_{(x:A)} B(x)]\!]^{\mathrm{e}} &:\equiv (\mathrm{R}_{\Sigma(A,B)}, \varepsilon_{\mathrm{R}_{\Sigma}}) \\
[\![A + B]\!]^{\mathrm{e}} &:\equiv (\mathrm{R}_{+(A,B)}, \varepsilon_{\mathrm{R}_{+}}) \\
[\![\mathrm{Id}_A(a,b)]\!]^{\mathrm{e}} &:\equiv (\mathrm{R}_{\mathrm{Id}_A}, \varepsilon_{\mathrm{R}_{\mathrm{Id}}})
\end{aligned}
$$

The definition extends to the term constructors in the same way as in the previous interpretation.

The substitution lemma is also valid for the interpretation $[\![-]\!]^{\mathrm{e}}$ and thus we can finally state the validity theorem:

Theorem 3.3.6 (Validity)**.** *The following statements hold:*

1. *If* $\Gamma \vdash A$*, then* $[\![\Gamma \vdash A]\!]^{\mathrm{e}}$ *is well-defined;*
2. *If* $\Gamma \vdash a : A$*, then* $[\![\Gamma \vdash a : A]\!]^{\mathrm{e}}$ *is well-defined;*
3. *If* $\Gamma \vdash A \equiv B$*, then* $[\![\Gamma \vdash A \equiv B]\!]^{\mathrm{e}}$ *is well-defined;*
4. *If* $\Gamma \vdash a \equiv b : A$*, then* $[\![\Gamma \vdash a \equiv b : A]\!]^{\mathrm{e}}$*.*

Therefore, if $\mathcal{J}$ *is a derivable judgement, then it is valid w.r.t. the partial interpretation* $[\![-]\!]^{\mathrm{e}}$*.*

Proof. The proof is by induction over the derivation of $\mathcal{J}$. It follows essentially from the proof of the Thm.3.2.13 and the derivation of the witnesses that the correspondences induced by the interpretation are one-to-one. □

The validity of the interpretation implies the validity of the principle of invariance: in particular, any definable property P must respect equivalences and it follows from how the interpretation is defined. First, the interpretation can be extended to type variables: $[\![X\ \mathit{type}]\!]^{\mathrm{e}}$ corresponds to a triple $X_0\ \mathit{type}$, $X_1\ \mathit{type}$ and $X_\varepsilon(x_0, x_1)\ [x_0 : X_0, x_1 : X_1]$. P is taken to be a definable property, which means $X\ \mathit{type} \vdash P(X)\ \mathit{type}$. Then P is interpreted as follows:

$$[\![P(X)]\!]^{\mathrm{e}} \ :\equiv \ P_\varepsilon(z_0, z_1)\ \mathit{type}\ [z_0 : P_0(X_0), z_1 : P_1(X_1)]$$

together with $P_0(X_0)$ *type* and $P_1(X_1)$ *type*.
Let us assume that $A \simeq B$ holds for a pair of types A, B. Then, the interpretation $[\![A \simeq B]\!]^{\mathrm{e}}$ amounts to the following triple: $A_0 \simeq B_0$, $A_1 \simeq B_1$ and $A_\varepsilon(a_0, a_1) \simeq B_\varepsilon(b_0, b_1)$, where $a_0 : A_0, a_1 : A_1, b_0 : B_0, b_1 : B_1$. Since P is built out from the constructors, by Theorem 3.3.6 we get

$$P_0(A_0) \simeq P_0(B_0) \text{ and } P_1(A_1) \simeq P_1(B_1).$$

Moreover, given $p_0 : P_0(A_0)$, $p_1 : P_1(A_1)$, $q_0 : P_0(B_0)$ and $q_1 : P_1(B_1)$, we get

$$P_\varepsilon(p_0, p_1) \simeq P_\varepsilon(q_0, q_1).$$

Therefore, we have obtained the triple corresponding to the interpretation of $P(A) \simeq P(B)$ under the assumption that $A \simeq B$ holds.

3.4 Discussion

The aim of this section is to discuss some relevant aspects of the interpretations defined in the previous sections.
Syntactic Model. A remarkable feature of the model construction previously defined lies in the fact that preserves judgemental equalities and conversion is interpreted as conversion in the meta-theory. In this sense, the relational model can be understood as an instance of a syntactic model construction as described in [18]. In particular,

- the interpretation is a translation from a type theory (the source) to another type theory (the target).

- Computational adequacy holds:

$$\text{If } \Gamma \vdash A \equiv B, \text{then } [\![\Gamma \vdash A \equiv B]\!]$$

Furthermore, the rules for the 'relational' type constructors should fit the general scheme for inductively defined types [42]. Therefore, the meta-theory corresponds to the source type theory extended with inductively defined families of types. Presenting the construction as a syntactic model guarantees that computational information is preserved by the translation.

Universes. The interpretation can be extended in order to deal with universes. A rough sketch goes as follows: we define a correspondence

$$\mathrm{R}_{\mathcal{U}_i} : \mathcal{U}_i \to \mathcal{U}_i \to \mathcal{U}_{i+1}$$

and the introduction rules are provided by the formation rules for the relational constructors: e.g. $\mathrm{R}_{\mathbb{N}} : \mathrm{R}_{\mathcal{U}}(\mathbb{N}, \mathbb{N})$. The interpretation of the judgement $\Gamma \vdash \mathcal{U}_i \; type$ works as expected:

$$[\![\Gamma \vdash \mathcal{U}_i \; type]\!] \;:\equiv\; [\![\Gamma]\!], X, Y : \mathcal{U}_i \vdash \mathrm{R}_{\mathcal{U}_i}(X, Y) : \mathcal{U}_{i+1}$$

I have assumed a Russell version of universes, but the definition can be given also for Tarski universes.

A key point of Awodey's argument is that the introduction of universes allows to define *non-invariant* properties. His example is to consider a predicate $P(X) :\equiv A =_{\mathcal{U}} X$ that involves the identity type of the universe. Then, if $A \simeq B$, we have that $P(A)$ is trivially true and hence the validity of the principle of invariance amounts to show that $A \simeq B \to P(B)$, that is $A =_{\mathcal{U}} B$. This statement is clearly implied by Univalence, but it cannot be derived without assuming this axiom. Therefore, Awodey concludes that Univalence provides a way to restore (and internalise) the invariance under equivalence principle [2].

In the light of the refined model construction of the previous section, it is possible to require that a universe is interpreted by a correspondence $\mathrm{R}_{\mathcal{U}}$ *plus* the proof that such a relation is one-to-one and hence is an equivalence. Then we could *define* the identity type $\mathrm{Id}_{\mathcal{U}}(A, B)$ as $\mathtt{1to1}(A, B)$ and the resulting system should respect the invariance principle. However, it seems that if we want to validate the principle of invariance for universes by means of the proof-relevant relation model, then we have to assume that $\mathcal{U}$ is univalent and this seems to confirm Awodey's claim.

Inverse Diagram Models. As pointed out before, the inverse diagram models presented by Kapulkin and Lumsdaine in their companion papers [71, 72], were a major motivation behind the syntactic model construction given in the previous sections. Then, I think it is useful to include a brief discussion of

[2] In [17] an alternative way to fix this problem is presented by considering Tarski universes. Bocquet shows that it is possible to introduce a Tarski universe without necessarily breaking invariance. I would like to thank one of the anonymous referees for having pointed out this to me.

these kinds of models of type theory. Categories with attributes (**CwA**) are one of the possible categorical notions of model for a dependent type theory.

Definition 3.4.1. A category $\mathcal{C}$ is a **CwA** if it consists of the following data:

- $\mathcal{C}$ has a chosen terminal object $\mathbf{1}$;
- a functor $Ty : \mathcal{C}^{op} \to Set$;
- for each $A \in Ty(\Gamma)$, an object $\Gamma.A \in \mathcal{C}$ with a map $p_A : \Gamma.A \to \Gamma$;
- for each $A \in Ty(\Gamma)$ and $f : \Delta \to \Gamma$, a map $f.A : \Delta.f^*A \to \Gamma.A$ such that the following diagram is a pullback

$$\begin{array}{ccc} \Delta.f^*A & \xrightarrow{f.A} & \Gamma.A \\ {\scriptstyle p_{f^*A}}\downarrow & & \downarrow{\scriptstyle p_A} \\ \Delta & \xrightarrow[f]{} & \Gamma \end{array}$$

This definition corresponds to the structural rules of type theory. The rules for type constructors can be specified axiomatically by equipping $\mathcal{C}$ with axioms corresponding to them. This means that we can equip a **CwA** with a Π-structure, a Σ-structure and so on.
Kapulkin and Lumsdaine show how new models of type theory can be obtained starting from a **CwA** and then taking the diagram category $\mathcal{C}^I$, for I an inverse category. Moreover, if $\mathcal{C}$ is equipped with identity types, dependent products, dependent sums and unit type, then so is $\mathcal{C}^I$. Let us introduce the relation $a \prec b$ over the objects of I with the intended meaning that there exists a non-identity morphism $f : b \to a$. Then I is an inverse category if the relation $\prec$ is well-founded or, equivalently (classically), there are no infinite descending chains of non-identity morphisms in I. An inverse category is *homotopical* if it comes equipped with a distinguished class of morphisms that are the equivalences. Given a diagram $\Gamma : I \to \mathcal{C}$, we can define a Reedy type A over I, that is a diagram $\Gamma.A : I \to \mathcal{C}$ over Γ. The key insight is that such Reedy types correspond to type-theoretic contexts.
Among the examples of inverse categories considered in [71, 72], there is the category Span

$$0 \leftarrow r \rightarrow 1$$

which is an inverse category with ordering $0, 1 \prec r$. Then an object in $\mathcal{C}^{\mathsf{Span}}$ looks as

$$\gamma_0 : \Gamma_0, \gamma_1 : \Gamma_1 \vdash \Gamma_r(\gamma_0, \gamma_1)$$

and in particular a closed type A is given by

$$x_0 : A_0, x_1 : A_1 \vdash A_r(x_0, x_1)\ \mathit{type}$$

which makes evident the connection with the 'relational' model. Moreover, we can define a sub-model of $\mathcal{C}^{\mathsf{Span}}$ called $\mathcal{C}^{\mathsf{eqv}}$, where the two maps of Span are marked as equivalences. Roughly, this means that

$$\Gamma_r, x_0 : A_0 \vdash (x_1 : A_1, x_r : A_r(x_0, x_1))$$
$$\Gamma_r, x_1 : A_1 \vdash (x_0 : A_0, x_r : A_r(x_0, x_1))$$

are both contractible. Since A_r is a correspondence over A_0 and A_1, this corresponds to it being one-to-one and so an equivalence.
Then, the following result is stated in [71]:

Proposition 3.4.2 (Proposition 5.10 [71])**.** *1.* $\boldsymbol{C}^{\mathrm{eqv}}$ *is naturally equipped with* ***Id****-types;*

2. if $\boldsymbol{C}$ *additionally carries* Σ*-types (resp. unit types) then so does* $\boldsymbol{C}^{\mathsf{eqv}}$*;*

3. if $\boldsymbol{C}$ *has* Π*-types and functional extensionality, then so does* $\boldsymbol{C}^{\mathrm{eqv}}$*;*

4. moreover, in all these cases, the maps $P_i : \boldsymbol{C}^{\mathsf{eqv}} \to \boldsymbol{C}$ *preserve such structure.*

Moreover, the authors add a remark stating that a direct construction for the above proposition would amount to show that each constructor preserves equivalences and for this reason function extensionality is assumed. This perfectly matches the assumption of function extensionality in the construction of the refined relational model. Furthermore, the preservation of the logical structure of $\mathbf{C}$ in the case of $\mathbf{C}^{\mathrm{eqv}}$ directly follows from the general case for an arbitrary inverse category I. In the particular case of the relational model, we have seen that the soundness of the relational equivalences model essentially relies on the soundness of the general relational model and is only required to provide the additional data witnessing that such relations are equivalences.
It is worth mentioning that similar constructions of inverse diagram models have been considered by Shulman in [143]. There, he works with *type-theoretic fibration categories* as notion of model for dependent type theory. A

remarkable fact is that Shulman gives also a treatment of universes in inverse diagram models. Basically, he proves that a universe in $\mathbf{C}^I$ is univalent if the 'starting' universe in $\mathbf{C}$ is univalent. Therefore, the construction validates the Univalence axiom under the assumption that the universe in $\mathbf{C}$ is univalent and this fact seems to be in harmony with Awodey's remark discussed before. In light of these remarks, it is tempting to conclude that the relational model construction is the syntactical counterpart of the $\mathbf{C}^{\mathsf{Span}}$ construction considered by Kapulkin and Lumsdaine. However, this conclusion would not be entirely correct. The reason is a mismatch between arbitrary contexts, i.e. objects, of a **CwA** and inductively generated contexts of the type theory concretely presented. The problem is to guarantee that contexts of a **CwA** can be regarded as inductively generated objects in order to match the syntactic presentation. Such a difficulty disappears if *contextual categories* or C-systems are taken as notion of model for dependent types. In these structures, the objects are equipped with a length function and operations that closely mirror the inductive definition of contexts. In particular, there is a unique terminal object $\mathbf{1}$ such that every object of the category can be represented as $\mathbf{1}.A_1.\cdots.A.n$.
In [71] a way is described to build a contextual category from a given **CwA** $\mathcal{C}$ [3]: the fibrant slice $\mathcal{C}//\Gamma$ is the contextual category having as objects context extensions over Γ - that is, $\Gamma.\Delta :\equiv \Gamma.A_0.\cdots.A_{n-1}$-with all the relevant structure inherited from $\mathcal{C}$. In particular, $\mathcal{C}//\mathbf{1}$ is the contextual core of $\mathcal{C}$. With these definitions at hand, the relational model can be regarded as the contextual core of the **CwA** $\mathcal{C}_T^{\mathsf{Span}}$, where $\mathcal{C}_T$ is the syntactic contextual **CwA**. Finally, the work by Kapulkin and Lumsdaine suggests that we can consider more general inverse categories and it would be interesting to get a better understanding of these general cases from the point of view of syntactic models: e.g., it should be possible define infinitary parametricity translations. Also, the relation with the work presented in [25] should be explored.

,

[3] In particular, see Def.4.3 [71].

Chapter 4

MF & the Compatibility with HoTT

In this chapter, I present the two-level system known as *Minimalist Foundation* (**MF**) and review the basic motivations behind its introduction. The bulk of the chapter is devoted to a proof of the *compatibility*, in a technical sense that I will specify below, of **MF** with **HoTT**. **MF** was first introduced in [88] and then fully worked out in [89] and consists of an extensional level called **emTT** built over an intensional level called **mTT**. The two levels are suitably connected and the whole system turns out to be a flexible and conceptually well-grounded foundational framework.

4.1 Motivations for MF

Classical mathematics is usually supposed to be developed within the set-theoretical framework provided by **ZFC**. This theory emerged as the standard setting for classical mathematics during the last century overcoming rival theories and nowadays represents the foundational framework virtually adopted by the majority of mathematicians.
In the case of constructive mathematics, there is no single framework enjoying a status comparable to that of **ZFC**. Several different theories have been proposed as foundations for constructive mathematics, to the extent that many works in constructive mathematics explicitly state within which particular framework the results that they contain must be taken to hold. Moreover, a proof of a certain theorem that makes use of principles available in a particular

framework is not always automatically reproducible in another framework, where those principle might not be valid. Further, the status of single axioms may vary across different contexts: in some cases, a certain axiom may lead to unwanted consequences, whereas in others the same axiom turns out to be perfectly acceptable.

Among the systems proposed as foundational systems for constructive mathematics, there are different versions of Martin-Löf Type Theory, various presentations of the internal theory of an elementary topos, constructive set theories such as **IZF** and **CZF**, Feferman's systems of (constructive) explicit mathematics and possibly other theories as well.

As one may expect, these theories assume different primitive notions and enjoy different properties. This variety of systems can be classified according to distinct criteria. A first relevant distinction is between *intensional* and *extensional* theories. It is usually assumed that mathematics is thoroughly extensional, in the sense that mathematical objects are dealt with without paying attention to how they are presented to us and how they are defined or constructed. It depends on the context how extensionality is formalized: e.g. in set theory there is a well-known axiom of extensionality stating when two sets are equal, while in type theory, as we have seen, the function extensionality axiom declares two (type-theoretic) functions to be equal if they are pointwise equal and the propositional extensionality axiom identifies two propositions which are logically equivalent. Intensional theories, on the other hand, take into account how objects are presented and defined: roughly speaking, even if two objects exhibit the same external behaviour, nevertheless they cannot be automatically identified. An elementary example consists in taking two functions f, g from $\mathbb{N}$ to $\mathbb{N}$ that are defined as follows: $f(n) :\equiv n+3-3$ and $g(n) :\equiv n$. Trivially, they yield always the same value for every argument and their graphs are indistinguishable. On the other hand, the way they are defined and the calculations they prescribe are obviously different. In an extensional framework, this last remark is entirely superfluous, whereas in an intensional system it must be taken into consideration. Also the distinction between intensional theories and extensional ones is sometimes a matter of degree. In this respect, the examples of different versions of type theory are instructive.

Another fundamental criterion to discriminate between foundational frameworks is given by *predicativity*. Roughly speaking, the definition of a certain object x is impredicative if it quantifies over the collection to which x itself belongs. A theory like **ZFC** is thoroughly impredicative (e.g. Power Set

Axiom, Axiom of Separation), but this is in line with the fact that impredicative definitions are part of classical mathematics. However, it is widely acknowledged that, at least philosophically, impredicativity is especially problematic from a constructive point of view: an impredicative definition seems to assume the existence of the object that is being defined. It follows that a definition cannot be read as an 'instruction' to build an object. However, there are examples of impredicative constructive theories: e.g., intuitionistic higher-order logic (as a presentation of the internal theory of a generic elementary topos) or a type theory such as the Calculus of Constructions. On the other hand, many examples are known of constructive theories that are classified as predicative: e.g., different formulations of Martin-Löf Type Theory, some versions of Feferman's systems of explicit mathematics and also set theories such as **CZF**. However, the word 'predicative' must be taken with a grain of salt. As we will get to see, the concept of predicativity can be shaped in different ways and different analyses give rise to distinct notions of predicativity. This being the case, a theory that is predicative according to a certain notion might not be such on the basis of another one.

A general phenomenon that takes place when moving from a classical to an intuitionistic context is the *splitting* of definitions. Notions that are classically equivalent cease to be such when intuitionistic logic is adopted. In particular, it becomes interesting to formulate *positive* versions of such notions. An elementary example is the difference between a set being *inhabited* and a set being *non-empty*. Intuitionistically, the former notion is stronger than the latter, since it is its positive counterpart. Another example is provided by the notion *apartness* between elements of a given set, that is the positive version of the notion asserting that two elements are not equal. This phenomenon depends on how some logical constants are interpreted and thus reduces to a matter of logic. Most importantly, it highlights the fact that in a constructive setting we can make more fine-grained distinctions between mathematical concepts in accordance with Bishop's principle for which 'meaningful distinctions deserve to be preserved'.

Intuitionistic logic is generally assumed to be equipped with an informal semantics in terms of proofs, namely BHK. However, due to the informal nature of this semantics, there have been different attempts to give a more precise characterization of it: most notably, Kleene-style realizability, functional interpretations such Gödel's *Dialectica* and the Curry-Howard isomorphism. Also we have seen that the correspondence between propositions and types can be framed in different ways that directly affect the status of principles

with foundational and mathematical interest such as the Axiom of Choice. We can argue over what is the most faithful formal counterpart of BHK, but almost every attempt agrees on the fact that intuitionistic logic has a natural *computational* flavour and thus an intuitionistic proof can be regarded as a *program*. Under this perspective, it is quite natural to take intensional aspects seriously, since programs are generally understood as intensional objects.
An intensional framework is naturally computationally-oriented and usually supports a native notion of computation: proofs are endowed with computational content and, generally, algorithms can be extracted from them that can be implemented on a computer. On the other hand, developing mathematics in an intensional context might turn out to be quite cumbersome and alien to the way of thinking of the working mathematician. In this respect, an extensional framework seems to be best suited for mathematical practice. Therefore, the main problem is how to harmonize these two fundamental aspects within a single framework.
Given this picture, it is natural to ask for a framework **T** that can serve as common core between the different foundational theories existing in the literature and as basis to compare the principles available in them. All these theories should be thought of as extensions of the common core **T**. Furthermore, a natural assumption would be to ask that **T** be compatible with the other theories, where by *compatibility* the following is meant:

> A theory **T** is said to be *compatible* with another theory **S** iff there exists a translation from **T** to **S** such that a proof of a theorem in **T** can be translated into a proof of the same theorem in **S** and the meaning of logical and set-theoretical constructors is preserved.

Then, compatibility is set up as requirement stronger than the condition that the translation preserves derivability. In particular, if **T** is our common core theory, then a theorem that has been established in this framework can be automatically reproduced or transported into another compatible theory with essentially the same proof. A proof given in this system will make use of a minimal set of principles that are available also in the systems that turn out to be compatible with it. In light of this, individuating a theory that may serve as common core paves the way for a kind of (constructive) reverse mathematics.
The previous discussion leads to the individuation of a set of conditions that a theory **T** should simultaneously satisfy to be considered a suitable common core. These conditions are the following:

1. It must be based on intuitionistic logic and hence it must be consistent with AC, since it is informally validated by BHK.

2. It must be predicative: every object must be inductively defined and presented by a finite number of rules. Predicativity is also crucial to ensure compatibility with other systems.

3. The admissible operations must turn out to be effective, in the sense that they preserve computational content. More formally, consistency with formal Church's Thesis is required and also the existence of an interpretation that allows program-extraction. Church's thesis is the principle asserting that every total number-theoretic function is internally computable. In a type-theoretic language, it is stated as follows:

$$\forall f \in \mathbb{N} \to \mathbb{N}\ \exists e \in \mathbb{N}\ (\forall x \in \mathbb{N}\ \exists y \in \mathbb{N}\ T(e,x,y)\ \wedge\ U(y) =_{\mathbb{N}} f(x))$$

   where $T(e,x,y)$ is the *Kleene's T predicate* with the intended meaning that 'y is the code for the computation executed by a computable function with index e applied to the input x' and $U(y)$ stands for 'the result of the computation encoded by y'.

4. The computational properties of intensional systems should be preserved. At the same time, useful constructions (e.g. quotients) that are available in extensional frameworks should be admissible, since they provide familiar tools for doing mathematics in the usual way.

5. It must be consistent with the addition of classical principles. This condition is in the spirit of Bishop's approach to constructive mathematics, where this is regarded as a kind of neutral and general mathematics. Moreover, also classical theories can be seen as extensions of the basic framework.

6. It must be compatible with the main existing foundational frameworks.

There are many well-known results in the literature that prevent the most relevant existing foundational systems from satisfying all these conditions taken together. For instance, it is known [157] that Heyting arithmetic in all finite types, HA^ω, if extended with Church's Thesis, the Axiom of Choice and function extensionality, is inconsistent. Then, the same result

extends to the extensional version of Martin-Löf Type Theory. Further, it is also known that CZF extended with AC allows to derive LEM (Goodman-Myhill Theorem [111]). The same theorem holds in the context of elementary toposes (Diaconescu Theorem [38]). These results point out that choice in extensional settings has necessarily classical consequences and thus choice and extensionality are constructively incompatible. A slightly different example is provided by the fact that constructive and predicative theories such as CZF and Martin-Löf Type Theory do not enjoy stability w.r.t. predicativity if classical principles are assumed. That is, even if they are consistent with classical logic, the resulting classical extensions are impredicative [1, 103]. Finally, also a system like **HoTT** that combines extensional and intensional features and satisfies many of the above conditions fails to be an adequate common core: e.g., Univalence Axiom is known to destroy the computational properties of the system, for instance canonicity, since it allows to get terms that cannot be reduced to canonical form. Therefore, the design of a system that simultaneously satisfy all the conditions I have listed above turns out to be a very delicate matter.
A formal system that has been explicitly introduced to meet all these requirements simultaneously is the system **MF**, which I am going to present in the next section.

4.2 The system MF

The Minimalist Foundation (**MF**) is a two-level formal system. The presence of two levels is meant to accommodate the benefits of both intensional and extensional systems. Indeed, the first level **mTT** is a version of intensional intuitionistic type theory with a primitive type of propositions; the second level **emTT**, instead, is a kind of extensional type theory that could be also described as a predicative *local* set theory in the sense of [14]. Of course, the two levels are not unrelated and this makes it possible to consider **MF** as a single foundational framework. Roughly speaking, the extensional sets arise as quotients of the corresponding intensional sets 'forgetting' some computational information; however, this information can be restored by means of a quotient model construction interpreting **emTT** into **mTT** and thus providing a justification for the extensional constructs introduced. More abstractly, the setoid model can be described as an instance of the quotient completion categorical construction in the sense of [91]. Therefore, **emTT**

appears to be a fragment of the quotient completion of the intensional level **mTT**.
Finally, a 'third level' is provided by a Kleene realizability interpretation of the intensional level that allows the extraction of programs from proofs and thus confirms the effective character of all the constructions that can be actually performed in **mTT**. However, it must be stressed that the actual foundation **MF** consists only of the two levels **mTT** and **emTT**, and thus speaking of a third level must not be understood literally.
In the following, I will give a brief description of the syntax of the system, stressing the most important aspects. A detailed presentation of the rules can be found in the Appendix to [89].
Notation. I will stick to the notation adopted in [89]: the symbol '$=$' refers to judgemental equality and '$\in$' to typing judgements. Intensional propositional equality will be denoted by 'Id', while extensional propositional equality by 'Eq'. Usually, eliminators are denoted by 'el' with a subscript referring to the relevant type constructor. These choices allows us to distinguish **MF**-symbols from **HoTT**-symbols for analogous notions when I will present an interpretation of the former system into the latter.
The calculus **mTT** includes four primitive sorts: *collections*, *sets*, *propositions* and *small propositions*. The distinction between sets and collections can be understood by analogy with the distinction between sets and classes in some axiomatizations of set theory. In this setting, sets can be regarded as those collections that are inductively generated. The introduction of the distinction is also motivated by the practice of constructive mathematics (e.g. topology), where in some cases objects that cannot be inductively generated must be considered. Similarly, small propositions are those propositions that are closed under quantification over sets. Further, **mTT** consists of the following basic forms of judgements in the style of Martin-Löf Type Theory:

- A *type* $[\Gamma]$;
- $A = B$ *type* $[\Gamma]$;
- $a \in A$ $[\Gamma]$;
- $a = b \in A$ $[\Gamma]$.

where '*type*' is assumed to be a meta-variable ranging over the four primitive sorts. The informal explanation given above together with the fact that a

proposition (resp. small proposition) can be thought of as the collection (resp. set) of its proofs provides a justification for the following rules:

$$\frac{A\ set\ [\Gamma]}{A\ col\ [\Gamma]}\ (\text{S-in-C}) \qquad \frac{A\ prop_s\ [\Gamma]}{A\ prop\ [\Gamma]}\ (\text{sP-in-P})$$

$$\frac{A\ prop\ [\Gamma]}{A\ col\ [\Gamma]}\ (\text{P-in-C}) \qquad \frac{A\ prop_s\ [\Gamma]}{A\ set\ [\Gamma]}\ (\text{sP-in-S})$$

The rules S-in-C and sP-in-P are not mere inclusions, since they have direct consequences on the way the elimination rules for sets and small propositions are formulated: indeed, the constructors of these sorts eliminate towards collections and propositions respectively. On the other hand, it is a crucial fact that propositions cannot eliminate towards sets and collections, since in this way the derivation of the type-theoretic AC is blocked [1].
The set-constructors of **mTT** comprehend $\mathsf{N}_0, \mathsf{N}_1, \mathsf{List}, +, \Sigma$ and Π, whereas the logical constructors are $\bot, \wedge, \vee, \rightarrow, \exists, \forall$ and Id. Further, a collection of small propositions is included and is denoted by $\mathsf{prop_s}$. And collections are closed under functions from sets to $\mathsf{prop_s}$[2].
The rules for context formation and the structural rules are the standard ones. Technically, a significant modification is that term congruence rules are replaced by the following explicit substitution rules for terms:

$$\frac{\begin{array}{c} c(x_1,\ldots,x_n) \in C(x_1,\ldots,x_n)\ [x_1 \in A_1,\ldots,x_n \in A_n(x_1,\ldots,x_{n-1})] \\ a_1 = b_1 \in A_1\ \cdots\ a_n = b_n \in A_n(a_1,\ldots,a_{n-1}) \end{array}}{c(a_1,\ldots,a_n) = c(b_1,\ldots,b_n) \in C(a_1,\ldots,a_n)}\ (\text{Sub})$$

As a consequence, the ξ-rule is left out. This point must be considered in relation with the realizability interpretation of type theory. It was Martin-Löf who observed in [105] that the ξ-rule poses serious problems in the context of a realizability model. In general, it remains an open problem to show

[1] Even the weaker AC! is not derivable in **mTT** [86].

[2] Actually, if the closure of collections under dependent products is assumed, this follows as a special case.

the consistency of the intensional theory including the ξ-rule with Church's Thesis, while it has been proved in [70] that this theory with the ξ-rule replaced by the explicit substitution rule (Sub) displayed above is consistent with Church's Thesis and therefore the same holds for **mTT**.
The extensional level **emTT** has the same primitive sorts and judgements as **mTT**. The congruence rules for type and term constructors are added and hence also the ξ-rule. These rules express the fact that all the constructors of **emTT** respect judgemental equality. Further, the η-rule for the dependent product is also included.
The theory is extensional in the strong sense that propositional equality satisfies the reflection rule. The logical and set-theoretical constructors of **emTT** are those of **mTT** with the significant difference that **emTT**-sets are closed also under *effective quotients*. The possibility of including quotient sets without deriving classical principles is essentially due to the fact that propositions are regarded as *monotypes*, i.e. they are inhabited by at most one canonical element and satisfy *definitional proof-irrelevance*:

$$\frac{A\ prop\ [\Gamma] \qquad p \in A\ [\Gamma] \qquad q \in A\ [\Gamma]}{p = q \in A\ [\Gamma]}\ \text{(P-mono)}$$

$$\frac{A\ prop\ [\Gamma] \qquad p \in A\ [\Gamma]}{\mathtt{true} \in A\ [\Gamma]}\ \text{(P-true)}$$

Therefore, every two proofs of a proposition are *judgementally* equal and the unique canonical proof-term is denoted by 'true'. In this sense, we are forgetting all the information encoded in the proof-terms of propositions that now are considered only at the level of provability.
Another fundamental difference with **mTT** is that **emTT** includes the quotient of the collection of small propositions under the relation of equiprovability instead of the intensional collection of small propositions. This quotient amounts to the power collection of the singleton:

$$\mathcal{P}(1)\ col\ [\Gamma]\ (\mathcal{P}(1)\text{-F}) \qquad \frac{A\ s\text{-}prop\ [\Gamma]}{[A] \in \mathcal{P}(1)\ [\Gamma]}\ (\mathcal{P}(1)\text{-I})$$

$$\frac{\mathsf{true} \in A \leftrightarrow B \ [\Gamma]}{[A] = [B] \in \mathcal{P}(1) \ [\Gamma]} \ (\mathcal{P}(1)\text{-eq})$$

Furthermore, effectiveness holds for $\mathcal{P}(1)$:

$$\frac{[A] = [B] \in \mathcal{P}(1) \ [\Gamma]}{\mathsf{true} \in A \leftrightarrow B \ [\Gamma]} \ (\mathcal{P}(1)\text{-eff})$$

Finally, $\mathcal{P}(1)$ is assumed to satisfy a kind of uniqueness principle:

$$\frac{A \in \mathcal{P}(1) \ [\Gamma]}{A = [\mathsf{Eq}_{\mathcal{P}(1)}(A, [tt])] \ [\Gamma]} \ (\mathcal{P}(1)\text{-}\eta)$$

where $tt \ :\equiv \ \bot \to \bot$ stands for the truth constant. Collections are also closed under functions from sets to $\mathcal{P}(1)$ and thus the following notation can be introduced $\mathcal{P}(A) \ :\equiv \ A \to \mathcal{P}(1)$. In fact, the availability of effective quotient sets as well as of the extensional power collection of the singleton makes **emTT** a quite natural setting for developing mathematics in a familiar manner. Basic set-theoretical notions can be encoded in **emTT**: let A be a set and $a \in A$ and $S \in \mathcal{P}(A)$. Then we can introduce the following notation

$$a \ \varepsilon \ S \ :\equiv \ \mathsf{Eq}_{\mathcal{P}(1)}(S(a), [tt]) \ .$$

Moreover, let $B(x)$ be a small proposition indexed over a set A, then we can introduce the following notation for subsets

$$\{x \in A \mid B(x)\} \in \mathcal{P}(A) \ :\equiv \ \lambda x.[B(x)] \in \mathcal{P}(A) \ .$$

With these definitions at hand, the following basic facts can be established:

Lemma 4.2.1. *The equality between subsets is extensional: the judgement*

$$\{x \in A \mid B(x)\} = \{x \in A \mid C(x)\}$$

is derivable in **emTT** *iff the following judgement is derivable as well:*

$$\forall (x \in A) B(x) \leftrightarrow C(x)$$

Proof. Let us assume $\{x \in A \mid B(x)\} = \{x \in A \mid C(x)\}$. Then we get $\lambda x.[B(x)] = \lambda x.[C(x)] \in \mathcal{P}(A)$. Let $y \in A$, then $[B(y)] = [C(y)] \in \mathcal{P}(1)$ and, hence by $\mathcal{P}(1)$-eff, $\mathsf{true} \in B(y) \leftrightarrow C(y)$. Therefore, $\mathsf{true} \in \forall(x \in A)B(x) \leftrightarrow C(x)$.
On the other hand, let us assume $\forall(x \in A)B(x) \leftrightarrow C(x)$. Then by $\mathcal{P}(1)$-eq, $[B(x)] = [C(x)] \in \mathcal{P}(1)$ and thus $\lambda x.[B(x)] = \lambda x.[C(x)] \in \mathcal{P}(A)$, that is $\{x \in A \mid B(x)\} = \{x \in A \mid C(x)\}$. □

Lemma 4.2.2. *The following comprehension principle holds: the judgement*

$$a \; \varepsilon \; \{x \in A \mid B(x)\} \leftrightarrow B(a)$$

is derivable in **emTT**.

Proof. Let us assume $a \; \varepsilon \; \{x \in A \mid B(x)\}$. Then by unfolding the definition, we get $\mathsf{Eq}_{\mathcal{P}(1)}(B(a), [tt])$ and therefore $B(a) = [tt]$. Thus, by $\mathcal{P}(1)$-eff, $\mathsf{true} \in B(a) \leftrightarrow tt$ and thus $B(a)$ holds.
Conversely, let us assume $B(a)$. Then by $\mathcal{P}(1)$-η, $\mathsf{Eq}_{\mathcal{P}(1)}(B(a), [tt])$, that is $a \; \varepsilon \; \{x \in A \mid B(x)\}$. □

Lemma 4.2.3. *The following uniqueness principle for subsets holds: the judgement*

$$S = \{x \in A \mid x \; \varepsilon \; S\} \in \mathcal{P}(A)$$

is derivable in **emTT**.

Proof. $S = \lambda x.S(x) = \lambda x.[\mathsf{Eq}_{\mathcal{P}(1)}(S(x), [tt])] = \{x \in A \mid x \; \varepsilon \; S\}$. □

These elementary facts provide further evidence for the claim that **emTT** can be thought as a many-sorted set theory.
If we collect what we have said so far about **MF**, then it is clear that this system satisfies all the conditions listed above for a theory be assumed as common core. The system is based on intuitionistic logic and is consistent with AC and Church's Thesis as it has been established by a Kleene realizability interpetration. Extensional constructs are available at the extensional level that is built on the top of **mTT** which, instead, enjoys good computational properties as normalization, canonicity and decidability of type checking. Further, the theory is predicative and is consistent with classical logic, since it can been thought of as a subsystem of Martin-Löf Type Theory. From the point of view of the general program outlined above, the last point that must be checked is the compatibility with the other foundational theories. For

example, the compatibility of **mTT** with intensional Martin-Löf Type Theory follows from the identification of logical and set-theoretical constructors and then by interpreting **mTT**-sets as types in the first universe $\mathcal{U}_0$ and **mTT**-collections as types living outside $\mathcal{U}_0$, that in turn interprets the collection of small propositions. Another example is provided by the interpretation of **emTT** into the internal language of a generic elementary topos that can be presented as an extensional dependent type theory as done in [88]. Another interesting case is the compatibility of **emTT** with constructive set theories that has been investigated in [106].
In the light of the recent developments, investigating the compatibility with **HoTT** has become a pressing problem, since this framework is very popular and represents an exciting new perspective on the foundations of mathematics. Also, it must be stressed that all the examples of compatibility proofs given so far concern one level of **MF** at a time: **mTT** is compared with intensional theories, **emTT**, instead, with extensional ones. A striking feature of **HoTT** is that it can interpret both the levels of **MF** simultaneously and thus it is the only foundational framework, as far as I know, to possess this property.

4.3 The compatibility of MF with HoTT

As we have seen in Chapter 2, **HoTT** is characterized by the presence of the Univalence Axiom that implies various extensional principles within a general intensional setting. The problem of providing a computational interpretation of this axiom has led to the birth of cubical type theory [24], for which a normalization result has been established recently [145]. In what follows, I will show that both levels of **MF** are *compatible* with **HoTT**. The result crucially relies on the extensionality principles available in **HoTT** and hence on Univalence. Another fundamental ingredient is provided by the presence of (effective) set quotients that are an instance of higher inductive types. Moreover, the interpretation preserves the distinction between **MF**-sorts: propositions and small propositions are interpreted as h-propositions in **HoTT** and sets and collections are interpreted as h-sets in **HoTT**. A similar result cannot be achieved in plain **MLTT**, because there propositions would be identified with types. The main difficulty will be to define a correct translation of the extensional level **emTT**. The problem is solved by defining a suitable class of isomorphisms called canonical and interpreting judgemental equality as existence of a canonical isomorphism. Further, these isomorphisms

enjoy a set of closure properties which are exactly what is needed to define a sound translation of **emTT** in **HoTT**. It must be noted that the technique is based on the interpretation of **emTT** into **mTT** designed in [89]. A significant difference is that Univalence and effective set quotients allow us to avoid any setoid model construction. Finally, I will discuss the relationship with some related works. Canonical isomorphisms made an early appearance in [65], where a non-constructive conservativity proof of extensional type theory over intensional type theory extended with function extensionality and uniqueness of identity proofs axioms is provided. The work [114] is an attempt to make Hofmann's interpretation effective: canonical isomorphisms are replaced by a heterogeneous equality and the translation is defined as a *multifunction*. The translation defined by Oury has been improved in [171]. As we will see, the translation of **emTT** into **HoTT** is also multi-functional. This section is the result of collaboration with Milly Maietti.

4.3.1 Preliminaries

The following subsection is meant to recall some basic definitions and facts about **HoTT** that will turn out to be useful later. In what follows, by **HoTT** a fragment of the theory presented in [155] is denoted which includes the following constructors:

$$\mathbf{0} \mid \mathbf{1} \mid \mathsf{List} \mid + \mid \mathrm{Id} \mid \Sigma \mid \Pi \mid \mid\mid A \mid\mid \mid A/R \mid \mathcal{U}_i$$

where $\mid\mid A \mid\mid$ is the propositional truncation of a type A and A/R is the quotient of the *h-set* A under the equivalence relation R. Moreover, we assume two univalent universes $\mathcal{U}_0$ and $\mathcal{U}_1$ such that $\mathcal{U}_0 : \mathcal{U}_1$.
In what follows, I choose some proof-terms for the statements expressing properties of h-sets and h-propositions which will be employed later.

Lemma 4.3.1. *If A is an h-set, then* $\mathrm{Id_A}$ *is an h-proposition: there exists a proof-term*

$$\mathfrak{p}_{Id} : \Pi_{A:\mathcal{U}_i}\ \Pi_{s:\mathsf{isSet}(A)}\ \Pi_{a,b:A}\ \mathsf{isProp}(\mathrm{Id}_A(a,b))$$

Theorem 4.3.2 ([155] Thm.7.1.7)**.** *If A is a* n-type*, then it is also a* $\mathsf{n}+1$-type

Corollary 4.3.3. *Every h-proposition is an h-set: i.e. there exists a proof-term*

$$\mathfrak{s}_{coe} : \Pi_{A:\mathcal{U}_i}\ \mathsf{isProp}(A) \to \mathsf{isSet}(A).$$

Lemma 4.3.4. *The empty type* $\mathbf{0}$ *and the unit type* $\mathbf{1}$ *are h-propositions. The propositional truncation of a type* A *is a h-proposition. Further, h-propositions are closed under* $\times$ *and* Π *(and thus also* $\to$*). Therefore, the following proof-terms exist*

$$\mathfrak{p}_1 : \mathsf{isProp}(\boldsymbol{1}) \qquad \mathfrak{p}_0 : \mathsf{isProp}(\boldsymbol{0})$$
$$\mathfrak{p}_{\to} : \Pi_{A,B:\mathcal{U}_i}\ \Pi_{q:\mathsf{isProp}(B)}\ \mathsf{isProp}(A \to B)$$
$$\mathfrak{p}_{\times} : \Pi_{A,B:\mathcal{U}_i}\ \Pi_{p:\mathsf{isProp}(A),q:\mathsf{isProp}(B)}\ \mathsf{isProp}(A \times B)$$
$$\mathfrak{p}_{\Pi} : \Pi_{A:\mathcal{U}_i}\ \Pi_{B:A\to\mathcal{U}_i}\ \Pi_{p:\Pi_{x:A}\mathsf{isProp}(B(x))}\mathsf{isProp}(\Pi_{x:A}B(x))$$
$$\mathfrak{p}_{||\ ||} : \Pi_{A:\mathcal{U}_i}\mathsf{isProp}(||A||)$$

Since h-propositions are not closed under Σ and $+$, we need to apply propositional truncation to define disjunction and existential quantification: $P \vee Q$ is identified with $||P + Q||$ and $\exists_{x:A}\ P(x)$ with $||\Sigma_{x:A}\ P(x)||$.

Definition 4.3.5. The disjunction of h-propositions P and Q is defined as

$$P \vee Q :\equiv ||P + Q||$$

Its canonical constructors are in turn defined as follows : given $p : P$ and $q : Q$

$$\mathsf{inl}_{\vee}(p) :\equiv |\mathsf{inl}(p)| : P \vee Q \qquad \mathsf{inr}_{\vee}(q) :\equiv |\mathsf{inr}(q)| : P \vee Q$$

and its eliminator is defined as follows: for any C such that $s : \mathsf{isProp}(C)$, any $e : P \vee Q$ and any $d_1(x) : C\ [x : P]$ and $d_2(y) : C\ [y : Q]$

$$\mathsf{ind}_{\vee}(e, x.d_1(x),\ y.d_2(y), s\ ) :\equiv \mathsf{ind}_{||\ ||}(e\,,\ z.\mathsf{ind}_{+}(z,\ x.d_1(x)\,,\ y.d_2(y)\ ), s\ ) : C$$

Lemma 4.3.6. *The disjunction defined in def. 4.3.5 satisfies the following* β *definitional equalities: for any* C *such that* $s : \mathsf{isProp}(C)$*, any* $p : P$ *and* $q : Q$*, and any* $d_1(x) : C\ [x : P]$ *and* $d_2(y) : C\ [y : Q]$

$$\mathsf{ind}_{\vee}(\mathsf{inl}_{\vee}(p), x.d_1(x)\,, y.d_2(y), s) \equiv d_1(p) : C$$

$$\mathsf{ind}_{\vee}(\mathsf{inr}_{\vee}(q), x.d_1(x)\,, y.d_2(y), s) \equiv d_2(q) : C$$

Definition 4.3.7. For any h-set A and any family of h-propositions $P(x)$ $[x : A]$, the existential quantification is defined as

$$\exists_{x:A}\ P(x)\ :\equiv ||\Sigma_{x:A}\ P(x)||$$

Its canonical constructor is defined as follows: for $a : A$ and $p : P(a)$

$$(a,_{\exists}\, p) :\equiv |(a, p)| : \exists_{x:A}\ P(x)$$

and its eliminator is defined as follows: for any C such that $s : \mathsf{isProp}(C)$, any $e : \exists_{x:A}\ P(x)$ and any $d(x, y) : C$ $[x : A, y : P(x)]$:

$$\mathsf{ind}_{\exists}(e,\ x.y.d(x, y), s\ ) :\equiv \mathsf{ind}_{||\ ||}(e\ ,\ z.\mathsf{ind}_{\Sigma}(z,\ x.y.d(x, y)), s\ ) : C$$

Lemma 4.3.8. *The existential quantifier defined in def. 4.3.7 satisfies the following β definitional equality: for any C such that $s : \mathsf{isProp}(C)$, any $a : A$ and $p : P(a)$ and any $q : Q$ and $d(x, y) : C$ $[x : A, y : P(x)]$*

$$\mathsf{ind}_{\exists}((a,_{\exists}\, p),\ x.y.d(x, y), s\ ) \equiv d(a, p) : C$$

The fact that $\vee$ and $\exists$ are h-propositions is encoded by means of the following proof-terms:

$$\mathfrak{p}_{\vee}\ :\equiv\ \lambda A, B.\ \mathfrak{p}_{||\ ||}(A + B) : \Pi_{A,B:\mathcal{U}_i}\ \mathsf{isProp}(A \vee B)$$
$$\mathfrak{p}_{\exists}\ :\equiv\ \lambda A, B.\ \mathfrak{p}_{||\ ||}(\Sigma_{x:A}\ B(x)) : \Pi_{A:\mathcal{U}_i}\ \Pi_{B:A\to\mathcal{U}_i}\ \mathsf{isProp}(\exists_{x:A}B(x))$$

Definition 4.3.9. We can define the logic of h-propositions as follows:

$$\begin{array}{lcl}
\top & :\equiv & \mathbf{1} \\
\bot & :\equiv & \mathbf{0} \\
P \wedge Q & :\equiv & P \times Q \\
P \supset Q & :\equiv & P \to Q \\
P \vee Q & :\equiv & ||P + Q|| \\
\exists_{x:A}\ P(x) & :\equiv & ||\Sigma_{x:A}\ P(x)|| \\
\forall_{x:A}\ P(x) & :\equiv & \Pi_{x:A}\ P(x)
\end{array}$$

For the purposes of the interpretation, I introduce also a different representation of the logic of h-propositions, where *all* logical constants are taken to be truncated. It is mainly a technical trick that will turn out to be useful when we will consider the interpretation of **emTT**. The next lemma shows that it is possible to proceed this way:

Lemma 4.3.10. *If $P : \mathcal{U}_i$ and $s : \mathsf{isProp}(P)$, then $|-| : P \to ||P||$ is an equivalence, i.e. there is an inverse $|-|^{-1} : ||P|| \to P$ which satisfies $|-| \circ |-|^{-1} =_{||P||} \mathsf{id}_{||P||}$ and $|-|^{-1} \circ |-| =_P \mathsf{id}_P$. Therefore $P =_{\mathcal{U}_i} ||P||$ holds.*

Proof. We can simply define $|z|^{-1} :\equiv \mathsf{ind}_{||\ ||}(z, (x).x, s)$ since P is a h-proposition. Note that for any $z : ||P||$, the equality $|(|z|^{-1})| =_{||P||} z$ is only propositionally valid, while $|(|p|)|^{-1} \equiv p : P$ holds for any $p : P$. The rest follows from Univalence and from P being a h-proposition. □

Definition 4.3.11. The alternative definition of h-logic consists of the following clauses:

$$\begin{array}{lcl}
\top & :\equiv & ||\ \mathbf{1}\ || \\
\bot & :\equiv & ||\ \mathbf{0}\ || \\
P \wedge Q & :\equiv & ||P \times Q|| \\
P \supset Q & :\equiv & ||P \to Q|| \\
P \vee Q & :\equiv & ||P + Q|| \\
\exists_{x:A}\ P(x) & :\equiv & ||\Sigma_{x:A}\ P(x)|| \\
\forall_{x:A}\ P(x) & :\equiv & ||\Pi_{x:A}\ P(x)||
\end{array}$$

Accordingly, we can add the following proof-terms to the previously defined $\mathfrak{p}_\vee$ and $\mathfrak{p}_\exists$:

$$\begin{array}{lcl}
\mathfrak{p}_{||\mathbf{1}||} & :\equiv & \mathfrak{p}_{||\ ||}\ (\mathbf{1}) : \mathsf{isProp}(||\ \mathbf{1}\ ||) \\
\mathfrak{p}_{||\mathbf{0}||} & :\equiv & \mathfrak{p}_{||\ ||}\ (\mathbf{0}) : \mathsf{isProp}(||\ \mathbf{0}\ ||) \\
\mathfrak{p}_{||\times||} & :\equiv & \lambda A, B.\ \mathfrak{p}_{||\ ||}(A \times B) : \Pi_{A,B:\mathcal{U}_i} \mathsf{isProp}(||A \times B||) \\
\mathfrak{p}_{||\to||} & :\equiv & \lambda A, B.\ \mathfrak{p}_{||\ ||}(A \to B) : \Pi_{A,B:\mathcal{U}_i} \mathsf{isProp}(||A \to B||) \\
\mathfrak{p}_{||\Pi||} & :\equiv & \lambda A, B.\ \mathfrak{p}_{||\ ||}(\Pi_{x:A} B(x)) : \Pi_{A:\mathcal{U}_i} \Pi_{B:A\to\mathcal{U}_i} \mathsf{isProp}(||\Pi_{x:A} B(x)||)
\end{array}$$

Furthermore, the constructors for the conjunction, the implication and the universal quantifier are given in the following definitions and we can show that they satisfy the associated β-equalities:

Definition 4.3.12. Given $a : A$ and $b : B$ and $c : ||A \times B||$ and given $s : \mathsf{isProp}(A)$ and $t : \mathsf{isProp}(B)$, we define

$$(a,_\wedge b) \;:\equiv\; |(a,b)| : ||A \times B|| \qquad \begin{array}{l} \mathsf{pr}_{1\,\wedge}(c) \;:\equiv\; \mathsf{ind}_{||\ ||}(\, c, z.\mathsf{pr}_1(z), s\ ) \\ \mathsf{pr}_{2\,\wedge}(c) \;:\equiv\; \mathsf{ind}_{||\ ||}(\, c, z.\mathsf{pr}_2(z), t\ ) \end{array}$$

Definition 4.3.13. Given $a : A$ and $b : B\ [x : A]$ and $c : ||A \to B||$ and $s : \mathsf{isProp}(B)$, we define

$$\lambda_\to x.b \;:\equiv\; |\lambda x.b| : ||A \to B|| \qquad c_\to(a) :\equiv\; \mathsf{ind}_{||\ ||}(\, c, x.x(a), s\ )$$

Definition 4.3.14. Given $a : A$ and $b : B(x)\ [x : A]$ and $c : ||\Pi_{x:A}\ B(x)||$ and $s : \mathsf{isProp}(B(a))$, we define

$$\lambda_\forall x.b \;:\equiv\; |\lambda x.b| : ||\Pi_{x:A} B(x)|| \qquad c_\forall(a) :\equiv\; \mathsf{ind}_{||\ ||}(\, c, x.x(a), s\ )$$

Lemma 4.3.15. *The usual β-definitional equalities for the projections of conjunctions in Def.4.3.12*

$$\mathsf{pr}_{1\,\wedge}((a,_\wedge b)\ ) \;\equiv\; a \qquad\qquad \mathsf{pr}_{2\,\wedge}((a,_\wedge b)\ ) \;\equiv\; b$$

for functions of implications in Def.4.3.13 and universal quantifiers in Def.4.3.14

$$(\lambda_\to x.b)_\to(a) \;\equiv\; b[a/x] \qquad\qquad (\lambda_\forall x.b)_\forall(a) \;\equiv\; b[a/x]$$

are valid in **HoTT**.

Proof. They follow by the elimination rule of the propositional truncation and the usual β-definitional equalities for the corresponding types under truncation. □

We will crucially use the fact that h-sets are closed under the following type constructors:

Lemma 4.3.16. *H-sets are closed under Π (and hence $\to$), Σ (and hence $\times$), $+$ and* List. *Furthermore, for any h-set A and any equivalence relation R defined as an h-proposition, then the quotient type A/R is an h-set. Therefore, the following proof-terms exist:*

$$\mathfrak{s}_1 : \mathsf{isSet}(\mathbf{1}) \qquad \mathfrak{s}_0 : \mathsf{isSet}(\mathbf{0}) \qquad \mathfrak{s}_{\mathbb{N}} : \mathsf{isSet}(\mathbb{N})$$

$$\mathfrak{s}_\Pi : \Pi_{A:\mathcal{U}_i}\ \Pi_{B:A\to\mathcal{U}_i}\ \Pi_{s:\Pi_{x:A}\mathsf{isSet}(B(x))}\mathsf{isSet}(\Pi_{x:A}B(x))$$

$$\mathfrak{s}_\Sigma : \Pi_{A:\mathcal{U}_i}\ \Pi_{B:A\to\mathcal{U}_i}\ \Pi_{s:\mathsf{isSet}(A)}\ \Pi_{t:\Pi_{x:A}\mathsf{isSet}(B(x))}\mathsf{isSet}(\Sigma_{x:A}B(x))$$

$$\mathfrak{s}_+ : \Pi_{A,B:\mathcal{U}_i}\ \Pi_{s:\mathsf{isSet}(A)}\ \Pi_{t:\mathsf{isSet}(B)}\ \mathsf{isSet}(A+B)$$

$$\mathfrak{s}_{\mathsf{List}} : \Pi_{A:\mathcal{U}_i}\Pi_{s:\mathsf{isSet}(A)}\ \mathsf{isSet}(\mathsf{List}(A))$$

$$\mathfrak{s}_Q : \Pi_{A:\mathcal{U}_i}\ \Pi_{R:A\to A\to\mathcal{U}_i}\ \Pi_{s:\mathsf{isSet}(A)}\ \Pi_{p:\mathsf{isProp}(R)}\ \Pi_{r:\mathrm{equiv}(R)}\ \mathsf{isSet}(A/R)$$

where $\mathrm{equiv}(R)$ *is an abbreviation for the fact that* R *is an equivalence relation.*

For any 'external' natural number i, the type of h-sets within the universe $\mathcal{U}_i$ is defined as follows

$$\mathsf{Set}_{\mathcal{U}_i} :\equiv \Sigma_{(X:\mathcal{U}_i)}\ \mathsf{IsSet}(X)$$

H-sets are also closed under a sub-universe classifier $\mathsf{Prop}_{\mathcal{U}_0}$ including all the h-propositions in $\mathcal{U}_0$:

$$\mathsf{Prop}_{\mathcal{U}_0} := \Sigma_{(X:\mathcal{U}_0)}\ \mathsf{isProp}(X)$$

Lemma 4.3.17. $\mathsf{Prop}_{\mathcal{U}_0}$ *is an h-set.*

The proof-term inhabiting $\mathsf{isSet}(\mathsf{Prop}_{\mathcal{U}_0})$ is denoted by $\mathfrak{s}_{\mathsf{Prop}_0}$.

Remark 4.3.18. However, $\mathsf{Prop}_{\mathcal{U}_0}$ is not 'small', since it is not a type in $\mathcal{U}_0$, but lives in a higher universe (see section 10.1 in [155]).

Finally, I recall the following crucial property of set quotients: let A be a h-set and R an equivalence relation over A, then the quotient A/R is *effective*:

$$[a_1] =_{A/R} [a_2] \ \leftrightarrow\ R(a_1, a_2).$$

and we can consistently assume that if $A : \mathcal{U}_i$ and $R(x, y) : \mathcal{U}_i$ $[x, y : A]$, then $A/R : \mathcal{U}_i$, i.e. the quotient lives in the same universe as the carrier set (see the cubical interpretation of HIT in [26]).

4.3.2 The interpretation of mTT

In this subsection, I define the interpretation of the first level of **MF**, namely **mTT**, in **HoTT** and show that the interpretation is sound and preserves substitution and judgemental equality. Further, also the distinction between **MF**-sorts is preserved, contrary to what happens when **mTT** is interpreted in **MLTT**.
We are going to define by induction over the raw-syntax of **mTT** a partial interpretation map $(-)^{\blacksquare}$, mapping **mTT** raw-expressions into the raw-syntax of **HoTT**. Further, we introduce two auxiliary partial functions pr_{S} and pr_{P} associating to a type symbol of **HoTT** raw-syntax either a proof that it is a h-set, in the case of pr_{S}, or a proof that it is a h-proposition, in the case of pr_{P}. In this way, we can interpret **mTT**-collections as h-sets in $\mathcal{U}_1$ and

mTT-sets as h-sets in $\mathcal{U}_0$. Similarly, **mTT**-propositions are interpreted as h-propositions in $\mathcal{U}_1$ and **mTT**-small propositions as h-propositions in $\mathcal{U}_0$. Compared to the presentation of **mTT** in [89], the formulation of the collection of small propositions is modified in order to make the interpretation in **HoTT** smoother. In fact, we introduce a Tarski-style definition of this collection that comes equipped with codes for small propositions and a decoding operator:

$$\text{Pr}_1)\ \ \widehat{\bot} \in \mathsf{prop_s} \qquad \text{Pr}_2)\ \frac{p \in \mathsf{prop_s} \quad q \in \ \mathsf{prop_s}}{p \mathbin{\widehat{\vee}} q \in \mathsf{prop_s}}$$

$$\text{Pr}_3)\ \frac{p \in \mathsf{prop_s} \qquad q \in \mathsf{prop_s}}{p \mathbin{\widehat{\to}} q \in \mathsf{prop_s}} \qquad \text{Pr}_4)\ \frac{p \in \mathsf{prop_s} \qquad q \in \mathsf{prop_s}}{p \mathbin{\widehat{\wedge}} q \in \mathsf{prop_s}}$$

$$\text{Pr}_5)\ \frac{p \in \mathsf{prop_s}\ [x \in A] \qquad A\ set}{\widehat{(\forall x \in A)}\, p \in \mathsf{prop_s}} \qquad \text{Pr}_6)\ \frac{p \in \mathsf{prop_s}\ [x \in A] \qquad A\ set}{\widehat{(\exists x \in A)}\, p \in \mathsf{prop_s}}$$

$$\text{Pr}_7)\ \frac{A\ set \quad a \in A \quad b \in A}{\widehat{\mathsf{Id}}(A, a, b) \in \mathsf{prop_s}}$$

Therefore, elements of the collection of small propositions can be decoded as small propositions by means of the following operator

$$\tau\text{-Pr})\ \frac{p \in \mathsf{prop_s}}{\tau(p)\ prop_s}$$

which satisfies the following definitional equalities:

$$\text{eq-Pr}_1)\ \ \tau(\widehat{\bot}) = \bot\ prop_s \qquad \text{eq-Pr}_2)\ \frac{p \in \mathsf{prop_s} \quad q \in \mathsf{prop_s}}{\tau(p \mathbin{\widehat{\vee}} q) = \tau(p) \vee \tau(q)\, prop_s}$$

$$\text{eq-Pr}_3)\ \frac{p \in \mathsf{prop_s} \qquad q \in \mathsf{prop_s}}{\tau(p \mathbin{\widehat{\to}} q) = \tau(p) \to \tau(q)\, prop_s} \qquad \text{eq-Pr}_4)\ \frac{p \in \mathsf{prop_s} \quad q \in \mathsf{prop_s}}{\tau(p \mathbin{\widehat{\wedge}} q) = \tau(p) \wedge \tau(q)\, prop_s}$$

$$\text{eq-Pr}_5)\ \frac{p \in \mathsf{prop_s}\ [x \in A] \qquad A\ set}{\tau(\widehat{(\forall x \in A)}\, p) = (\forall x \in A)\, \tau(p)\, prop_s} \qquad \text{eq-Pr}_6)\ \frac{p \in \mathsf{prop_s}\ [x \in A] \qquad A\ set}{\tau(\widehat{(\exists x \in A)}\, p) = (\exists x \in A)\, \tau(p)\, prop_s}$$

$$\text{eq-Pr}_7)\ \frac{A\ set \quad a \in A \quad b \in A}{\tau(\,\widehat{\mathsf{Id}}(A, a, b)\,) = \mathsf{Id}(A, a, b)\, prop_s}$$

Definition 4.3.19 (**mTT**-raw syntax)**.** The raw-syntax of **mTT** is defined as follows:

1. Pre-contexts:

   - The empty list $[-]$ is the empty pre-context.
   - If Γ is a pre-context, A is a pre-type and and x is a variable (not occurring in Γ), then $[\Gamma, x \in A\ ]$ is a pre-context.

2. Pre-types:

   - The pre-sets and pre-collections are the following:

     $$\mathsf{prop_s} \mid \Sigma_{x\in A} B(x) \mid A \to \mathsf{prop_s} \mid \Pi_{x\in A} B(x) \mid \mathsf{N}_0 \mid \mathsf{N}_1 \mid A + B \mid \mathsf{List}(A)$$

   - The pre-propositions are the following:

     $$\bot \mid A \wedge B \mid A \vee B \mid A \to B \mid \exists_{x\in A} B(x) \mid \forall_{x\in A} B(x) \mid \mathrm{Id}(A, a, b)$$

- Pre-terms:

$$x \mid \widehat{\bot} \mid p \widehat{\wedge} q \mid p \widehat{\vee} q \mid p \widehat{\to} q \mid \widehat{\exists_{x\in A}} B(x) \mid \widehat{\forall_{x\in A}} B(x) \mid \widehat{\mathrm{Id}}(A, a, b) \mid$$
$$(a, b) \mid \mathsf{El}_{\Sigma}(d, c) \mid \lambda x.b(x) \mid \mathsf{Ap}(f, a) \mid \mathsf{emp_0}(c) \mid \star \mid \mathsf{El}_{\mathsf{N}_1}(t, c) \mid$$
$$\mathsf{inl}(a) \mid \mathsf{inr}(b) \mid \mathsf{El}_{+}(c, d_A, d_B) \mid \epsilon \mid \mathsf{cons}(\ell, a) \mid \mathsf{El}_{\mathsf{List}}(c, d, l) \mid$$
$$\mathsf{r}_0(c) \mid \mathsf{inl}_{\vee}(a) \mid \mathsf{inr}_{\vee}(b) \mid \mathsf{El}_{\vee}(d, c_A, c_B) \mid \langle a, _{\wedge}\, b\rangle \mid \pi_1(c) \mid \pi_2(c) \mid$$
$$\lambda_{\to} x.b \mid \mathsf{Ap}_{\to}(f, a) \mid \langle a, _{\exists}\, b\rangle \mid \mathsf{El}_{\exists}(d, c) \mid \lambda_{\forall} x.b(x) \mid \mathsf{Ap}_{\forall}(f, a) \mid$$
$$\mathsf{id}_A(a) \mid \mathsf{El}_{\mathrm{Id}}(p, c)$$

After having introduced **mTT**-raw syntax, we can define the interpretation:

Definition 4.3.20. We define a partial interpretation

$$(-)^{\blacksquare} : \text{raw-syntax } (\mathbf{mTT}) \ \to\ \text{raw-syntax } (\mathbf{HoTT})$$

together with two auxiliary partial mappings

$$\mathsf{pr}_{\mathsf{P}}(-) : \text{raw-syntax } (\mathbf{HoTT}) \ \rightarrow \ \text{raw-syntax } (\mathbf{HoTT})$$

and

$$\mathsf{pr}_{\mathsf{S}}(-) : \text{raw-syntax } (\mathbf{HoTT}) \ \rightarrow \ \text{raw-syntax } (\mathbf{HoTT})$$

where the first associates to some type expressions of **HoTT** a proof-term encoding the fact that they are h-propositions, while the latter associates to some type expressions of **HoTT** a proof-term encoding the fact that they are h-sets.
The translation naturally extends to **mTT**-contexts:

$$\begin{aligned}
&([-])^{\blacksquare} \ :\equiv \ \cdot \\
&(\Gamma, x \in A)^{\blacksquare} \ :\equiv \ \Gamma^{\blacksquare}, \ x : A^{\blacksquare} \\
&(x \in A \ [\Gamma])^{\blacksquare} \ :\equiv \ x : A^{\blacksquare} \ [\Gamma^{\blacksquare}]
\end{aligned}$$

The interpretation of **mTT** pre-judgements is defined as follows:

$$\begin{aligned}
&(A \ \mathit{set} \ [\Gamma])^{\blacksquare} \ :\equiv \ A^{\blacksquare} : \mathcal{U}_0 \ [\Gamma^{\blacksquare}] \text{ s.th. } \mathsf{pr}_{\mathsf{S}}(A^{\blacksquare}) : \mathsf{isSet}(A^{\blacksquare}) \\
&(A \ \mathit{col} \ [\Gamma])^{\blacksquare} \ :\equiv \ A^{\blacksquare} : \mathcal{U}_1 \ [\Gamma^{\blacksquare}] \text{ s.th. } \mathsf{pr}_{\mathsf{S}}(A^{\blacksquare}) : \mathsf{isSet}(A^{\blacksquare}) \\
&(P \ \mathit{prop}_s \ [\Gamma])^{\blacksquare} \ :\equiv \ P^{\blacksquare} : \mathcal{U}_0 \ [\Gamma^{\blacksquare}] \text{ s.th. } \mathsf{pr}_{\mathsf{P}}(P^{\blacksquare}) : \mathsf{isProp}(P^{\blacksquare}) \\
&(P \ \mathit{prop} \ [\Gamma])^{\blacksquare} \ :\equiv \ P^{\blacksquare} : \mathcal{U}_1 \ [\Gamma^{\blacksquare}] \text{ s.th. } \mathsf{pr}_{\mathsf{P}}(P^{\blacksquare}) : \mathsf{isProp}(P^{\blacksquare}) \\
&(A = B \ \mathit{set} \ [\Gamma])^{\blacksquare} \ :\equiv \ (A^{\blacksquare}, \mathsf{pr}_{\mathsf{S}}(A^{\blacksquare})) \ \equiv (B^{\blacksquare}, \mathsf{pr}_{\mathsf{S}}(B^{\blacksquare})) : \mathsf{Set}_{\mathcal{U}_0} \ [\Gamma^{\blacksquare}] \\
&(A = B \ \mathit{col} \ [\Gamma])^{\blacksquare} \ :\equiv \ (A^{\blacksquare}, \mathsf{pr}_{\mathsf{S}}(A^{\blacksquare})) \ \equiv (B^{\blacksquare}, \mathsf{pr}_{\mathsf{S}}(B^{\blacksquare})) : \mathsf{Set}_{\mathcal{U}_1} \ [\Gamma^{\blacksquare}] \\
&(P = Q \ \mathit{prop}_s \ [\Gamma])^{\blacksquare} \ :\equiv \ (P^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}(P^{\blacksquare})) \equiv (Q^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}(Q^{\blacksquare})) : \mathsf{Prop}_{\mathcal{U}_0} \ [\Gamma^{\blacksquare}] \\
&(P = Q \ \mathit{prop} \ [\Gamma])^{\blacksquare} \ :\equiv \ (P^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}(P^{\blacksquare})) \equiv (Q^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}(Q^{\blacksquare})) : \mathsf{Prop}_{\mathcal{U}_1} \ [\Gamma^{\blacksquare}] \\
&(a \in A \ \ [\Gamma])^{\blacksquare} \ :\equiv \ a^{\blacksquare} : A^{\blacksquare} \ [\Gamma^{\blacksquare}] \\
&(a = b \in A \ [\Gamma])^{\blacksquare} \ :\equiv \ a^{\blacksquare} \equiv b^{\blacksquare} : A^{\blacksquare} \ [\Gamma^{\blacksquare}]
\end{aligned}$$

Finally, the interpretation of **mTT**-raw syntax is given as follows: **mTT**-variables are interpreted as **HoTT**-variables without renaming and **mTT**-types and terms raw expressions are interpreted as specified in the following clauses

$(\mathsf{prop_s})^\blacksquare :\equiv \mathsf{Prop}_{\mathcal{U}_0}$

$\mathsf{pr_S}((\mathsf{prop_s})^\blacksquare) :\equiv \mathfrak{s}_{\mathsf{Prop_0}}$

$(\tau(p))^\blacksquare :\equiv \mathsf{pr}_1(p^\blacksquare)$

$\mathsf{pr_P}((\tau(p))^\blacksquare) :\equiv \mathsf{pr}_2(p^\blacksquare)$

$\mathsf{pr_S}((\tau(p))^\blacksquare) :\equiv \mathfrak{s}_{coe}((\tau(p))^\blacksquare, \mathsf{pr_P}((\tau(p))^\blacksquare))$

$(\widehat{\bot})^\blacksquare :\equiv \ (\mathbf{0},\ \mathfrak{p}_0)$

$(\widehat{\top})^\blacksquare \ :\equiv \ (\mathbf{1},\ \mathfrak{p}_1\,)$

$(p \mathbin{\widehat{\vee}} q)^\blacksquare :\equiv (\mathsf{pr}_1(p^\blacksquare) \vee \mathsf{pr}_1(q^\blacksquare)\ ,\ \mathfrak{p}_\vee(\mathsf{pr}_1(p^\blacksquare), \mathsf{pr}_1(q^\blacksquare)\,))$

$(p \mathbin{\widehat{\wedge}} q)^\blacksquare :\equiv (\mathsf{pr}_1(p^\blacksquare) \times \mathsf{pr}_1(q^\blacksquare)\ ,\ \mathfrak{p}_\times(\mathsf{pr}_1(p^\blacksquare), \mathsf{pr}_1(q^\blacksquare), \mathsf{pr}_2(p^\blacksquare), \mathsf{pr}_2(q^\blacksquare)\,))$

$(p \mathbin{\widehat{\rightarrow}} q)^\blacksquare \ :\equiv \ (\mathsf{pr}_1(p^\blacksquare) \to \mathsf{pr}_1(q^\blacksquare)\ ,\ \mathfrak{p}_\to(\mathsf{pr}_1(p^\blacksquare), \mathsf{pr}_1(q^\blacksquare), \mathsf{pr}_2(p^\blacksquare), \mathsf{pr}_2(q^\blacksquare)\,))$

$(\widehat{\exists}_{x\in A}\, p(x))^\blacksquare \ :\equiv \ (\exists_{x:A^\blacksquare}\ p(x)^\blacksquare,\ \mathfrak{p}_\exists(A^\blacksquare, \lambda x.\mathsf{pr}_1(p(x)^\blacksquare)))$

$(\widehat{\forall}_{x\in A}\, p(x))^\blacksquare \ :\equiv \ (\Pi_{x:A^\blacksquare}\ p(x)^\blacksquare\ ,\ \mathfrak{p}_\Pi(A^\blacksquare, \lambda x.\mathsf{pr}_1(p(x)^\blacksquare),\ \lambda x.\mathsf{pr}_2(p(x)^\blacksquare)))$

$(\widehat{\mathrm{Id}}(A, a, b))^\blacksquare \ :\equiv \ (\mathrm{Id}_{A^\blacksquare}(a^\blacksquare, b^\blacksquare)\ ,\ \mathfrak{p}_{Id}(A^\blacksquare, \mathsf{pr_S}(A^\blacksquare), a^\blacksquare, b^\blacksquare\ ))$

$(A \ \to \ \mathsf{prop_s})^\blacksquare :\equiv \ A^\blacksquare \ \to \ \mathsf{Prop}_{\mathcal{U}_0}$

$(\lambda x.b(x))^\blacksquare \ :\equiv \ \lambda x.b(x)^\blacksquare$

$\mathsf{pr_S}((A \ \to \ \mathsf{prop_s})^\blacksquare) :\equiv \mathfrak{s}_\Pi(A^\blacksquare, \lambda x : A^\blacksquare.\mathsf{Prop}_{\mathcal{U}_0}, \mathfrak{s}_{\mathsf{Prop_0}}\ )$

$(\mathsf{Ap}(f, a))^\blacksquare :\equiv \ f^\blacksquare(a^\blacksquare)$

$(\ \Sigma_{x\in A} B(x)\ )^\blacksquare \ :\equiv \Sigma_{x:A^\blacksquare}\ B(x)^\blacksquare$

$(\langle a, b\rangle)^\blacksquare \ :\equiv \ (a^\blacksquare, b^\blacksquare)$

$(\mathrm{El}_\Sigma(d, c))^\blacksquare \ :\equiv \ \mathsf{ind}_\Sigma(d^\blacksquare, x.y.c(x, y)^\blacksquare)$

$\mathsf{pr_S}((\Sigma_{x\in A} B(x)\ )^\blacksquare) \ :\equiv \ \mathfrak{s}_\Sigma(A^\blacksquare, \lambda x : A^\blacksquare.B(x)^\blacksquare, \mathsf{pr_S}(A^\blacksquare), \lambda x : A^\blacksquare.\mathsf{pr_S}(B(x)^\blacksquare))$

$(\Pi_{x\in A} B(x))^\blacksquare \ :\equiv \ \Pi_{x:A^\blacksquare}\ B(x)^\blacksquare$

$(\lambda x.b(x))^\blacksquare \ :\equiv \ \lambda x.b(x)^\blacksquare$

$\mathsf{pr_S}(\Pi_{x\in A} B(x)^\blacksquare) \ :\equiv \ \mathfrak{s}_\Pi(A^\blacksquare, \lambda x : A^\blacksquare.B(x)^\blacksquare, \lambda x : A^\blacksquare.\mathsf{pr_S}(B(x)^\blacksquare))$

$(\mathsf{Ap}(f, a))^\blacksquare \ :\equiv \ f^\blacksquare(a^\blacksquare)$

$(\mathsf{N}_0)^\blacksquare \;:\equiv\; \mathbf{0}$

$(\mathsf{emp}_0(c))^\blacksquare \;:\equiv\; \mathsf{ind}_0(c^\blacksquare)$

$\mathsf{pr}_\mathsf{S}((\mathsf{N}_0)^\blacksquare) \;:\equiv\; \mathfrak{s}_0$

$(\mathsf{N}_1)^\blacksquare \;:\equiv\; \mathbf{1}$

$(\star)^\blacksquare \;:\equiv\; \star$

$\mathsf{pr}_\mathsf{S}((\mathsf{N}_1)^\blacksquare) \;:\equiv\; \mathfrak{s}_1$

$(\mathrm{El}_{\mathsf{N}_1}(t, c))^\blacksquare \;:\equiv\; \mathsf{ind}_1(t^\blacksquare, c^\blacksquare)$

$(A + B)^\blacksquare \;:\equiv\; A^\blacksquare + B^\blacksquare$

$(\mathsf{inl}(a))^\blacksquare \;:\equiv\; \mathsf{inl}(a^\blacksquare)$

$(\mathsf{inr}(b))^\blacksquare \;:\equiv\; \mathsf{inr}(b^\blacksquare)$

$(\mathrm{El}_+(c, d_A, d_B))^\blacksquare \;:\equiv\; \mathsf{ind}_+(c^\blacksquare, x.d_A(x)^\blacksquare, y.d_B(y)^\blacksquare)$

$\mathsf{pr}_\mathsf{S}((A + B)^\blacksquare) \;:\equiv\; \mathfrak{s}_+(A^\blacksquare, B^\blacksquare, \mathsf{pr}_\mathsf{S}(A^\blacksquare), \mathsf{pr}_\mathsf{S}(B^\blacksquare))$

$(\mathsf{List}(A))^\blacksquare \;:\equiv\; \mathsf{List}(A^\blacksquare)$

$(\epsilon)^\blacksquare \;:\equiv\; \mathsf{nil}$

$(\mathsf{cons}(\ell, a))^\blacksquare \;:\equiv\; \mathsf{cons}(\ell^\blacksquare, a^\blacksquare)$

$\mathsf{pr}_\mathsf{S}((\mathsf{List}(A))^\blacksquare) \;:\equiv\; \mathfrak{s}_{\mathsf{List}}(A^\blacksquare, \mathsf{pr}_\mathsf{S}(A^\blacksquare))$

$(\mathrm{El}_{\mathsf{List}}(c, d, l))^\blacksquare \;:\equiv\; \mathsf{ind}_{\mathsf{List}}(c^\blacksquare, d^\blacksquare, x.y.z.l(x, y, z)^\blacksquare)$

$(\bot)^\blacksquare \;:\equiv\; \mathbf{0}$

$(\mathsf{r}_0(c))^\blacksquare \;:\equiv\; \mathsf{ind}_0(c^\blacksquare)$

$\mathsf{pr}_\mathsf{P}((\bot)^\blacksquare) \;:\equiv\; \mathfrak{p}_0$

$\mathsf{pr}_\mathsf{S}((\bot)^\blacksquare) \;:\equiv\; \mathfrak{s}_{coe}((\bot)^\blacksquare, \mathsf{pr}_\mathsf{P}((\bot)^\blacksquare))$

$$
\begin{aligned}
&(A \vee B)^{\blacksquare} \;:\equiv\; A^{\blacksquare} \vee B^{\blacksquare}\\
&(\mathsf{inl}_{\vee}(a))^{\blacksquare} \;:\equiv\; \mathsf{inl}_{\vee}(a^{\blacksquare})\\
&(\mathsf{inr}_{\vee}(b))^{\blacksquare} \;:\equiv\; \mathsf{inr}_{\vee}(b^{\blacksquare})\\
&(\mathrm{El}_{\vee}(d, c_A, .c_B))^{\blacksquare} \;:\equiv\; \mathsf{ind}_{\vee}(d^{\blacksquare}, x.c_1(x)^{\blacksquare}, y.c_2(y)^{\blacksquare})\\
&\mathsf{pr}_{\mathsf{P}}((A \vee B)^{\blacksquare}) \;:\equiv\; \mathfrak{p}_{\vee}(A^{\blacksquare}, B^{\blacksquare})\\
&\mathsf{pr}_{\mathsf{S}}((A \vee B)^{\blacksquare}) \;:\equiv\; \mathfrak{s}_{coe}((A \vee B)^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}((A \vee B)^{\blacksquare}))
\end{aligned}
$$

$$
\begin{aligned}
&(A \wedge B)^{\blacksquare} :\equiv\; A^{\blacksquare} \times B^{\blacksquare}\\
&(\langle a,_{\wedge}\, b\rangle)^{\blacksquare} \;:\equiv\; (a^{\blacksquare}, b^{\blacksquare})\\
&\mathsf{pr}_{\mathsf{P}}((A \wedge B)^{\blacksquare}) \;:\equiv\; \mathfrak{p}_{\times}(A^{\blacksquare}, B^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}(A^{\blacksquare}), \mathsf{pr}_{\mathsf{P}}(B^{\blacksquare}))\\
&(\pi_i(c))^{\blacksquare} \;:\equiv\; \mathsf{pr}_i(c^{\blacksquare})(\text{with } i \in \{1,2\})\\
&\mathsf{pr}_{\mathsf{S}}((A \wedge B)^{\blacksquare}) \;:\equiv\; \mathfrak{s}_{coe}((A \wedge B)^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}((A \wedge B)^{\blacksquare}))
\end{aligned}
$$

$$
\begin{aligned}
&(A \to B)^{\blacksquare} \;:\equiv\; A^{\blacksquare} \to B^{\blacksquare}\\
&(\lambda_{\to} x.b)^{\blacksquare} \;:\equiv\; \lambda x.b^{\blacksquare}\\
&\mathsf{pr}_{\mathsf{P}}((A \to B)^{\blacksquare}) \;:\equiv\; \mathfrak{p}_{\to}(A^{\blacksquare}, B^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}(A^{\blacksquare}), \mathsf{pr}_{\mathsf{P}}(B^{\blacksquare}))\\
&(\mathsf{Ap}_{\to}(f, a))^{\blacksquare} \;:\equiv\; f^{\blacksquare}(a^{\blacksquare})\\
&\mathsf{pr}_{\mathsf{S}}((A \to B)^{\blacksquare}) \;:\equiv\; \mathfrak{s}_{coe}((A \to B)^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}((A \to B)^{\blacksquare}))
\end{aligned}
$$

$$
\begin{aligned}
&(\exists_{x \in A} B(x))^{\blacksquare} \;:\equiv\; \exists_{x:A^{\blacksquare}}\, B(x)^{\blacksquare}\\
&(\langle a,_{\exists}\, b\rangle)^{\blacksquare} \;:\equiv\; (a^{\blacksquare},_{\exists}\, b^{\blacksquare})\\
&(\mathrm{El}_{\exists}(d, c))^{\blacksquare} \;:\equiv\; \mathsf{ind}_{\exists}(d^{\blacksquare}, x.y.c(x, y)^{\blacksquare})\\
&\mathsf{pr}_{\mathsf{P}}((\exists_{x \in A} B(x))^{\blacksquare}) \;:\equiv\; \mathfrak{p}_{\exists}(A^{\blacksquare}, \lambda x : A^{\blacksquare}.B(x)^{\blacksquare})\\
&\mathsf{pr}_{\mathsf{S}}((\exists_{x \in A} B(x))^{\blacksquare}) \;:\equiv\; \mathfrak{s}_{coe}((\exists_{x \in A} B(x))^{\blacksquare},\ \mathsf{pr}_{\mathsf{P}}((\exists_{x \in A} B(x))^{\blacksquare}))
\end{aligned}
$$

$$
\begin{aligned}
&(\forall_{x \in A} B(x))^{\blacksquare} :\equiv \Pi_{x:A^{\blacksquare}}\, B(x)^{\blacksquare}\\
&(\lambda_{\forall} x.b(x))^{\blacksquare} \;:\equiv\; \lambda x.b(x)^{\blacksquare}\\
&\mathsf{pr}_{\mathsf{P}}((\forall_{x \in A} B(x))^{\blacksquare}) \;:\equiv\; \mathfrak{p}_{\Pi}(A^{\blacksquare}, \lambda x : A^{\blacksquare}.B(x)^{\blacksquare}, \lambda x : A^{\blacksquare}.\mathsf{pr}_{\mathsf{P}}(B(x)^{\blacksquare}))\\
&(\mathsf{Ap}_{\forall}(f, a))^{\blacksquare} \;:\equiv\; f^{\blacksquare}(a^{\blacksquare})\\
&\mathsf{pr}_{\mathsf{S}}((\forall_{x \in A} B(x))^{\blacksquare}) \;:\equiv\; \mathfrak{s}_{coe}((\forall_{x \in A} B(x))^{\blacksquare},\ \mathsf{pr}_{\mathsf{P}}((\forall_{x \in A} B(x))^{\blacksquare}))
\end{aligned}
$$

$$
\begin{aligned}
(\mathrm{Id}(A,a,b))^{\blacksquare} &:\equiv \mathrm{Id}_{A^{\blacksquare}}(a^{\blacksquare}, b^{\blacksquare}) \\
(\mathsf{id}_{\mathsf{A}}(a))^{\blacksquare} &:\equiv \mathsf{refl}_{a^{\blacksquare}} \\
(\mathrm{El}_{\mathrm{Id}}(p,c))^{\blacksquare} &:\equiv \mathsf{ind}_{\mathrm{Id}}(p^{\blacksquare},\ x.c(x)^{\blacksquare}) \\
\mathsf{pr}_{\mathsf{P}}((\mathrm{Id}(A,a,b))^{\blacksquare}) &:\equiv \mathfrak{p}_{Id}(A^{\blacksquare}, \mathsf{pr}_{\mathsf{S}}(A^{\blacksquare}), a^{\blacksquare}, b^{\blacksquare}) \\
\mathsf{pr}_{\mathsf{S}}((\mathrm{Id}(A,a,b))^{\blacksquare}) &:\equiv \mathfrak{s}_{coe}((\mathrm{Id}(A,a,b))^{\blacksquare},\ \mathsf{pr}_{\mathsf{P}}((\mathrm{Id}(A,a,b))^{\blacksquare}))
\end{aligned}
$$

Then, we show that the interpretation is well-defined over substitution, in the sense that substitution over **mTT**-expressions corresponds to substitution over **HoTT**-expressions, as it is expressed by the following lemma

Lemma 4.3.21. *If A is a raw-type in* **mTT***, b is a* **mTT** *raw-term and x is a variable occurring free in A, then*

$$(A[b/x])^{\blacksquare} :\equiv A^{\blacksquare}[b^{\blacksquare}/x^{\blacksquare}]$$

If a and b are **mTT** *raw-terms and x is a variable occurring free in a, then*

$$(a[b/x])^{\blacksquare} :\equiv a^{\blacksquare}[b^{\blacksquare}/x^{\blacksquare}]$$

Proof. The lemma is established by structural induction. □

Now the main theorem of this subsection asserts the validity of the interpretation defined above:

Theorem 4.3.22. *If $\mathcal{J}$ is a derivable judgement in* **mTT***, then the interpretation of $\mathcal{J}$ is a derivable judgement in* **HoTT**.
Moreover, if P prop $[\Gamma]$ and P prop$_s$ $[\Gamma]$ are derivable judgements in **mTT***, then $\mathsf{pr}_{\mathsf{S}}(P^{\blacksquare}) : \mathsf{isSet}(P^{\blacksquare})\ [\Gamma^{\blacksquare}]$ is derivable in* **HoTT***, where*

$$\mathsf{pr}_{\mathsf{S}}(P^{\blacksquare}) :\equiv \mathfrak{s}_{coe}((P)^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}(P^{\blacksquare})) : \mathsf{isSet}(P^{\blacksquare}).$$

Proof. The proof is by induction over the derivation of $\mathcal{J}$. The subtyping rules (sP-in-P) and (S-in-C) are trivially validated, since the universes are cumulative. The remaining two (P-in-C) and (sP-in-S) are interpreted as follows: let us consider the rule (sP-in-S) first, since the other is validated by a similar reasoning. By induction hypothesis, $P^{\blacksquare} : \mathcal{U}_0\ [\Gamma^{\blacksquare}]$ and $\mathsf{pr}_{\mathsf{P}}(P^{\blacksquare}) : \mathsf{isProp}(P^{\blacksquare})\ [\Gamma^{\blacksquare}]$ are derivable. Moreover, we can also derive

$\mathsf{pr}_{\mathsf{S}}(P^{\blacksquare}) \equiv \mathfrak{s}_{coe}(P^{\blacksquare}, \mathsf{pr}_{\mathsf{P}}(P^{\blacksquare})) : \mathsf{isSet}(P^{\blacksquare})$, that makes the conclusion valid. The other rule follows by applying the same argument within the second universe $\mathcal{U}_1$.
The collection of small propositions is interpreted by $\mathsf{Prop}_{\mathcal{U}_0} : \mathcal{U}_1\ [\Gamma^{\blacksquare}]$ together with $\mathfrak{s}_{\mathsf{prop}_0} : \mathsf{isSet}(\mathsf{Prop}_{\mathcal{U}_0})\ [\Gamma^{\blacksquare}]$. The rule for the decoding operator τ

$$\frac{p \in \mathsf{prop}_s\ [\Gamma]}{\tau(p)\ prop_s\ [\Gamma]}\ (\tau\text{-Pr})$$

is interpreted as follows: by induction hypothesis, $p^{\blacksquare} : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^{\blacksquare}]$ and thus $\mathsf{pr}_1(p^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}]$ and $\mathsf{pr}_2(p^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1(p^{\blacksquare}))\ [\Gamma^{\blacksquare}]$ are derivable, that are the interpretation of the conclusion by definition of the interpretation.
The encoding rules (Pr1-Pr7) are interpreted in the following way:

(Pr1) is trivially validated.

(Pr2) by induction hypothesis, $p^{\blacksquare} : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^{\blacksquare}]$ and $q^{\blacksquare} : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^{\blacksquare}]$. Therefore

$$\mathsf{pr}_1(p^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}] \text{ with } \mathsf{pr}_2(p^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1(p^{\blacksquare}))\ [\Gamma^{\blacksquare}]$$

and also

$$\mathsf{pr}_1(q^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}] \text{ with} \mathsf{pr}_2(q^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1(q^{\blacksquare}))\ [\Gamma^{\blacksquare}]$$

Hence,

$$\mathsf{pr}_1(p^{\blacksquare}) \vee \mathsf{pr}_1(q^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}]$$

and

$$\mathfrak{p}_{\vee}(\mathsf{pr}_1(p^{\blacksquare}), \mathsf{pr}_1(q^{\blacksquare})) : \mathsf{isProp}(\mathsf{pr}_1(p^{\blacksquare}) \vee \mathsf{pr}_1(q^{\blacksquare}))\ [\Gamma^{\blacksquare}]$$

follow.

(Pr3) by induction hypothesis, $p^{\blacksquare} : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^{\blacksquare}]$ and $q^{\blacksquare} : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^{\blacksquare}]$. It follows that

$$\mathsf{pr}_1(p^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}] \text{ with } \mathsf{pr}_2(p^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1(p^{\blacksquare}))\ [\Gamma^{\blacksquare}]$$

and also

$$\mathsf{pr}_1(q^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}] \text{ with } \mathsf{pr}_2(q^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1(q^{\blacksquare}))\ [\Gamma^{\blacksquare}]$$

Then

$$\mathsf{pr}_1(p^{\blacksquare}) \to \mathsf{pr}_1(q^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}]$$

and

$$\mathfrak{p}_{\to}(\mathsf{pr}_1(p^{\blacksquare}), \mathsf{pr}_1(q^{\blacksquare}), \mathsf{pr}_2(p^{\blacksquare}), \mathsf{pr}_2(q^{\blacksquare})) : \mathsf{isProp}(\mathsf{pr}_1(p^{\blacksquare}) \to \mathsf{pr}_1(q^{\blacksquare}))\ [\Gamma^{\blacksquare}].$$

(Pr4) by induction hypothesis, we can derive the same judgements as in Pr2 and Pr3. Then $\mathsf{pr}_1(p^\blacksquare) \times \mathsf{pr}_1(q^\blacksquare) : \mathcal{U}_0\ [\Gamma^\blacksquare]$ and

$$\mathfrak{p}_\times(\mathsf{pr}_1(p^\blacksquare), \mathsf{pr}_1(q^\blacksquare), \mathsf{pr}_2(p^\blacksquare), \mathsf{pr}_2(q^\blacksquare)) : \mathsf{isProp}(\mathsf{pr}_1(p^\blacksquare) \times \mathsf{pr}_1(q^\blacksquare))\ [\Gamma^\blacksquare].$$

(Pr5) by induction hypothesis, $p(x)^\blacksquare : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^\blacksquare, x : A^\blacksquare]$ and $A^\blacksquare : \mathcal{U}_0\ [\Gamma^\blacksquare]$ such that $\mathsf{pr}_\mathsf{S}(A^\blacksquare) : \mathsf{isSet}(A^\blacksquare)\ [\Gamma^\blacksquare]$. Then $\mathsf{pr}_1(p(x)^\blacksquare) : \mathcal{U}_0\ [\Gamma^\blacksquare]$ and $\mathsf{pr}_2(p(x)^\blacksquare) : \mathsf{isProp}(\mathsf{pr}_1(p(x)^\blacksquare))\ [\Gamma^\blacksquare]$ and thus we can conclude

$$\Pi_{x:A^\blacksquare} \mathsf{pr}_1(p(x)^\blacksquare) : \mathcal{U}_0\ [\Gamma^\blacksquare]$$

and

$$\mathfrak{p}_\Pi(A^\blacksquare, \lambda x.(\mathsf{pr}_1(p(x)^\blacksquare)), \lambda x.(\mathsf{pr}_2(p(x)^\blacksquare))) : \mathsf{isProp}(\Pi_{x:A^\blacksquare} \mathsf{pr}_1(p(x)^\blacksquare))\ [\Gamma^\blacksquare].$$

(Pr6) by induction hypothesis, the premises are interpreted as those of Pr5. Then $\exists_{x:A^\blacksquare} \mathsf{pr}_1(p(x)^\blacksquare) : \mathcal{U}_0\ [\Gamma^\blacksquare]$ and

$$\mathfrak{p}_\exists(A^\blacksquare, \lambda x.\mathsf{pr}_1(p(x)^\blacksquare)) : \mathsf{isProp}(\exists_{x:A^\blacksquare} \mathsf{pr}_1(p(x)^\blacksquare))\ [\Gamma^\blacksquare].$$

(Pr7) by induction hypothesis, $A^\blacksquare : \mathcal{U}_0\ [\Gamma^\blacksquare]$, $\mathsf{pr}_\mathsf{S}(A^\blacksquare) : \mathsf{isSet}(A^\blacksquare)\ [\Gamma^\blacksquare]$ and $a^\blacksquare, b^\blacksquare : A^\blacksquare\ [\Gamma^\blacksquare]$. Then

$$\mathrm{Id}_{A^\blacksquare}(a^\blacksquare, b^\blacksquare) : \mathcal{U}_0\ [\Gamma^\blacksquare]$$

and

$$\mathfrak{p}_{Id}(A^*, \mathsf{pr}_\mathsf{S}(A^*), a^*, b^*) : \mathsf{isProp}(\mathrm{Id}_{A^\blacksquare}(a^\blacksquare, b^\blacksquare))\ [\Gamma^\blacksquare].$$

The decoding operator satisfies the judgemental equalities (eq-Pr1)-(eq-Pr7), that are also validated by construction:

(eq-Pr1)

$$(\tau(\widehat{\bot})\ [\Gamma])^\blacksquare \ :\equiv\ \mathsf{pr}_1((\widehat{\bot})^\blacksquare) : \mathcal{U}_0\ [\Gamma^\blacksquare]$$

and

$$\mathsf{pr}_\mathsf{P}(\tau(\widehat{\bot}))\ :\equiv\ \mathsf{pr}_2((\widehat{\bot})^\blacksquare) : \mathsf{isProp}(\mathsf{pr}_1(\widehat{\bot})^\blacksquare)\ [\Gamma^\blacksquare].$$

But $\mathsf{pr}_1((\widehat{\bot})^\blacksquare) \equiv \mathbf{0}$ and $\mathsf{pr}_2((\widehat{\bot})^\blacksquare) \equiv \mathfrak{p}_\mathbf{0}$ and thus the judgemental equality is validated.

(eq-Pr2)

$$(\tau(p \mathbin{\hat{\vee}} q)\ [\Gamma])^{\blacksquare} \ :\equiv\ \mathsf{pr}_1((p \mathbin{\hat{\vee}} q)^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}]$$

and

$$\mathsf{pr}_{\mathsf{P}}(\tau(p \mathbin{\hat{\vee}} q)) \ :\equiv\ \mathsf{pr}_2((p \mathbin{\hat{\vee}} q)^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1((p \mathbin{\hat{\vee}} q)^{\blacksquare})).$$

Then

$$\mathsf{pr}_1((p \mathbin{\hat{\vee}} q)^{\blacksquare}) \equiv \mathsf{pr}_1(p^{\blacksquare}) \vee \mathsf{pr}_1(q^{\blacksquare})$$

and

$$\mathsf{pr}_2((p \mathbin{\hat{\vee}} q)^{\blacksquare}) \equiv \mathfrak{p}_{\vee}(\mathsf{pr}_1(p^{\blacksquare}), \mathsf{pr}_1(q^{\blacksquare}))$$

Therefore, the judgemental equality immediately follows.

(eq-Pr3)

$$(\tau(p \mathbin{\hat{\to}} q)\ [\Gamma])^{\blacksquare} \ :\equiv\ \mathsf{pr}_1((p \mathbin{\hat{\to}} q)^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}]$$

and

$$\mathsf{pr}_{\mathsf{P}}((\tau(p \mathbin{\hat{\to}} q))^{\blacksquare}) \ :\equiv\ \mathsf{pr}_2((p \mathbin{\hat{\to}} q)^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1((p \mathbin{\hat{\to}} q)^{\blacksquare})).$$

Then

$$\mathsf{pr}_1((p \mathbin{\hat{\to}} q)^{\blacksquare}) \equiv \mathsf{pr}_1(p^{\blacksquare}) \to \mathsf{pr}_1(q^{\blacksquare})$$

and

$$\mathsf{pr}_2((p \mathbin{\hat{\to}} q)^{\blacksquare}) \equiv \mathfrak{p}_{\to}(\mathsf{pr}_1(p^{\blacksquare}), \mathsf{pr}_1(q^{\blacksquare}), \mathsf{pr}_2(p^{\blacksquare}), \mathsf{pr}_2(q^{\blacksquare})).$$

Therefore, the judgmental equality immediately follows.

(eq-Pr4)

$$(\tau(p \mathbin{\hat{\wedge}} q)[\Gamma])^{\blacksquare} \ :\equiv\ \mathsf{pr}_1((p \mathbin{\hat{\wedge}} q)^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}]$$

and

$$\mathsf{pr}_{\mathsf{P}}((\tau(p \mathbin{\hat{\wedge}} q))^{\blacksquare}) \ :\equiv\ \mathsf{pr}_2((p \mathbin{\hat{\wedge}} q)^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1((p \mathbin{\hat{\wedge}} q)^{\blacksquare})).$$

Then

$$\mathsf{pr}_1((p \mathbin{\hat{\wedge}} q)^{\blacksquare}) \equiv \mathsf{pr}_1(p^{\blacksquare}) \times \mathsf{pr}_1(q^{\blacksquare})$$

and

$$\mathsf{pr}_2((p \mathbin{\hat{\wedge}} q)^{\blacksquare}) \equiv \mathfrak{p}_{\times}(\mathsf{pr}_1(p^{\blacksquare}), \mathsf{pr}_1(q^{\blacksquare}), \mathsf{pr}_2(p^{\blacksquare}), \mathsf{pr}_2(q^{\blacksquare})).$$

Therefore, the judgemental equality immediately follows.

(eq-Pr5)

$$(\tau(\widehat{\forall}_{x\in A}p(x))\ [\Gamma])^{\blacksquare} \ :\equiv\ \mathsf{pr}_1((\widehat{\forall}_{x\in A}p(x))^{\blacksquare}) : \mathcal{U}_0\ [\Gamma^{\blacksquare}]$$

and

$$\mathsf{pr}_{\mathsf{P}}((\tau(\widehat{\forall}_{x\in A}p(x)))^{\blacksquare}) \ :\equiv\ \mathsf{pr}_2((\widehat{\forall}_{x\in A}p(x))^{\blacksquare}) : \mathsf{isProp}(\mathsf{pr}_1((\widehat{\forall}_{x\in A}p(x))^{\blacksquare})).$$

Then

$$\mathsf{pr}_1((\widehat{\forall}_{x\in A}p(x))^{\blacksquare}) \equiv \Pi_{x:A^{\blacksquare}}\mathsf{pr}_1(p(x)^{\blacksquare})$$

and

$$\mathsf{pr}_2((\widehat{\forall}_{x\in A}p(x))^{\blacksquare}) \equiv \mathfrak{p}_{\Pi}(A^{\blacksquare}, \lambda x.\mathsf{pr}_1(p(x)^{\blacksquare}), \lambda x.\mathsf{pr}_2(p(x)^{\blacksquare})).$$

Therefore, the judgemental equality immediately follows.

(eq-Pr6) and (eq-Pr7) are omitted, since they follow the same pattern.

The validity of judgements forming propositions also follows by the definition of the interpretation and Lemmas 4.3.1 and 4.3.4 together with Def.4.3.5 and Def.4.3.7.
The validity of judgements forming **mTT**-sets follows from the definition of the interpretation, the closure of h-sets under those constructors (Lemma 4.3.16) and the closure of $\mathcal{U}_0$ under set-constructors. The same argument works for the judgements forming **mTT**-collections. The rules for forming contexts and the structural rules are also trivially validated by the interpretation. Finally, the way the interpretation is defined for judgemental equalities makes immediately valid **mTT**-conversions. In particular, the judgemental equalities associated to the disjunction connective and the existential quantifier are valid thanks to the Lemmas 4.3.6 and 4.3.8.

□

Remark 4.3.23. It must be noted that the validity of the interpretation $(-)^{\blacksquare}$ depends on the proof-terms that have been chosen in the previous subsection. In particular, this fact is fundamental to establish that two judgementally equal **mTT**-types are interpreted as *judgementally equal pairs* in **HoTT**; that is, not only the resulting types in **HoTT** are judgementally equal, but also the associated proof-terms witnessing that such types are h-sets or h-propositions. The same remark applies to the validity of the coercion of propositions into sets.

4.3.3 An alternative interpretation of mTT

Def.4.3.11 allows to define an alternative interpretation for **mTT**-propositions. They can be all interpreted by a systematic application of propositional truncation. As result, we have access to a kind of 'canonical' representation of h-propositions in **HoTT**. The alternative interpretation enjoys the same properties as the interpretation given in the previous subsection, since only those clauses that concern propositions are changed and this modification does not have any significant consequence over the translation. This alternative representation of h-propositions will be compulsory in the case of **emTT**-propositions later, because in that case canonical isomorphisms between h-propositions must be taken into account.

Definition 4.3.24. We define a partial interpretation

$$(-)^{\bullet} : \text{raw-syntax } (\mathbf{mTT}) \ \rightarrow \ \text{raw-syntax } (\mathbf{HoTT})$$

together with two auxiliary partial mappings

$$\mathsf{pr}_{\mathsf{P}}(-) : \text{raw-syntax } (\mathbf{HoTT}) \ \rightarrow \ \text{raw-syntax } (\mathbf{HoTT})$$

and

$$\mathsf{pr}_{\mathsf{S}}(-) : \text{raw-syntax } (\mathbf{HoTT}) \ \rightarrow \ \text{raw-syntax } (\mathbf{HoTT}).$$

The interpretation $(-)^{\bullet}$ extends to contexts and variables in the same way as the interpretation given in Def.4.3.20.
The interpretation of **mTT** pre-judgements regarding propositions and small propositions is the following:

$$\begin{aligned}
&(P \ \mathit{prop}_s \ [\Gamma])^{\bullet} \ :\equiv \ ||P^{\bullet}|| : \mathcal{U}_0 \ [\Gamma^{\bullet}] \text{ s.th. } \mathsf{pr}_{\mathsf{P}}(||P^{\bullet}||) : \mathsf{isProp}(||P^{\bullet}||)\\
&(P \ \mathit{prop} \ [\Gamma])^{\bullet} \ :\equiv \ ||P^{\bullet}|| : \mathcal{U}_1 \ [\Gamma^{\bullet}] \text{ s.th. } \mathsf{pr}_{\mathsf{P}}(||P^{\bullet}||) : \mathsf{isProp}(||P^{\bullet}||)\\
&(P = Q \ \mathit{prop}_s \ [\Gamma])^{\bullet} \ :\equiv \ (||P^{\bullet}||, \mathsf{pr}_{\mathsf{P}}(||P^{\bullet}||)) \equiv (||Q^{\bullet}||, \mathsf{pr}_{\mathsf{P}}(||Q^{\bullet}||)) : \mathsf{Prop}_{\mathcal{U}_0} \ [\Gamma^{\bullet}]\\
&(P = Q \ \mathit{prop} \ [\Gamma])^{\bullet} \ :\equiv \ (||P^{\bullet}||, \mathsf{pr}_{\mathsf{P}}(||P^{\bullet}||)) \equiv (||Q^{\bullet}||, \mathsf{pr}_{\mathsf{P}}(||Q^{\bullet}||)) : \mathsf{Prop}_{\mathcal{U}_1} \ [\Gamma^{\bullet}]
\end{aligned}$$

The definition left unchanged for the other judgement forms.
For **mTT**-constructors I display only those clauses which are different from the corresponding ones in Def.4.3.20

$$(\tau(p))^\bullet :\equiv ||\mathsf{pr}_1(p^\bullet)||$$
$$\mathsf{pr}_\mathsf{P}((\tau(p))^\bullet) :\equiv \mathsf{pr}_2(p^\bullet)$$
$$\mathsf{pr}_\mathsf{S}((\tau(p))^\bullet) :\equiv \mathfrak{s}_{coe}((\tau(p))^\bullet, \mathsf{pr}_\mathsf{P}((\tau(p))^\bullet))$$
$$(\widehat{\bot})^\bullet :\equiv (||\mathbf{0}||, \mathfrak{p}_{||\mathbf{0}||})$$
$$(\widehat{\top})^\bullet :\equiv (||\mathbf{1}||, \mathfrak{p}_{||\mathbf{1}||})$$
$$(p \mathbin{\widehat{\vee}} q)^\bullet :\equiv (\mathsf{pr}_1(p^\bullet) \vee \mathsf{pr}_1(q^\bullet) \ , \ \mathfrak{p}_\vee(\mathsf{pr}_1(p^\bullet), \mathsf{pr}_1(q^\bullet)))$$
$$(p \mathbin{\widehat{\wedge}} q)^\bullet :\equiv (||\mathsf{pr}_1(p^\bullet) \times \mathsf{pr}_1(q^\bullet)|| \ , \ \mathfrak{p}_{||\times||}(\mathsf{pr}_1(p^\bullet), \mathsf{pr}_1(q^\bullet)))$$
$$(p \mathbin{\widehat{\to}} q)^\bullet :\equiv (||\mathsf{pr}_1(p^\bullet) \to \mathsf{pr}_1(q^\bullet)|| \ , \ \mathfrak{p}_{||\to||}(\mathsf{pr}_1(p^\bullet), \mathsf{pr}_1(q^\bullet)))$$
$$(\widehat{\exists}_{x\in A}\, p(x))^\bullet :\equiv (\exists_{x:A^\bullet}\ p(x)^\bullet, \ \mathfrak{p}_\exists(A^\bullet, \lambda x.\mathsf{pr}_1(p(x)^\bullet)) \ )$$
$$(\widehat{\forall}_{x\in A}\, p(x))^\bullet :\equiv (||\Pi_{x:A^\bullet}\ p(x)^\bullet|| \ , \ \mathfrak{p}_{||\Pi||}(\Pi_{x:A^\bullet}\ \mathsf{pr}_1(p(x)^\bullet)))$$
$$(\widehat{\mathrm{Id}}(A, a, b))^\bullet :\equiv (||\mathrm{Id}_{A^\bullet}(a^\bullet, b^\bullet)|| \ , \ \mathfrak{p}_{||\ ||}(\mathrm{Id}^\bullet_A(a^\bullet, b^\bullet)))$$

$$(\bot)^\bullet :\equiv ||\mathbf{0}||$$
$$(\mathsf{r}_0(c))^\bullet :\equiv \mathsf{ind}_0(c^\bullet)$$
$$\mathsf{pr}_\mathsf{P}((\bot)^\bullet) :\equiv \mathfrak{p}_\bot$$
$$\mathsf{pr}_\mathsf{S}((\bot)^\bullet) :\equiv \mathfrak{s}_{coe}((\bot)^\bullet, \mathsf{pr}_\mathsf{P}((\bot)^\bullet))$$

$$(A \wedge B)^\bullet :\equiv ||A^\bullet \times B^\bullet||$$
$$(\langle a, {}_\wedge\, b\rangle)^\bullet :\equiv (a^\bullet, {}_\wedge\, b^\bullet)$$
$$\mathsf{pr}_\mathsf{P}((A \wedge B)^\bullet) :\equiv \mathfrak{p}_\wedge(A^\bullet, B^\bullet)$$
$$(\pi_i(c))^\bullet :\equiv \mathsf{pr}_{i\,\wedge}(c^\bullet)(\text{with } i \in \{1, 2\})$$
$$\mathsf{pr}_\mathsf{S}((A \wedge B)^\bullet) :\equiv \mathfrak{s}_{coe}((A \wedge B)^\bullet, \mathsf{pr}_\mathsf{P}((A \wedge B)^\bullet))$$

$$(A \to B)^\bullet :\equiv ||A^\bullet \to B^\bullet||$$
$$(\lambda_\to x.b)^\bullet :\equiv \lambda_\to x.b^\bullet$$
$$\mathsf{pr}_\mathsf{P}((A \to B)^\bullet) :\equiv \mathfrak{p}_\supset(A^\bullet, B^\bullet)$$
$$(\mathsf{Ap}_\to(f, a))^\bullet :\equiv f^\bullet_\to(a^\bullet)$$
$$\mathsf{pr}_\mathsf{S}((A \to B)^\bullet) :\equiv \mathfrak{s}_{coe}((A \to B)^\bullet, \mathsf{pr}_\mathsf{P}((A \to B)^\bullet))$$

$$
\begin{aligned}
(\forall_{x\in A}B(x))^\bullet &:\equiv ||\Pi_{x:A^\bullet}\ B(x)^\bullet|| \\
(\lambda_\forall x.b(x))^\bullet &:\equiv \lambda_\forall x.b(x)^\bullet \\
\mathsf{pr_P}((\forall_{x\in A}B(x))^\bullet) &:\equiv \mathfrak{p}_\forall(A^\bullet, \lambda x : A^\bullet.B(x)^\bullet) \\
(\mathsf{Ap}_\forall(f,a))^\bullet &:\equiv f^\bullet_\forall(a^\bullet) \\
\mathsf{pr_S}((\forall_{x\in A}B(x))^\bullet) &:\equiv \mathfrak{s}_{coe}((\forall_{x\in A}B(x))^\bullet,\ \mathsf{pr_P}((\forall_{x\in A}B(x))^\bullet)) \\
(\mathrm{Id}(A,a,b))^\bullet &:\equiv ||\mathrm{Id}_{A^\bullet}(a^\bullet,b^\bullet)|| \\
(\mathsf{id_A}(a))^\bullet &:\equiv |\mathsf{refl}_{a^\bullet}| \\
(\mathrm{El}_{\mathrm{Id}}(p,c))^\bullet &:\equiv \mathsf{ind}_{||\ ||}(p^\bullet, z.\mathsf{ind}_{\mathrm{Id}}(z, x.c(x)^\bullet)) \\
\mathsf{pr_P}((\mathrm{Id}(A,a,b))^\bullet) &:\equiv \mathfrak{p}_{||\ ||}(A^\bullet, a^\bullet, b^\bullet, \mathrm{Id}_{A^\bullet}(a^\bullet,b^\bullet)) \\
\mathsf{pr_S}((\mathrm{Id}(A,a,b))^\bullet) &:\equiv \mathfrak{s}_{coe}((\mathrm{Id}(A,a,b))^\bullet,\ \mathsf{pr_P}((\mathrm{Id}(A,a,b))^\bullet))
\end{aligned}
$$

It is possible to prove prove the corresponding versions of lemma 4.3.21 and theorem 4.3.22 for the alternative interpretation $(-)^\bullet$:

Lemma 4.3.25. *If A is a raw-type in* **mTT**, *b is a* **mTT** *raw-term and x is a variable occurring free in A, then*

$$(A[b/x])^\bullet \ :\equiv\ A^\bullet[b^\bullet/x^\bullet]$$

If a and b are **mTT** *raw-terms and x is a variable occurring free in a, then*

$$(a[b/x])^\bullet \ :\equiv\ a^\bullet[b^\bullet/x^\bullet]$$

Theorem 4.3.26. *If $\mathcal{J}$ is a derivable judgement in* **mTT**, *then $\mathcal{J}^\bullet$ is a derivable judgement in* **HoTT**.

Proof. In general, the proof follows the pattern of the proof of Thm.4.3.22 and many cases are literally the same. The cases that are different are those concerning small propositions and propositions, since their interpretation has been modified. I will illustrate only few cases, because the others can be dealt with essentially in the same way.
The rule for the decoding operator τ is validated by the interpretation, since the premise is interpreted as $p^\bullet : \mathsf{Prop}_{\mathcal{U}_0}[\Gamma^\bullet]$ and then we get $||\mathsf{pr}_1(p^\bullet)|| : \mathcal{U}_0\ [\Gamma^\bullet]$ with $\mathsf{pr}_2(p^\bullet) : \mathsf{isProp}(||\mathsf{pr}_1(p^\bullet)||)\ [\Gamma^\bullet]$.

As example, let us consider the rule (Pr4). By induction hypothesis, we have $p^\bullet : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^\bullet]$ and $q^\bullet : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^\bullet]$. Then, we can derive

$$||\mathsf{pr}_1(p^\bullet)|| : \mathcal{U}_0\ [\Gamma^\bullet] \text{ with } \mathsf{pr}_2(p^\bullet) : \mathsf{isProp}(||\mathsf{pr}_1(p^\bullet)||)\ [\Gamma^\bullet]$$

and

$$||\mathsf{pr}_1(q^\bullet)|| : \mathcal{U}_0\ [\Gamma^\bullet] \text{ with } \mathsf{pr}_2(q^\bullet) : \mathsf{isProp}(||\mathsf{pr}_1(q^\bullet)||)\ [\Gamma^\bullet]$$

Therefore, we can conclude $||\mathsf{pr}_1(p^\bullet) \times \mathsf{pr}_1(q^\bullet)|| : \mathcal{U}_0\ [\Gamma^\bullet]$ with

$$\mathfrak{p}_{||\times||}(\mathsf{pr}_1(p^\bullet), \mathsf{pr}_1(q^\bullet)) : \mathsf{isProp}(||\mathsf{pr}_1(p^\bullet) \times \mathsf{pr}_1(q^\bullet)||)\ [\Gamma^\bullet].$$

The associated judgemental equality (eq-Pr4) is trivially validated as follows: $(\tau(p \widehat{\wedge} q)\ [\Gamma])^\bullet \ :\equiv\ ||\mathsf{pr}_1((p \widehat{\wedge} q)^\bullet)|| : \mathcal{U}_0\ [\Gamma^\bullet]$. But

$$||\mathsf{pr}_1((p \widehat{\wedge} q)^\bullet)|| \equiv ||\mathsf{pr}_1(p^\bullet) \times \mathsf{pr}_1(q^\bullet)||$$

and hence (eq-Pr4) is trivially valid.
Let us consider the case of conjunction to show how propositions are interpreted:

($\wedge$-F) By induction hypothesis, we get

$$||A^\bullet|| : \mathcal{U}_1\ [\Gamma^\bullet] \text{ with } \mathsf{pr}_\mathsf{P}(||A^\bullet||) : \mathsf{isProp}(||A^\bullet||)\ [\Gamma^\bullet]$$

and

$$||B^\bullet|| : \mathcal{U}_1\ [\Gamma^\bullet] \text{ with } \mathsf{pr}_\mathsf{P}(||B^\bullet||) : \mathsf{isProp}(||B^\bullet||)\ [\Gamma^\bullet]$$

Then $||A^\bullet \times B^\bullet|| : \mathcal{U}_1\ [\Gamma^\bullet]$ such that

$$\mathsf{pr}_\mathsf{P}(||A^\bullet \times B^\bullet||) \equiv \mathfrak{p}_{||\times||}(A^\bullet, B^\bullet) : \mathsf{isProp}(||A^\bullet \times B^\bullet||)\ [\Gamma^\bullet].$$

($\wedge$-I) By induction hypothesis, we get $|a^\bullet| : ||A^\bullet||\ [\Gamma^\bullet]$ and $|b^\bullet| : ||B^\bullet||\ [\Gamma^\bullet]$. Then, we can conclude $(a^\bullet,_\wedge b^\bullet) \equiv |(a^\bullet, b^\bullet)| : ||A^\bullet \times B^\bullet||\ [\Gamma^\bullet]$.

($\wedge$-E$_1$) By induction hypothesis, $c^\bullet : ||A^\bullet \times B^\bullet||\ [\Gamma^\bullet]$. Then, we can conclude $\mathsf{pr}_{1_\wedge}(c^\bullet) \equiv \mathsf{ind}_{||\ ||}(c^\bullet, x.y.\mathsf{pr}_1(x,y)) : A^\bullet\ [\Gamma^\bullet]$.

($\wedge$-E$_2$) It is similar to the previous case.

Furthermore, the two computation rules associated to the conjunction are validated by definition of the interpretation and by Lemma 4.3.15.

□

4.3.4 Canonical Isomorphisms

In extensional type theory, the availability of the reflection rule as elimination rule for propositional equality makes valid some judgemental equalities that do not hold in the intensional version. Furthermore, we know that this rule collapses the distinction between judgemental and propositional equality and thus makes fundamental properties such as the decidability of type checking fail. Therefore, we do not have access to a genuine intensional notion of equality with the assumption of the reflection rule and it follows from what we have said so far that there is no straightforward way of interpreting the fully extensional theory into a theory without the reflection rule. In particular, this implies that there is no translation of **emTT** into **HoTT** as direct as that from **mTT** to **HoTT**.
Informally, an application of the reflection rule consists in deleting a proof-term: we can conclude $a \equiv b : A$ $[\Gamma]$ from $p : \mathrm{Id}_A(a, b)$ $[\Gamma]$. In this sense, the correspondence between the derivation and the information encoded in the proof-terms is disentangled. The naive solution would amount to restoring the proof-terms deleted by the reflection rule making them explicit. A judgemental equality derived in the extensional context would be interpreted as follows:

$$[\![\Gamma \vdash_E a \equiv b : A]\!] :\equiv \text{there exists } p \text{ such that } \Gamma \vdash_I p : \mathrm{Id}_A(a, b)$$

Following this pattern, the conversion rule

$$\frac{\Gamma \vdash_E A \equiv B \qquad \Gamma \vdash_E a : A}{\Gamma \vdash_E a : B}\,(\mathsf{conv})$$

should be interpreted as:

$$[\![\Gamma \vdash_E A \equiv B]\!] :\equiv \text{there exists } p \text{ such that } \Gamma \vdash_I p : \mathrm{Id}_{\mathcal{U}}(A, B)$$

$$[\![\Gamma \vdash_E a : A]\!] :\equiv \Gamma \vdash_I a : A$$

Therefore

$$[\![\Gamma \vdash_E a : B]\!] :\equiv \Gamma \vdash_I p_*(a) : B$$

and hence the interpretation would involve an application of transport.

The naive approach outlined above fails to provide a well-defined, coherent interpretation of the extensional type theory. The main problem lies in the fact that the interpretation would depend on the derivation of the new proof-terms that are inserted in the translation process. Since type expressions may depend on these proof-terms, then the same expression may have different interpretation values (in general, as many as the possible derivations). Then, the result is that the interpretation is not well-defined.
A possible way-out is provided by the introduction of a thin class of isomorphisms, called *canonical*, which were introduced in [89]. There, they were introduced in the context of the interpretation of **emTT** into **mTT**. A similar class of isomorphism was also considered by Hofmann in [65].

Definition 4.3.27. Let $A, B : \mathcal{U}_i$ be h-sets under a context Γ. An indexed isomorphism $\mu_A^B : A \to B\ [\Gamma]$ is a map from the h-set A to the h-set B which satisfies

$$\Pi_{(x:A)}\mathrm{Id}_A((\mu_A^B)^{-1}(\mu_A^B(x)), x) \ \times\ \Pi_{(y:B)}\mathrm{Id}_B(\mu_A^B((\mu_A^B)^{-1}(y)), y)$$

where $(\mu_A^B)^{-1}$ is the inverse of μ_A^B going from B to A.

Remark 4.3.28. Let μ_A^B, ν_A^B be (indexed) isomorphisms from A to B such that for any $x : A$, $\mu_A^B(x) =_B \nu_A^B(x)$, then $\mu_A^B = \nu_A^B$ immediately follows by function extensionality.

Definition 4.3.29. Given a dependent type $\Gamma \vdash A : \mathcal{U}$, we define the notion of *transport operation* by induction over the length of Γ:

1. If $\Gamma \ :\equiv \cdot$, then there are no transport operations;
2. If $\Gamma \ :\equiv \Delta, x : B$ and $A \ :\equiv \ C(x)$ then a *transport operation* is defined as
$$\mathsf{trp}^1(p, -) : C(x) \to C(x')\ [\Delta, x : B, x' : B, p : x =_B x']$$
where $\mathsf{trp}^1(p, -) \ :\equiv \mathsf{transport}^C(p, -)$ as given in Lemma 1.3.2.
3. If $\Gamma \ :\equiv \Delta, x : B, y_1 : D_1, \ldots, y_n : D_n\ (1 \leqslant n)$ and $A \ :\equiv \ C(x, y_1, \ldots, y_n)$ then a *transport operation* is defined as
$$\mathsf{trp}^{n+1}(p, -) : C(x, y_1, \ldots, y_n) \to C(x', \mathsf{trp}^1(p, y_1), \ldots \mathsf{trp}^n(p, y_n))$$
$$[\Delta, x : B, y_1 : D_1, \ldots, y_n : D_n, x' : B, p : x =_B x']$$
where $\mathsf{trp}^{n+1}(p, -) \ :\equiv \ \mathsf{ind}_{\mathrm{Id_B}}(p, x.(\lambda w.w))$ is defined by eliminating in
$$C(x, y_1, \ldots, y_n) \ \to \ C(x', \mathsf{trp}^1(p, y_1), \ldots, \mathsf{trp}^n(p, y_n))$$

Remark 4.3.30. When we are dealing with h-sets, $\mathsf{trp}(p, -)$ is 'natural' in p, since any two identity proofs are equal. Moreover, I recall from [155] that transport gives rise to an equivalence, where the quasi-inverse is induced by the inverse path.

In the following, I give a definition of canonical isomorphisms between dependent h-sets. Such a definition is meant to generalize the notion of transport between types dependent on equal elements. It must be stressed that canonical isomorphisms do not coincide with all the isomorphisms, because they are assumed to preserve canonical elements. Among the properties enjoyed by canonical isomorphisms, there is the fact that identities are canonical isomorphisms. However, the assumption that for each type in $\mathcal{U}_0$ and $\mathcal{U}_1$ the associated identity map is a canonical isomorphism yields a contradiction in presence of the Univalence Axiom. For this reason, an *inductive* (non-univalent) universe of h-sets, called Set_{mf}, is defined, which will be used to interpret **emTT**-sets.

Definition 4.3.31. Let Set_{mf} be the type inductively generated from the following clauses:

- If $A :\equiv \mathsf{Prop}_{\mathcal{U}_0}$, or $A :\equiv \mathbf{0}$, or $A :\equiv \mathbf{1}$, or $A :\equiv \mathbb{N}$ then $A : Set_{mf}\ [\Gamma]$ for any context Γ.
- $||B|| : Set_{mf}\ [\Gamma]$ for any type $B : \mathcal{U}_1\ [\Gamma]$.
- $\Sigma_{x:B}\ C(x) : Set_{mf}\ [\Gamma]$ for any $B : Set_{mf}\ [\Gamma]$ and $C(x) : Set_{mf}\ [\Gamma, x : B]$.
- $\Pi_{x:B}\ C(x) : Set_{mf}\ [\Gamma]$ for any $B : Set_{mf}\ [\Gamma]$ and $C(x) : Set_{mf}\ [\Gamma, x : B]$.
- $B + C : Set_{mf}\ [\Gamma]$ for any $B : Set_{mf}\ [\Gamma]$ and $C : Set_{mf}\ [\Gamma]$.
- $\mathsf{List}(B) : Set_{mf}\ [\Gamma]$ for any $B : Set_{mf}\ [\Gamma]$.
- $B/R : Set_{mf}\ [\Gamma]$ for any h-set $B : \mathcal{U}_0\ [\Gamma]$ and an equivalence relation $R(x, y) : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma, x : B, y : B]$ such that $B : Set_{mf}\ [\Gamma]$ and $R(x, y) : Set_{mf}\ [\Gamma, x : B, y : B]$.

Remark 4.3.32. Formally the universe Set_{mf} is given as an inductive-recursive definition and hence it can be defined by extending **HoTT** with induction-recursion. However, it is important to note that for our purposes it suffices to define this universe in the meta-theory. This approach is chosen

because the univalent universe $\mathcal{U}_1$ lacks an inductive elimination, which would be contradictory with the Univalence Axiom.
Furthermore, Set_{mf} has been presented as a Russell-universe to make the notation smoother. However, it would be easy to present it as a Tarski-universe where the elements are *codes* for the selected type formers and there is a decoding map which sends a code to the corresponding element in the univalent universe $\mathcal{U}_1$.

The introduction of the inductive universe of h-sets Set_{mf} is instrumental to give the following definition of the type of canonical isomorphisms between h-sets contained in Set_{mf}:

Definition 4.3.33 (Canonical Isomorphisms)**.** The type of indexed canonical isomorphisms $\mu^{A_2}_{A_1} : A_1 \to A_2\ [\Gamma]$, where $A_1, A_2 : Set_{mf}$, is inductively generated as follows:

- If $A :\equiv \mathsf{Prop}_{\mathcal{U}_0}$, or $A :\equiv \mathbf{0}$, or $A :\equiv \mathbf{1}$, or $A :\equiv \mathbb{N}$, then the identity morphism $\mathsf{id}^A_A :\equiv \lambda x.x : A \to A\ [\Gamma]$ is a canonical isomorphism, that is trivially an isomorphism and the inverse $\mu^{A^{-1}}_A$ is also given by the identity.

- If $A_1 :\equiv ||B_1|| : Set_{mf}$ and $A_2 :\equiv ||B_2|| : Set_{mf}$, then any isomorphism $\mu^{||B_2||}_{||B_1||} : ||B_1|| \to ||B_2||\ [\Gamma]$ is canonical and the chosen inverse is denoted with $\mu^{A_2{}^{-1}}_{A_1}$.

- If $A_1 :\equiv \Sigma_{x:B_1}\ C_1(x)\ [\Gamma]$ and $A_2 :\equiv \Sigma_{x':B_2}\ C_2(x')\ [\Gamma]$ and $\mu^{B_2}_{B_1} : B_1 \to B_2\ [\Gamma]$ and $\mu^{C_2(\mu^{B_2}_{B_1}(x))}_{C_1(x)} : C_1(x) \to C_2(\mu^{B_2}_{B_1}(x))\ [\Gamma, x : B_1]$ are canonical isomorphisms, then any map

$$\mu^{\Sigma_{x':B_2}\ C_2(x')}_{\Sigma_{x:B_1}\ C_1(x)} : \Sigma_{x:B_1}\ C_1(x) \to \Sigma_{x':B_2}\ C_2(x')\ [\Gamma]$$

  such that

$$\mu^{\Sigma_{x':B_2}\ C_2(x')}_{\Sigma_{x:B_1}\ C_1(x)}(z) = (\mu^{B_2}_{B_1}(\mathsf{pr}_1(z)), \mu^{C_2(\mu^{B_2}_{B_1}(\mathsf{pr}_1(z)))}_{C_1(\mathsf{pr}_1(z))}(\mathsf{pr}_2(z)))$$

  for any $z : \Sigma_{x:B_1}\ C_1(x)$, is a canonical isomorphism with the inverse given by

$$\mu_{A_1}^{A_2}{}^{-1} \;:\equiv\; \lambda z.(\, \mu_{B_1}^{B_2}{}^{-1}(\mathsf{pr}_1(z))\,,\, (\mu^{C_2(\mu_{B_1}^{B_2}(\mu_{B_1}^{B_2}{}^{-1}(\mathsf{pr}_1(z))))}_{C_1(\mu_{B_1}^{B_2}{}^{-1}(\mathsf{pr}_1(z)))})^{-1} \circ \mathsf{trp}(p_\mu{}^{-1}, -)(\mathsf{pr}_2(z))\,)$$

where $p_\mu : \mu_{B_1}^{B_2}(\mu_{B_1}^{B_2}{}^{-1}(\mathsf{pr}_1(z))) =_{B_2} \mathsf{pr}_1(z)$ and provided that $\mu_{B_1}^{B_2}$ and $\mu_{C_1(x)}^{C_2(\mu_{B_1}^{B_2}(x))}$ come equipped with inverses $\mu_{B_1}^{B_2}{}^{-1}$ and $\mu_{C_1(x)}^{C_2(\mu_{B_1}^{B_2}(x))}{}^{-1}$ respectively.

- If $A_1 \;:\equiv\; \Pi_{x:B_1}\, C_1(x)\; [\Gamma]$ and $A_2 \;:\equiv\; \Pi_{x':B_2}\, C_2(x')\; [\Gamma]$ and $\mu_{B_1}^{B_2} : B_1 \to B_2\; [\Gamma]$ and $\mu_{C_1(x)}^{C_2(\mu_{B_1}^{B_2}(x))} : C_1(x) \;\to\; C_2(\mu_{B_1}^{B_2}(x))\; [\Gamma, x : B_1]$ are canonical isomorphisms, then any map

$$\mu^{\Pi_{x':B_2}\, C_2(x')}_{\Pi_{x:B_1}\, C_1(x)} : \Pi_{x:B_1}\, C_1(x) \to \Pi_{x':B_2}\, C_2(x')\; [\Gamma]$$

such that

$$\mu^{\Pi_{x':B_2}\, C_2(x')}_{\Pi_{x:B_1}\, C_1(x)}(f) \;=\; \lambda x' : B_2.(\, \mathsf{trp}(p_\mu\,, -) \circ \mu^{C_2(\mu_{B_1}^{B_2}(\mu_{B_2}^{B_2}{}^{-1}(x')))}_{C_1(\mu_{B_1}^{B_2}{}^{-1}(x'))})(f(\mu_{B_1}^{B_2}{}^{-1}(x')))$$

is a canonical isomorphism, for any $f : \Pi_{x:B_1}\, C_1(x)$ and for any $p_\mu : \mu_{B_1}^{B_2}(\mu_{B_2}^{B_2}{}^{-1}(x')) =_{B_2} x'$,
where the body after the lambda is the arrow

$$C_1(\mu_{B_2}^{B_2}{}^{-1}(x')) \xrightarrow{\;\alpha\;} C_2(\mu_{B_1}^{B_2}(\mu_{B_2}^{B_2}{}^{-1}(x'))\,) \xrightarrow{\;\mathsf{trp}(p_\mu, -)\;} C_2(x')$$

applied to the value $f(\mu_{B_1}^{B_2}{}^{-1}(x'))$ and α is just an abbreviation for the canonical isomorphism $\mu^{C_2(\mu_{B_1}^{B_2}(\mu_{B_2}^{B_2}{}^{-1}(x')))}_{C_1(\mu_{B_1}^{B_2}{}^{-1}(x'))}$. The inverse is given by

$$(\mu_{A_1}^{A_2})^{-1} \;=\; \lambda f'.\lambda x : B_1.(\, (\mu^{C_2(\mu_{B_1}^{B_2}(x))}_{C_1\,(x)})^{-1}(f'(\mu_{B_1}^{B_2}(x)))\,)$$

provided that $\mu_{B_1}^{B_2}$ and $\mu_{C_1(x)}^{C_2(\mu_{B_1}^{B_2}(x))}$ come equipped with inverses $\mu_{B_1}^{B_2}{}^{-1}$ and $\mu_{C_1(x)}^{C_2(\mu_{B_1}^{B_2}(x))}{}^{-1}$ respectively.

- If $A_1 \;:\equiv\; B_1 + C_1$ and $A_2 \;:\equiv\; B_2 + C_2$ and $\mu_{B_1}^{B_2} : B_1 \to B_2\ [\Gamma]$ and $\mu_{C_1}^{C_2} : C_1 \to C_2\ [\Gamma]$ are canonical isomorphisms, then any map

$$\mu_{B_1+C_1}^{B_2+C_2} : B_1 + C_1 \to B_2 + C_2\ [\Gamma]$$

such that

$$\mu_{B_1+C_1}^{B_2+C_2}(z) \;=\; \mathsf{ind}_+(z, z_0.\mathsf{inl}(\mu_{B_1}^{B_2}(z_0)), z_1.\mathsf{inr}(\mu_{C_1}^{C_2}(z_1)))$$

for any $z : B_1 + C_1$, is a canonical isomorphism with the inverse given by

$$\mu_{A_1}^{A_2{}^{-1}} \;=\; \lambda z.\mathsf{ind}_+(z, z_0.\mathsf{inl}(\mu_{B_1}^{B_2{}^{-1}}(z_0)), z_1.\mathsf{inr}(\mu_{C_1}^{C_2{}^{-1}}(z_1)))$$

provided that $\mu_{B_1}^{B_2}$ and $\mu_{C_1}^{C_2}$ come equipped with inverses $\mu_{B_1}^{B_2{}^{-1}}$ and $\mu_{C_1}^{C_2{}^{-1}}$.

- If $A_1 \;:\equiv\; \mathsf{List}(B_1)$ and $A_2 \;:\equiv\; \mathsf{List}(B_2)$ and $\mu_{B_1}^{B_2} : B_1 \to B_2\ [\Gamma]$ is a canonical isomorphism, then any map

$$\mu_{\mathsf{List}(B_1)}^{\mathsf{List}(B_2)} : \mathsf{List}(B_1) \to \mathsf{List}(B_2)\ [\Gamma]$$

such that

$$\mu_{\mathsf{List}(B_1)}^{\mathsf{List}(B_2)}(z) \;=\; \mathsf{ind}_{\mathsf{List}}(z, \epsilon, (x, y, u).\mathsf{cons}(u, \mu_{B_1}^{B_2}(y))$$

for any $z : \mathsf{List}(B_1)$, is a canonical isomorphism with the inverse given by

$$\mu_{A_1}^{A_2{}^{-1}} \;=\; \lambda z.\mathsf{ind}_{\mathsf{List}}(z, \epsilon, (x, y, z).\mathsf{cons}(z, \mu_{B_1}^{B_2{}^{-1}}(y))$$

provided that $\mu_{B_1}^{B_2}$ comes equipped with inverse $\mu_{B_1}^{B_2{}^{-1}}$.

- If $A_1 \;:\equiv\; B_1/R_1$ and $A_2 \;:\equiv\; B_2/R_2$ and $\mu_{B_1}^{B_2} : B_1 \;\to\; B_2\ [\Gamma]$ is a canonical isomorphism and $R_1(x, y) \;\leftrightarrow\; R_2(\mu_{B_1}^{B_2}(x)\,, \mu_{B_1}^{B_2}(y))\ [\Gamma, x : B_1, y : B_1]$ holds, then any map

$$\mu_{B_1/R_1}^{B_2/R_2} \;:\; B_1/R_1 \;\to\; B_2/R_2\ [\Gamma]$$

such that

$$\mu^{B_2/R_2}_{B_1/R_1}(z) \;=\; \mathsf{ind}_Q(z, x.\mu^{B_2}_{B_1}(x))$$

for any $z : B_1/R_1$, is a canonical isomorphism with the inverse given by

$$\mu^{A_2}_{A_1}{}^{-1} \;=\; \lambda z.\mathsf{ind}_Q(z, x.\mu^{B_2}_{B_1}{}^{-1}(x))$$

provided that $\mu^{B_2}_{B_1}$ comes equipped with an inverse $\mu^{B_2}_{B_1}{}^{-1}$.

Canonical isomorphisms enjoy a set of properties which will turn out to be fundamental to interpret correctly **emTT**. First, they are closed under substitution:

Lemma 4.3.34 (Closure under Substitution). *If $\mu^B_A : A \to B\ [\Gamma]$ is a canonical isomorphism and $\Gamma :\equiv\ \Delta, x : E, y_1 : C_1, \ldots, y_n : C_n$, then the map*

$$\mu^B_A[e/x, z_1/y_1, \ldots, z_n/y_n]$$

from

$$A[e/x, z_1/y_1, \ldots, z_n/y_n]$$

to

$$B[e/x, z_1/y_1, \ldots, z_n/y_n]$$

under the context

$$[\ \Delta, e : E, z_1 : C_1[e/x], \ldots, z_n : C_n[e/x, z_1/y_1, \ldots, z_{n-1}/y_{n-1}]\ ]$$

is a canonical isomorphism.

Proof. The proof is by structural induction over the definition of canonical isomorphism. □

We can show that any h-set $A\ [\Gamma]$ of **HoTT** in Set_{mf} is equipped with *canonical* transports:

Lemma 4.3.35. *Let $A : Set_{mf}\ [\Gamma]$ be a type in* **HoTT**. *Then A has canonical transport operations.*

Proof. By induction over the structure of A. Without loss of generality, we consider only the case $\mathsf{trp}^1(p, -)$, since the n-ary case is a straightforward extension.

- If A is ground type, then the only transport operations are the identities that are canonical according to Def.4.3.33.

- If $A :\equiv \ ||B||$ and $\Gamma :\equiv \ \Delta, x : E$, then transport operations are canonical since they are isomorphisms and any isomorphism between h-propositions is canonical by Def.4.3.33.

- If $A :\equiv \ \Sigma_{(y:B)}C(y)$ and $\Gamma :\equiv \ \Delta, x : E$, then

$$\mathsf{trp}(p, -) : A \to A[x'/x] \ [\Delta, x : E, x' : E, p : x =_E x']$$

satisfies, for any $z : \Sigma_{(y:B)}C(y)$,

$$\mathsf{trp}(p, z) = (\mathsf{trp}(p, \mathsf{pr}_1(z)), \mathsf{trp}(p, \mathsf{pr}_2(z))$$

which follows by Id-elimination, is canonical by Def.4.3.33, since the transport operations of the components B and $C(y)$ are canonical by inductive hypothesis.

- If $A :\equiv \ \Pi_{(y:B)}C(y)$ and $\Gamma :\equiv \ \Delta, x : E$, then

$$\mathsf{trp}(p, -) : A \to A[x'/x] \ [\Delta, x : E, x' : E, p : x =_E x']$$

satisfies, for any $f : \Pi_{(y:B)}C(y)$,

$$\mathsf{trp}(p, f) = \lambda y'.\mathsf{trp}^C(p, f(\mathsf{trp}^B(p^{-1}, y')))$$

and is canonical by Def.4.3.33, since the transport operations of B and $C(y)$ are canonical by inductive hypothesis.

- If $A :\equiv B + C$ and $\Gamma :\equiv \ \Delta, x : E$, then

$$\mathsf{trp}(p, -) : A \to A[x'/x] \ [\Delta, x : E, x' : E, p : x =_E x']$$

satisfies, for any $z : B + C$,

$$\mathsf{trp}(p, z) = \mathsf{ind}_+(z, z_0.\mathsf{inl}(\mathsf{trp}(p, z_0)), z_1.\mathsf{inr}(\mathsf{trp}(p, z_1)))$$

and is canonical by Def.4.3.33, since the transport operations of B and C are canonical by inductive hypothesis.

- If $A :\equiv \mathsf{List}(B)$ and $\Gamma :\equiv\ \Delta, x : E$, then

$$\mathsf{trp}(p, -) : A \to A[x'/x]\ [\Delta, x : E, x' : E, p : x =_E x']$$

satisfies, for any $z : \mathsf{List}(B)$,

$$\mathsf{trp}(p, z) = \mathsf{ind}_{\mathsf{List}}(z, \epsilon, (x, y, z).\mathsf{cons}(z, \mathsf{trp}(p, y))$$

and is canonical by Def.4.3.33, since the transport operation of B is canonical by inductive hypothesis.

- If $A :\equiv B/R$ and $\Gamma :\equiv\ \Delta, x : E$, then

$$\mathsf{trp}(p, -) : A \to A[x'/x]\ [\Delta, x : E, x' : E, p : x =_E x']$$

satisfies, for any $z : B/R$,

$$\mathsf{trp}(p, z) = \mathsf{ind}_Q(z, w.[\mathsf{trp}(p, w)])$$

and is canonical by Def.4.3.33, since the transport operation of B is canonical by inductive hypothesis.

□

Corollary 4.3.36. *For any transport operation* $\mathsf{trp}(p, -)$*, its inverse given by* $\mathsf{trp}(p^{-1}, -)$ *is a canonical isomorphism as well.*

The following proposition states that an identity morphism is always a canonical isomorphism:

Proposition 4.3.37 (**Identities are canonical**). *For any h-set* $A : \mathcal{U}_1\ [\Gamma]$ *in* Set_{mf}*, the map* $\mathsf{id}_A : A \to A\ [\Gamma]$ *is a canonical isomorphism.*

Proof. The proof is by structural induction over A.

- If A is a ground type, then the statement is trivially valid according to Def.4.3.33.

- If $A\ :\equiv\ ||B||$, then that id_A is a canonical isomorphism is immediate, since the identity map is an isomorphism.

- If $A :\equiv \Sigma_{x:B}C(x)$, then by induction hypothesis id_B and $\mathsf{id}_{C(x)}$ are canonical isomorphisms and thus, for any $z : \Sigma_{x:B}C(x)$,

$$\nu_{\Sigma_{x:B}C(x)}(z) :\equiv (\mathsf{id}_B(\mathsf{pr}_1(z)), \mathsf{id}^{C(\mathsf{id}_B(\mathsf{pr}_1(z)))}_{C(\mathsf{pr}_1(z))}(\mathsf{pr}_2(z)))$$

But

$$(\mathsf{id}_B(\mathsf{pr}_1(z)), \mathsf{id}^{C(\mathsf{id}_B(\mathsf{pr}_1(z)))}_{C(\mathsf{pr}_1(z))}(\mathsf{pr}_2(z))) \equiv (\mathsf{pr}_1(z), \mathsf{pr}_2(z))$$

and $(\mathsf{pr}_1(z), \mathsf{pr}_2(z)) = z$. Therefore,

$$\nu_{\Sigma_{x:B}C(x)}(z) = \mathsf{id}_{\Sigma_{x:B}C(x)}(z)$$

which means that $\mathsf{id}_{\Sigma_{x:B}C(x)}$ is a canonical isomorphism, since by induction hypothesis its components are canonical.

- If $A :\equiv \Pi_{x:B}C(x)$, then by induction hypothesis id_B and $\mathsf{id}_{C(x)}$ are canonical isomorphisms. Then, for any $f : \Pi_{x:B}C(x)$,

$$\nu_{\Pi_{x:B}C(x)}(f) :\equiv \lambda x : B.(\mathsf{trp}(p_{\mathsf{id}}, -) \circ \mathsf{id}^{C(\mathsf{id}_B({\mathsf{id}_B}^{-1}(x)))}_{C({\mathsf{id}_B}^{-1}(x))})(f({\mathsf{id}_B}^{-1}(x)))$$

where $p : \mathsf{id}_B({\mathsf{id}_B}^{-1}(x)) = x$. But then this is equal to

$$\lambda x : B.\mathsf{id}^{C(x)}_{C(x)}(f(x)) = \lambda x : B.f(x) = f$$

Therefore,

$$\nu_{\Pi_{x:B}C(x)} = \mathsf{id}_{\Pi_{x:B}C(x)}$$

that implies that the latter is canonical, since its components are canonical by inductive hypothesis.

- If $A :\equiv B + C$, then by induction hypothesis id_B and id_C are canonical isomorphisms. Therefore, for any $z : B + C$,

$$\nu_{B+C}(z) :\equiv \mathsf{ind}_+(z, z_0.\mathsf{inl}(\mathsf{id}_B(z_0)), z_1.\mathsf{inr}(\mathsf{id}_C(z_1)))$$

But

$$\mathsf{ind}_+(z, z_0.\mathsf{inl}(\mathsf{id}_B(z_0)), z_1.\mathsf{inr}(\mathsf{id}_C(z_1))) \equiv \mathsf{ind}_+(z, z_0.\mathsf{inl}(z_0), z_1.\mathsf{inr}(z_1))$$

and $\mathsf{ind}_+(z, z_0.\mathsf{inl}(z_0), z_1.\mathsf{inr}(z_1)) = z$. Therefore,

$$\nu_{B+C}(z) = \mathsf{id}_{B+C}(z)$$

that implies that the latter is a canonical isomorphism.

- If $A :\equiv \mathsf{List}(B)$, then by induction hypothesis id_B is a canonical isomorphism. Then, for any $z : \mathsf{List}(B)$,

$$\nu_{\mathsf{List}(B)}(z) :\equiv \mathsf{ind}_{\mathsf{List}}(z, \epsilon, x.y.u.\mathsf{cons}(u, \mathsf{id}_B(y))$$

But

$$\mathsf{ind}_{\mathsf{List}}(z, \epsilon, x.y.u.\mathsf{cons}(u, \mathsf{id}_B(y)) \equiv \mathsf{ind}_{\mathsf{List}}(z, \epsilon, x.y.u.\mathsf{cons}(u, y))$$

and $\mathsf{ind}_{\mathsf{List}}(z, \epsilon, x.y.u.\mathsf{cons}(u, y)) = z$. Therefore,

$$\nu_{\mathsf{List}(B)}(z) = \mathsf{id}_{\mathsf{List}}(z)$$

that implies that the latter is canonical.

- If $A :\equiv B/R$, then by induction hypothesis id_B is a canonical isomorphism. Then, for any $z : B/R$,

$$\nu_{B/R}(z) :\equiv \mathsf{ind}_{\mathcal{Q}}(z, x.\mathsf{id}_B(x))$$

that is judgementally equal to $\mathsf{ind}_{\mathcal{Q}}(z, x.x)$ and $\mathsf{ind}_{\mathcal{Q}}(z, x.x) = z$. Therefore,

$$\nu_{B/R}(z) = \mathsf{id}_{B/R}(z)$$

that implies that the latter is canonical.

□

Two fundamental properties of canonical isomorphisms are *uniqueness* up to propositional equality and closure under *composition*. In particular, these properties are essential to interpret correctly the judgemental equalities of **emTT** and to show the validity theorem:

Proposition 4.3.38 (Uniqueness). *For any h-sets $A_1, A_2 : \mathcal{U}_1$ $[\Gamma]$ in Set_{mf}, if $\mu_{A_1}^{A_2} : A_1 \to A_2$ $[\Gamma]$ and $\nu_{A_1}^{A_2} : A_1 \to A_2$ $[\Gamma]$ are canonical isomorphisms, then $\mu_{A_1}^{A_2}(z) =_{A_2} \nu_{A_1}^{A_2}(z)$ $[\Gamma, z : A_1]$.*

Proof. - For the ground types, the canonical isomorphisms are the identities and hence the statement is trivially valid.

- If $A_1 :\equiv ||B||$ and $A_2 :\equiv ||C||$, then for any $z : ||B||$, $\mu_{||B||}^{||C||}(z) =_{||C||} \nu_{||B||}^{||C||}(z)$, since $||C||$ is a h-proposition.

- If $A_1 :\equiv \Sigma_{x:B_1}\ C_1(x)$ and $A_2\ :\equiv\ \Sigma_{x':B_2}\ C_2(x')$, then both $\mu^{A_2}_{A_1}$ and $\nu^{A_2}_{A_1}$ are defined componentwise according to Def.4.3.33. Let us assume that the components of the first are $\mu^{B_2}_{B_1}$ and $\mu^{C_2(\mu^{B_2}_{B_1}(x))}_{C_1(x)}$ and the components of the latter are $\nu^{B_2}_{B_1}$ and $\nu^{C_2(\nu^{B_2}_{B_1}(x))}_{C_1(x)}$. Then, by induction hypothesis

$$\mu^{B_2}_{B_1}(x) =_{B_2} \nu^{B_2}_{B_1}(x)\ [\Gamma, x : B_1]$$

and

$$\mathsf{trp}(p,\ \mu^{C_2(\mu^{B_2}_{B_1}(x))}_{C_1(x)}(y)) =_{C_2(\nu^{B_2}_{B_1}(x))} \nu^{C_2(\nu^{B_2}_{B_1}(x))}_{C_1(x)}(y)$$

under the context $[\Gamma, x : B_1, y : C_1(x), p : \mu^{B_2}_{B_1}(x) =_{B_2} \nu^{B_2}_{B_1}(x)\ ]$.

Therefore,

$$\mu^{\Sigma_{x':B_2}\ C_2(x')}_{\Sigma_{x:B_1}\ C_1(x)}(z) =_{\Sigma_{x':B_2}C_2(x')} \nu^{\Sigma_{x':B_2}\ C_2(x')}_{\Sigma_{x:B_1}\ C_1(x)}(z)\ [\Gamma, z : \Sigma_{x:B_1}\ C_1(x)].$$

- If $A_1\ :\equiv\ \Pi_{x:B_1}\ C_1(x)$ and $A_2\ :\equiv\ \Pi_{x':B_2}\ C_2(x')$, then the canonical isomorphisms $\mu^{A_2}_{A_1}$ and $\nu^{A_2}_{A_1}$ are defined as follows:

$$\mu^{A_2}_{A_1} = \lambda f.\lambda x'.(\ \mathsf{trp}(p_\mu\ , -) \circ \mu^{C_2(\mu^{B_2}_{B_1}(\mu^{B_2^{-1}}_{B_1}(x')))}_{C_1(\mu^{B_2^{-1}}_{B_1}(x'))})(f(\mu^{B_2^{-1}}_{B_1}(x'))\ )$$

and

$$\nu^{A_2}_{A_1} = \lambda f.\lambda x'.(\ \mathsf{trp}(p_\nu\ , -) \circ \nu^{C_2(\nu^{B_2}_{B_1}(\nu^{B_2^{-1}}_{B_1}(x')))}_{C_1(\nu^{B_2^{-1}}_{B_1}(x'))})(f(\nu^{B_2^{-1}}_{B_1}(x'))\ )$$

where $p_\mu\ : \mu^{B_2}_{B_1}(\mu^{B_2^{-1}}_{B_1}(x')) =_{B_2} x'$ and $p_\nu\ : \nu^{B_2}_{B_1}(\nu^{B_2^{-1}}_{B_1}(x')) =_{B_2} x'$.

By induction hypothesis, for any $f : \Pi_{x:B_1}C_1(x)$ and any $x : B_1$, there exists a proof-term q of type

$$\mu^{B_2}_{B_1}(x) =_{B_2} \nu^{B_2}_{B_1}(x)$$

for $x : B_1$ and the same holds for their inverses, which are canonical by induction hypothesis: that is, there exists a proof-term q' of type $\mu^{B_2{}^{-1}}_{B_1}(z) =_{B_1} \nu^{B_2{}^{-1}}_{B_1}(z)$ for $z : B_2$.

Then, we get that the following holds

$$\mathsf{trp}(q', f(\mu_{B_1}^{B_2{}^{-1}}(z))) = f(\nu_{B_1}^{B_2{}^{-1}}(z))$$

and also

$$q'' : \mu_{B_1}^{B_2}(\mu_{B_1}^{B_2^{-1}}(z)) = \nu_{B_1}^{B_2}(\nu_{B_1}^{B_2{}^{-1}}(z))$$

Moreover, by induction hypothesis uniqueness of canonical isomorphisms from $C_1(\mu_{B_1}^{B_2{}^{-1}}(z))$ to $C_2(\nu_{B_1}^{B_2}(\nu_{B_1}^{B_2{}^{-1}}(z))$ holds and hence we get the following:

$$\nu_{C_1(\nu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\nu_{B_1}^{B_2}(\nu_{B_1}^{B_2{}^{-1}}(z)))} \circ \mathsf{trp}(q', -) = \mathsf{trp}(q'', -) \circ \mu_{C_1(\mu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\mu_{B_1}^{B_2}(\mu_{B_1}^{B_2{}^{-1}}(z)))}$$

Therefore,

$$\mathsf{trp}(p_\nu, -) \circ (\, \nu_{C_1(\nu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\nu_{B_1}^{B_2}(\nu_{B_1}^{B_2{}^{-1}}(z)))} \circ \mathsf{trp}(q', -)\,) = \mathsf{trp}(p_\nu, -) \circ (\mathsf{trp}(q'', -) \circ \mu_{C_1(\mu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\mu_{B_1}^{B_2}(\mu_{B_1}^{B_2{}^{-1}}(z)))})$$

Further, since we know that transports commute, because they are uniquely determined up to propositional equality, we get

$$\mathsf{trp}(p_\nu, -) \circ (\mathsf{trp}(q'', -) = \mathsf{trp}(p_\mu, -)$$

And then we can derive

$$\mathsf{trp}(p_\mu, -) \circ \mu_{C_1(\mu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\mu_{B_1}^{B_2}(\mu_{B_1}^{B_2{}^{-1}}(z)))} = \mathsf{trp}(p_\nu, -) \circ (\, \nu_{C_1(\nu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\nu_{B_1}^{B_2}(\nu_{B_1}^{B_2{}^{-1}}(z)))} \circ \mathsf{trp}(q', -))$$

Hence, by applying these arrows to the value $f(\mu_{B_1}^{B_2^{-1}}(z))$ and since we know that $\mathsf{trp}(q', f(\mu_{B_1}^{B_2^{-1}}(z))) = f(\nu_{B_1}^{B_2{}^{-1}}(z))$, we conclude

$$\begin{aligned}
\mu_{A_1}^{A_2}(f, z) = &\quad (\ \mathsf{trp}(p_\mu, -) \circ\ \mu_{C_1(\mu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\mu_{B_1}^{B_2}(\mu_{B_1}^{B_2{}^{-1}}(z)))}\ )(f(\mu_{B_1}^{B_2^{-1}}(z))) \\
= &\quad \mathsf{trp}(p_\nu, -) \circ (\ \nu_{C_1(\nu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\nu_{B_1}^{B_2}(\nu_{B_1}^{B_2{}^{-1}}(z)))} \circ \mathsf{trp}(q', -)\ )(f(\mu_{B_1}^{B_2^{-1}}(z))) \\
= &\quad \mathsf{trp}(p_\nu, -) \circ \nu_{C_1(\nu_{B_1}^{B_2{}^{-1}}(z))}^{C_2(\nu_{B_1}^{B_2}(\nu_{B_1}^{B_2{}^{-1}}(z)))}(f(\nu_{B_1}^{B_2{}^{-1}}(z))) \\
= &\quad \nu_{A_1}^{A_2}(f, z)\ .
\end{aligned}$$

- If $A_1 :\equiv B_1 + C_1$ and $A_2 :\equiv B_2 + C_2$, then both $\mu_{A_1}^{A_2}$ and $\nu_{A_1}^{A_2}$ are defined as in Def.4.3.33. Then, $\mu_{B_1}^{B_2}$ and $\mu_{C_1}^{C_2}$ are canonical isomorphisms as well as $\nu_{B_1}^{B_2}$ and $\nu_{C_1}^{C_2}$. By induction hypothesis,

$$\mu_{B_1}^{B_2}(x) =_{B_2} \nu_{B_1}^{B_2}(x)\ [\Gamma, x : B_1]$$

and

$$\mu_{C_1}^{C_2}(y) =_{C_2} \nu_{C_1}^{C_2}(y)\ [\Gamma, y : C_1]$$

Therefore, it follows that

$$\mu_{B_1+B_2}^{C_1+C_2}(z) =_{C_1+C_2} \nu_{B_1+B_2}^{C_1+C_2}(z)\ [\Gamma, z : B_1 + B_2]$$

- If $A_1 :\equiv \mathsf{List}(B_1)$ and $A_2 :\equiv \mathsf{List}(B_2)$, then $\mu_{A_1}^{A_2}$ and $\nu_{A_1}^{A_2}$ are both defined as in Def.4.3.33. Therefore, $\mu_{B_1}^{B_2}$ and $\nu_{B_1}^{B_2}$ are canonical isomorphisms and by induction hypothesis, for any $x : B_1$, $\mu_{B_1}^{B_2}(x) =_{B_2} \nu_{B_1}^{B_2}(x)$. Therefore, the result follows by definition of canonical isomorphisms for lists.

- If $A_1 :\equiv B_1/R_1$ and $A_2 :\equiv B_2/R_2$ then $\mu_{A_1}^{A_2}$ and $\nu_{A_1}^{A_2}$ are both defined as in Def.4.3.33. Hence, $\mu_{B_1}^{B_2}$ and $\nu_{B_1}^{B_2}$ are canonical isomorphisms and both $R_1(x, y) \leftrightarrow R_2(\mu_{B_1}^{B_2}(x), \mu_{B_1}^{B_2}(y))$ and $R_1(x, y) \leftrightarrow R_2(\nu_{B_1}^{B_2}(x), \nu_{B_1}^{B_2}(y))$ hold. Then, by induction hypothesis

$$\mu_{B_1}^{B_2}(x) =_{B_2} \nu_{B_1}^{B_2}(x)\ [\Gamma, x : B_1]$$

and

$$R_2(\mu_{B_1}^{B_2}(x), \mu_{B_1}^{B_2}(y)) \leftrightarrow R_2(\nu_{B_1}^{B_2}(x), \nu_{B_1}^{B_2}(y))\ [\Gamma, x : B_1, y : B_1]$$

Therefore, it follows that

$$\mu_{B_1/R_1}^{B_2/R_2}(z) =_{B_2/R_2} \nu_{B_1/R_1}^{B_2/R_2}(z)\ [\Gamma, z : B_1/R_1]$$

□

Proposition 4.3.39 (Closure under composition). *For any h-sets $A_1, A_2, A_3 : \mathcal{U}_1\ [\Gamma]$ in Set_{mf}, if $\mu_{A_1}^{A_2} : A_1 \to A_2\ [\Gamma]$ and $\mu_{A_2}^{A_3} : A_2 \to A_3\ [\Gamma]$ are canonical isomorphisms, then $\mu_{A_2}^{A_3} \circ \mu_{A_1}^{A_2} : A_1 \to A_3\ [\Gamma]$ is a canonical isomorphism.*

Proof. - For ground types, the composition is the identity and thus is canonical by definition.

- For truncated types, it follows from the fact that isomorphisms are closed under composition.

- If $A_1 :\equiv \Sigma_{x:B_1} C_1(x)$ and $A_2 :\equiv \Sigma_{x':B_2} C_2(x')$ and $A_3 :\equiv \Sigma_{x'':B_3} C_3(x'')$, then by definition of canonical isomorphisms for indexed sums

$$\mu^{A_2}_{A_1} :\equiv \lambda z.(\mu^{B_2}_{B_1}(\mathsf{pr}_1(z)), \mu^{C_2(\mu^{B_2}_{B_1}(\mathsf{pr}_1(z)))}_{C_1(\mathsf{pr}_1(z))}(\mathsf{pr}_2(z)))$$

and

$$\mu^{A_3}_{A_2} :\equiv \lambda z.(\mu^{B_3}_{B_2}(\mathsf{pr}_1(z)), \mu^{C_3(\mu^{B_3}_{B_2}(\mathsf{pr}_1(z)))}_{C_2(\mathsf{pr}_1(z))}(\mathsf{pr}_2(z)))$$

Then, the composition of $\mu^{A_3}_{A_2} \circ \mu^{A_2}_{A_1}$ applied to $z : \Sigma_{x:B_1} C_1(x)$ amounts to

$$\begin{aligned} \mu^{A_3}_{A_2} \circ \mu^{A_2}_{A_1}(z) &:\equiv \mu^{\Sigma_{x'':B_3} C_3(x'')}_{\Sigma_{x':B_2} C_2(x')} \circ \mu^{\Sigma_{x':B_2} C_2(x')}_{\Sigma_{x:B_1} C_1(x)}(z) \\ &:\equiv (\mu^{B_3}_{B_2}(\mu^{B_2}_{B_1}(\mathsf{pr}_1(z))), \mu^{C_3(\mu^{B_3}_{B_2}(\mu^{B_2}_{B_1}(\mathsf{pr}_1(z))))}_{C_2(\mu^{B_2}_{B_1}(\mathsf{pr}_1(z)))}(\mu^{C_2(\mu^{B_2}_{B_1}(\mathsf{pr}_1(z)))}_{C_1(\mathsf{pr}_1(z))}(\mathsf{pr}_2(z)))) \end{aligned}$$

which is a canonical isomorphism by Def 4.3.33, since the components $\mu^{B_3}_{B_2} \circ \mu^{B_2}_{B_1}$ and $\mu^{C_3}_{C_2} \circ \mu^{C_2}_{C_1}$ are canonical isomorphisms by induction hypothesis.

- If $A_1 :\equiv \Pi_{x:B_1} C_1(x)$ and $A_2 :\equiv \Pi_{x':B_2} C_2(x')$ and $A_3 :\equiv \Pi_{x'':B_3} C_3(x'')$, then by definition of canonical isomorphisms for dependent products

$$\mu^{A_2}_{A_1} :\equiv \lambda f.\lambda x' : B_2.(\mathsf{trp}(p_{\mu^{A_2}_{A_1}}, -) \circ \mu^{C_2(\mu^{B_2}_{B_1}(\mu^{B_2^{-1}}_{B_1}(x')))}_{C_1(\mu^{B_2^{-1}}_{B_1}(x'))})(f(\mu^{B_2^{-1}}_{B_1}(x')))$$

for any $p_{\mu^{A_2}_{A_1}} : \mu^{B_2}_{B_1}(\mu^{B_2^{-1}}_{B_1}(x')) =_{B_2} x'$ and

$$\mu^{A_3}_{A_2} :\equiv \lambda f.\lambda x'' : B_3.(\mathsf{trp}(p_{\mu^{A_3}_{A_2}}, -) \circ \mu^{C_3(\mu^{B_3}_{B_2}(\mu^{B_3^{-1}}_{B_2}(x'')))}_{C_2(\mu^{B_3^{-1}}_{B_2}(x''))})(f(\mu^{B_3{}^{-1}}_{B_2}(x'')))$$

for any $p_{\mu^{A_3}_{A_2}} : \mu^{B_3}_{B_2}(\mu^{B_3{}^{-1}}_{B_2}(x'')) =_{B_3} x''$.

Hence, for any $x'' : B_3$ and any $f : \Pi_{x'':B_3} C_3(x'')$ their composition amounts to

$$\begin{aligned}
&\mu_{A_2}^{A_3} \circ \mu_{A_1}^{A_2}(f, x'') :\equiv \ \mu^{\Pi_{x'':B_3} C_3(x'')}_{\Pi_{x':B_2} C_2(x')} \circ \mu^{\Pi_{x':B_2} C_2(x')}_{\Pi_{x:B_1} C_1(x)} (f, x'') \\
&= \gamma(f(\mu_{B_1}^{B_2}{}^{-1}(\mu_{B_2}^{B_3}{}^{-1}(x'')))) \\
&= \delta(f(\mu_{B_1}^{B_2}{}^{-1}(\mu_{B_2}^{B_3}{}^{-1}(x''))))
\end{aligned}$$

where γ is an abbreviation for

$$(\mathsf{trp}(p_{\mu_{A_2}^{A_3}}, -) \circ \mu^{C_3(\mu_{B_2}^{B_3}(--))}_{C_2(--)}) \circ (\ \mathsf{trp}(p'_{\mu_{A_1}^{A_2}}, -) \circ \mu^{C_2(\mu_{B_1}^{B_2}(--))}_{C_1(--)})$$

and δ for

$$(\mathsf{trp}(p_{\mu_{A_2}^{A_3}}, -) \circ \mathsf{trp}(p''_{\mu_{A_1}^{A_2}}, -)) \circ (\ \mu^{C_3(\mu_{B_2}^{B_3}(--))}_{C_2(--)}) \circ \mu^{C_2(\mu_{B_1}^{B_2}(--))}_{C_1(--)}$$

and, in turn, $p'_{\mu_{A_1}^{A_2}}$ and $p''_{\mu_{A_1}^{A_2}}$ are the results of the following substitutions:

$$p'_{\mu_{A_1}^{A_2}} \ \equiv \ p_{\mu_{A_1}^{A_2}}[\mu_{B_2}^{B_3}{}^{-1}(x'')/x']$$

and

$$p''_{\mu_{A_1}^{A_2}} \ \equiv \ p_{\mu_{A_1}^{A_2}}[\mu_{B_1}^{B_2}{}^{-1} \circ \mu_{B_2}^{B_3}{}^{-1}(x'')/x'] \ .$$

Now, the last equality depends on this other equality

$$\mathsf{trp}(p''_{\mu_{A_1}^{A_2}}, -) \circ \ \mu^{C_3(\mu_{B_2}^{B_3}(--))}_{C_2(--)} = \mu^{C_3(\mu_{B_2}^{B_3}(--))}_{C_2(--)} \circ \mathsf{trp}(p'_{\mu_{A_1}^{A_2}}, -)$$

that holds by uniqueness of canonical isomorphisms from $C_2(\mu_{B_1}^{B_2}(\mu_{B_1}^{B_2}{}^{-1}(\mu_{B_2}^{B_3}{}^{-1}(z))))$ to $C_3(\mu_{B_2}^{B_3}(\mu_{B_2}^{B_3}{}^{-1}(z)))$.

Then, we can conclude that $\mu_{A_2}^{A_3} \circ \mu_{A_1}^{A_2}$ is a canonical isomorphism, since the components are canonical by induction hypothesis and transport operations compose.

- If $A_1 :\equiv B_1 + C_1$ and $A_2 :\equiv B_2 + C_2$ and $A_3 :\equiv B_3 + C_3$, then by definition of canonical isomorphisms for disjoint sums

$$\mu_{A_1}^{A_2} :\equiv \lambda z.\mathsf{ind}_+(z, z_0.\mathsf{inl}(\mu_{B_1}^{B_2}(z_0)), z_1.\mathsf{inr}(\mu_{C_1}^{C_2}(z_1)))$$

and

$$\mu_{A_2}^{A_3} :\equiv \lambda z.\mathsf{ind}_+(z, z_0.\mathsf{inl}(\mu_{B_2}^{B_3}(z_0)), z_1.\mathsf{inr}(\mu_{C_2}^{C_3}(z_1)))$$

Let us consider the composition $\mu_{A_2}^{A_3} \circ \mu_{A_1}^{A_2}$ applied to $z : B_1 + C_1$, then we get

$$\mu_{B_1+C_1}^{B_3+C_3}(z) \;:\equiv\; \mathsf{ind}_+(z, z_0.\mathsf{inl}(\mu_{B_2}^{B_3}(\mu_{B_1}^{B_2}(z_0))), z_1.\mathsf{inr}(\mu_{C_2}^{C_3}(\mu_{C_1}^{C_2}(z_1))))$$

that corresponds to $\mu_{A_2}^{A_3}(\mu_{A_1}^{A_2}(z))$ and is a canonical isomorphism according to Def.4.3.33.

- If $A_1 :\equiv \mathsf{List}(B_1)$ and $A_2 :\equiv \mathsf{List}(B_2)$ and $A_3 :\equiv \mathsf{List}(B_3)$, then

$$\mu_{A_1}^{A_2} :\equiv \lambda z.\mathsf{ind}_{\mathsf{List}}(z, \epsilon, x.y.u.\mathsf{cons}(u, \mu_{B_1}^{B_2}(y)))$$

and

$$\mu_{A_2}^{A_3} :\equiv \lambda z.\mathsf{ind}_{\mathsf{List}}(z, \epsilon, x.y.u.\mathsf{cons}(u, \mu_{B_2}^{B_3}(y)))$$

that are both canonical isomorphisms by induction hypothesis. Then let us consider the composition $\mu_{A_2}^{A_3} \circ \mu_{A_1}^{A_2}$ applied to $z : \mathsf{List}(B_1)$. We get

$$\mu_{\mathsf{List}(B_1)}^{\mathsf{List}(B_3)}(z) :\equiv \mathsf{ind}_{\mathsf{List}}(z, \epsilon, x.y.u.\mathsf{cons}(u, \mu_{B_2}^{B_3}(\mu_{B_1}^{B_2}(y))))$$

that corresponds to $\mu_{A_2}^{A_3}(\mu_{A_1}^{A_2}(z))$ and is canonical according to Def.4.3.33.

- If $A_1 :\equiv B/R_1$ and $A_2 :\equiv B_2/R_2$ and $A_3 :\equiv B_3/R_3$, then

$$\mu_{A_1}^{A_2} :\equiv \lambda z.\mathsf{ind}_Q(z, x.\mu_{B_1}^{B_2}(x))$$

and

$$\mu_{A_2}^{A_3} :\equiv \lambda z.\mathsf{ind}_Q(z, x.\mu_{B_2}^{B_3}(x))$$

and both are canonical isomorphisms by induction hypothesis. Then let us consider the composition $\mu_{A_2}^{A_3} \circ \mu_{A_1}^{A_2}$ applied to $z : B_1/R_1$. We get

$$\mu_{B_1/R_1}^{B_3/R_3}(z) \;:\equiv\; \mathsf{ind}_Q(z, x.\mu_{B_2}^{B_3}(\mu_{B_1}^{B_2}(x)))$$

that corresponds to $\mu_{A_2}^{A_3}(\mu_{A_1}^{A_2}(z))$ and is a canonical isomorphism according to Def.4.3.33.

□

Finally, we show that any canonical isomorphism is an isomorphism in the sense of Def.4.3.27 and has a canonical inverse:

Proposition 4.3.40 (Closure under inverses). *For any h-sets $A_1, A_2 : \mathcal{U}_1\ [\Gamma]$ in Set_{mf}, if $\mu_{A_1}^{A_2} : A_1 \to A_2\ [\Gamma]$ is a canonical isomorphism, then there exists $\mu_{A_1}^{A_2^{-1}} : A_2 \to A_1\ [\Gamma]$ such that for any $x : A_1$,*

$$\mu_{A_1}^{A_2^{-1}}(\mu_{A_1}^{A_2}(x)) =_{A_1} x$$

and for any $y : A_2$,

$$\mu_{A_1}^{A_2}(\mu_{A_1}^{A_2^{-1}}(y)) =_{A_2} y$$

and moreover $\mu_{A_1}^{A_2^{-1}}$ is canonical.

Proof. - In the case of ground types, the inverses are the identities and hence are canonical.

- If $A_1 :\equiv ||B||$ and $A_2 :\equiv ||C||$, then trivially the inverse of a canonical isomorphism is canonical.

- If $A_1 :\equiv \Sigma_{x:B_1} C_1(x)$ and $A_2 :\equiv \Sigma_{x':B_2} C_2(x')$, then $\mu_{B_1}^{B_2}$ and $\mu_{C_1(x)}^{C_2(\mu_{B_1}^{B_2}(x))}$ are canonical isomorphisms and have canonical inverses, by induction hypothesis. Therefore the inverse given in Def. 4.3.33

$$(\mu_{A_1}^{A_2})^{-1} :\equiv \lambda z.(\mu_{B_1}^{B_2{}^{-1}}(\mathsf{pr}_1(z)), (\mu_{C_1(\mu_{B_1}^{B_2{}^{-1}}(\mathsf{pr}_1(z)))}^{C_2(\mu_{B_1}^{B_2}(\mu_{B_1}^{B_2{}^{-1}}(\mathsf{pr}_1(z))))})^{-1} \circ \mathsf{trp}(p_\mu^{-1}, -)(\mathsf{pr}_2(z)))$$

where $p_\mu : \mu_{B_1}^{B_2}(\mu_{B_1}^{B_2{}^{-1}}(\mathsf{pr}_1(z))) =_{B_2} \mathsf{pr}_1(z)$ is canonical, since the components are canonical by induction hypothesis and transports are canonical by Lemma 4.3.35.

- If $A_1 :\equiv \Pi_{x:B_1} C_1(x)$ and $A_2 :\equiv \Pi_{x':B_2} C_2(x')$, then the components $\mu_{B_1}^{B_2}$ and $\mu_{C_1(x)}^{C_2(\mu_{B_1}^{B_2}(x))}$ are canonical and have canonical inverses by induction hypothesis. Therefore the inverse given in Def.4.3.33

$$(\mu_{A_1}^{A_2})^{-1} :\equiv \lambda f'.\lambda x.((\mu_{C_1(x)}^{C_2(\mu_{B_1}^{B_2}(x))})^{-1}(f'(\mu_{B_1}^{B_2}(x))))$$

is a canonical inverse: because, given $q_\mu : {\mu^{B_2}_{B_1}}^{-1}(\mu^{B_2}_{B_1}(x)) =_{B_1} x$, then for any $f' : \Pi_{x':B_2} C_2(x')$ and any $y : B_1$, the following holds

$$\mu^{A_2}_{A_1}{}^{-1}(f', y) = (\mathsf{trp}(q_\mu, -) \circ ((\mu^{C_2((\mu^{B_2}_{B_1}(\mu^{B_2}_{B_1}{}^{-1}(-))))}_{C_1(\mu^{B_2}_{B_1}{}^{-1}(-))})^{-1}) \circ \mathsf{trp}({q_\mu}^{-1}, -))(f'(\mu^{B_2}_{B_1}(y)))$$

that is a composition of isomorphisms which are canonical by induction hypothesis, since by uniqueness of canonical isomorphisms

$$\mathsf{trp}(q_\mu\ , -) \circ ((\mu^{C_2((\mu^{B_2}_{B_1}(\mu^{B_2}_{B_1}{}^{-1}(-))))}_{C_1(\mu^{B_2}_{B_1}{}^{-1}(-))})^{-1} \circ \mathsf{trp}({q_\mu}^{-1}, -)) = (\mu^{C_2(\mu^{B_2}_{B_1}(x))}_{C_1(x)})^{-1}$$

- The cases $+, \mathsf{List}, A/R$ are a straightforward application of the definition of canonical isomorphisms for these constructors and of the induction hypothesis.

□

4.3.5 The H-category of h-sets

The inductive type of canonical isomorphisms is denoted as $\mathsf{Ciso}(A, B)$ for $A, B : \mathcal{U}_1$ that are h-sets in Set_{mf}. The introduction of the universe Set_{mf} is fundamental, since a direct definition of $\mathsf{Ciso}(A, B)$ cannot be given directly in **HoTT**: assuming an inductive elimination over $\mathcal{U}_1$, which is univalent, would result in a contradiction. Alternatively, we can simply define the type of canonical isomorphisms in the meta-theory, along with the category $\mathsf{Set}_{mf/\cong_c}$ introduced below, following the approach in [89]. The interpretation of **emTT**-types will be carried out in this category.

Definition 4.3.41. The category $\mathsf{Set}_{mf/\cong_c}$ is defined as follows:

- *Ob*: equivalence classes $[A]$ of h-sets $A : Set_{mf}$ equated under canonical isomorphisms. Two objects A and B in $\mathsf{Set}_{mf/\cong_c}$ are declared equal by writing $[A] =_{\mathsf{Set}_{mf/\cong_c}} [B]$ if there exists a canonical isomorphism $\mu^B_A : A \to B$. Then, by Univalence in the ambient universe $\mathcal{U}_1$, $[A] =_{\mathsf{Set}_{mf/\cong_c}} [B]$ implies $A =_{\mathcal{U}_1} B$ in **HoTT**

- *Mor*: equivalence classes $[f] : [A] \to [B]$ induced by functions $f : A' \to B'$ between h-sets A' and B' such that $[A'] =_{\mathsf{Set}_{mf/\cong_c}} [A]$ and

$[B'] =_{\mathsf{Set}_{mf/\cong_c}} [B]$. Moreover, given two functions $f : A' \to B'$ and $g : A'' \to B''$ with $[A'] =_{\mathsf{Set}_{mf/\cong_c}} [A'']$ and $[B'] =_{\mathsf{Set}_{mf/\cong_c}} [B'']$, we define the equality of morphisms $f =_{\mathsf{Set}_{mf/\cong_c}} g$ as $\mu^{B''}_{B'} \circ f =_{A' \to B''} g \circ \mu^{A''}_{A'}$ for canonical isomorphisms $\mu^{A''}_{A'} : A' \to A''$ and $\mu^{B''}_{B'} : B' \to B''$. Finally, the morphism equality $[f] = [g]$ implies $f =_{A \to B} g$ for $f, g : A \to B$.

- Composition of $[f] : [A] \to [B]$ and $[g] : [B] \to [C]$ is defined as $[g \circ f] : A \to C$ for representatives $f : A' \to B'$ and $g : B' \to C'$. The identity is $[\mathsf{id}_A] : [A] \to [A]$, which is the equivalence class of the identity functions in **HoTT**.

Palmgren has discussed in [116] the problem of formalizing category-theoretic notions within type theory and, for this purpose, has introduced the notions of *E-category* and *H-category*. Since these definitions rely upon the notion of (dependent) setoid, I recall the following definition from [89]:

Definition 4.3.42. A *dependent* setoid $B(x)_\sim\ [x : A_\sim]$ is a dependent type $B(x)\ [x : A]$ with a prop-valued equivalence relation $\sim_{B(x)}$.
Furthermore, a dependent setoid comes equipped with *substitution morphisms* σ satisfying the following conditions:

- $x_0 : A, x_1 : A, p : x_0 \sim_A x_1, y : B(x_0) \vdash \sigma(p, y) : B(x_1)$
- $x_0 : A, x_1 : A, p : x_0 \sim_A x_1, y : B(x_0), y' : B(x_0), q : y \sim_{B(x_0)} y' \vdash \sigma(p, y) \sim_{B(x_1)} \sigma(p, y')$
- $x_0 : A, x_1 : A, p_0 : x_0 \sim_a x_1, p_1 : x_0 \sim_A x_1, y : B(x_0) \vdash \sigma(p_0, y) \sim_{B(x_1)} \sigma(p_1, y)$
- $x : A, \rho(x) : x \sim_A x, y : B(x) \vdash \sigma(\rho(x), y) \sim_{B(x)} y$
- $x_0 : A, x_1 : A, x_2 : A, y : B(x_0), p_0 : x_0 \sim_A x_1, p_1 : x_1 \sim_A x_2 \vdash \sigma(p_1, \sigma(p_0, y)) \sim_{B(x_2)} \sigma(\tau(x_0, x_1, x_2, p_0, p_1), y)$

Remark 4.3.43. The second condition states that substitution morphisms preserve the equivalence relation on the family $B(x_0)$. The third condition, instead, expresses that substitution morphisms do not depend on the choice of the paths, i.e. they are natural in the path. The last two conditions state, respectively, that a substitution morphism over the reflexive case amounts to the identity and that substitution morphisms are closed under composition.

Remark 4.3.44. If A is a type, then we can consider the setoid freely generated from the identity over A. This is the *free* setoid (A, Id_A). In such a case, the subsitution morphisms correspond to the transport operations. Indeed, by the elimination rule of propositional eqaulity, we can derive the properties of transport corresponding to the conditions that have been assumed on substitution morphisms: transport preserves equality, the transport of an element along a reflexivity path is equal to the element itself and the composition of transports is equal to the transport along the concatenation of paths.

The definition of E-category is the following:

Definition 4.3.45. An E-category $\mathcal{E}$ consists of the following data:

- a type of objects $C : \mathcal{U}$
- a *setoid* $\mathsf{Hom}(a, b)$ $[a, b : C]$
- for any $a : C$, $1_a : \mathsf{Hom}(a, a)$
- a setoid map $\circ : \mathsf{Hom}(b, c) \times \mathsf{Hom}(a, b) \to \mathsf{Hom}(a, c)$, which satisfies the usual associativity and unit conditions.

The notion of H-category is obtained from that of E-category by imposing a notion of equality over the objects in a way that is compatible with the composition operation:

Definition 4.3.46. A H-category $\mathcal{H}$ is a E-category with the following additional data:

- an equivalence relation $\sim_C$ over the type of objects C
- a family of maps $\tau_{a,b,p} : \mathsf{Hom}(a, b)$ for $p : a \sim_C b$ such that

1. $\tau_{a,a,p} = 1_a$ for $p : a \sim_C a$
2. $\tau_{a,b,p} = \tau_{a,b,q}$ for $p, q : a \sim_C b$
3. $\tau_{b,c,q} \circ \tau_{a,b,p} = \tau_{a,c,r}$ for $p : a \sim_C b, q : b \sim_C c$ and $r : a \sim_C c$.

Moreover, the family of maps $\tau_{a,b,p}$ is actually a family of isomorphisms.
Finally, we can show that the category $\mathsf{Set}_{mf/\cong_c}$ introduced above is an example of H-category:

Proposition 4.3.47. *The category* $\mathsf{Set}_{mf_{/\cong_c}}$ *is a H-category.*

Proof. The type of objects is the setoid whose support is Set_{mf} and the equality $A' \sim B'$ is defined as the truncation of the inductive type of canonical isomorphisms between A' and B' $||\mathsf{Ciso}(A', B')||$. The hom-set between tow objects $\mathsf{Hom}(A', B')$ is the setoid whose support is the set $A' \to B'$ and for $f, g : A' \to B'$, the equality is given by the propositional equality $f =_{A' \to B'} g$. For any $p : ||\mathsf{Ciso}(A', B')||$, $\tau_{A',B',p}$ is defined as $\mathsf{ind}_{||\ ||}(p, z.z)$, which is well-defined since any canonical isomorphism between two h-sets is unique and satisfies the condition of Def.4.3.46; they follow from the closure properties of canonical isomorphisms. □

4.3.6 The interpretation of emTT

The strategy to design a well-defined compatible interpretation of **emTT**in **HoTT** consists of two steps: first, a partial interpretation of **emTT**-raw syntax in **HoTT**-raw syntax is introduced and then, on the top of this partial interpretation, a direct interpretation $(-)^{\mathcal{D}} : \mathbf{emTT} \longrightarrow \mathsf{Set}_{mf_{/\cong_c}}$ of **emTT**-judgements is defined.
The main point is that the partial interpretation is now *multi-functional*: a single **emTT**-expression may have multiple interpretation values, but this time they are all related by canonical isomorphisms and therefore we avoid the problems discussed in the previous subsection. Then, to each **emTT**-judgement the interpretation associates a set of **HoTT**-judgements which are suitably related by canonical isomorphisms.
The use of canonical isomorphisms is crucial to interpret correctly the judgemental equalities of **emTT**. Moreover, when we are dealing with the interpretation of a raw expression depending on some other raw terms, we assume that the type of the latter has been suitably corrected by means of canonical isomorphisms. Finally, the Univalence Axiom plays a fundamental role to get a compatible translation, because it allows to convert canonical isomorphisms into propositional equalities between types in **HoTT**.
The interpretation multi-function will be denoted with $(-)^{\blacktriangledown}$ and with $(-)^{\tilde{\blacktriangledown}}$, if it is the case that a canonical isomorphism occurs in the interpretation. The notation $(-)^{\tilde{\blacktriangledown}}$ is taken from [89] and given a raw expression α, we will write $(\alpha)^{\tilde{\blacktriangledown}}$ instead of $\tilde{\alpha}^{\blacktriangledown}$. The following definitions explain the reasons to introduce this notation and present some preliminary notions:

Definition 4.3.48. The new judgement form

$$A =_{ext} B\ [\Gamma]$$

means that there exists a canonical isomorphism $\mu_A^B : A \to B\ [\Gamma]$ for A *type* $[\Gamma]$, B *type* $[\Gamma]$.

Definition 4.3.49. Let be C *type* $[\Gamma]$ and D *type* $[\Delta]$ with

$$\Gamma \ :\equiv\ [x_1 : A_1, \ldots, x_n : A_n]$$

and

$$\Delta \ :\equiv\ [y_1 : B_1, \ldots, y_n : B_n]$$

Then the new judgement form

$$C\ [\Gamma] =_{ext} D\ [\Delta]$$

means the following: the judgements

$$A_1 =_{ext} B_1,\ \ldots\ , A_n =_{ext} B_n\ [\mu_{A_1}^{B_1}(x_1)/y_1, \ldots, \mu_{A_{n-1}}^{B_{n-1}}(x_{n-1})/y_{n-1}]$$

are derivable and also the judgement $C =_{ext} \widetilde{D}\ [\Gamma]$ is derivable, where

$$\widetilde{D} \ :\equiv\ D[\mu_{A_1}^{B_1}(x_1)/y_1, \ldots, \mu_{A_n}^{B_n}(x_n)/y_n]$$

i.e. it is the result of substituting each y_i in D with $\mu_{A_i}^{B_i}$ for $i \in \{1, \ldots, n\}$.

Definition 4.3.50. Let be $c : C\ [\Gamma]$ and D *type* $[\Delta]$ such that $C\ [\Gamma] =_{ext} D\ [\Delta]$ with Γ and Δ as above. Then the new judgement form

$$c :_{ext} D\ [\Delta]$$

means that $\widetilde{c} : \widetilde{D}\ [\Gamma]$ is derivable, where

$$\widetilde{c} \ :\equiv\ \mu_C^D(\ c(\mu_{A_1}^{B_1}(x_1), \ldots, \mu_{A_n}^{B_n}(x_n))\ )$$

Definition 4.3.51. Let be $a :_{ext} A\ [\Gamma]$ and $b :_{ext} A\ [\Gamma]$. Then the new judgement form

$$a =_{ext} b :_{ext} A\ [\Gamma]$$

means that $p : \widetilde{a} =_{\widetilde{A}} \widetilde{b}\ [\Gamma]$ is derivable.

Remark 4.3.52. The two notations '$=_{ext}$' and '$:_{ext}$' represent, respectively, a notion of *extensional equality* and a notion of *extensional typing*, that amount to the existence of suitable canonical isomorphisms. The judgemental equalities of **emTT** will be interpreted as extensional equalities in this sense and thus rules like the conversion rule will be handled by requiring that the term in the conclusion *extensionally* belongs to a certain type.

The above definitions make clear the meaning of the notation $(-)^{\tilde{\blacktriangledown}}$ which I am going to introduce.

Definition 4.3.53 (Interpretation of **emTT**-raw syntax)**.** We define a multi-functional partial interpretation

$$(-)^{\blacktriangledown} : \text{ raw-syntax } (\mathbf{emTT}) \ \rightarrow \ \text{raw-syntax } (\mathbf{HoTT})$$

together with two auxiliary partial functions

$$\mathsf{pr}_{\mathsf{P}}(-) : \text{ raw-syntax } (\mathbf{HoTT}) \ \rightarrow \ \text{raw-syntax } (\mathbf{HoTT})$$

and

$$\mathsf{pr}_{\mathsf{S}}(-) : \text{ raw-syntax } (\mathbf{HoTT}) \ \rightarrow \ \text{raw-syntax } (\mathbf{HoTT})$$

The definition of $(-)^{\blacktriangledown}$ for **emTT**-contexts and assumptions of variables is given as follows:

$$\begin{aligned}
([\,])^{\blacktriangledown} &:\equiv \mathbf{1} \\
(\Gamma, x \in A)^{\blacktriangledown} &:\equiv \Gamma^{\blacktriangledown}, x : A^{\blacktriangledown} \\
(x \in A\ [\Gamma])^{\blacktriangledown} &:\equiv x : A^{\blacktriangledown}\ [\Gamma^{\blacktriangledown}]
\end{aligned}$$

The interpretation of **emTT**-raw judgements is defined as follows:

$$
\begin{aligned}
&(A\ set\ [\Gamma])^\star \ :\equiv\ A^\star : \mathcal{U}_0\ [\Gamma^\star] \text{ s.th. } \mathsf{pr_S}(A^\star) : \mathsf{isSet}(A^\star)\\
&(A\ col\ [\Gamma])^\star \ :\equiv\ A^\star : \mathcal{U}_1\ [\Gamma^\star] \text{ s.th. } \mathsf{pr_S}(A^\star) : \mathsf{isSet}(A^\star)\\
&(P\ prop_s\ [\Gamma])^\star \ :\equiv\ ||P^\star|| : \mathcal{U}_0\ [\Gamma^\star] \text{ s.th. } \mathsf{pr_P}(||P^\star||) : \mathsf{isProp}(||P^\star||)\\
&(P\ prop\ [\Gamma])^\star \ :\equiv\ ||P^\star|| : \mathcal{U}_1\ [\Gamma^\star] \text{ s.th. } \mathsf{pr_P}(||P^\star||) : \mathsf{isProp}(||P^\star||)\\
&(A = B\ set\ [\Gamma])^\star \ :\equiv\ (A^\star, \mathsf{pr_S}(A^\star)) \ =_{ext} (B^\star, \mathsf{pr_S}(B^\star)) : \mathsf{Set}_{\mathcal{U}_0}\ [\Gamma^\star]\\
&(A = B\ col\ [\Gamma])^\star \ :\equiv\ (A^\star, \mathsf{pr_S}(A^\star)) \ =_{ext} (B^\star, \mathsf{pr_S}(B^\star)) : \mathsf{Set}_{\mathcal{U}_1}\ [\Gamma^\star]\\
&(P = Q\ prop_s\ [\Gamma])^\star \ :\equiv\ (||P^\star||, \mathsf{pr_P}(||P^\star||)) =_{ext} (||Q^\star||, \mathsf{pr_P}(||Q^\star||)) : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^\star]\\
&(P = Q\ prop\ [\Gamma])^\star \ :\equiv\ (||P^\star||, \mathsf{pr_P}(||P^\star||)) =_{ext} (||Q^\star||, \mathsf{pr_P}(||Q^\star||)) : \mathsf{Prop}_{\mathcal{U}_1}\ [\Gamma^\star]\\
&(a \in A\ \ [\Gamma])^\star \ :\equiv\ a^{\tilde\star} :_{ext} A^\star\ [\Gamma^\star]\\
&(a = b \in A\ [\Gamma])^\star \ :\equiv\ a^{\tilde\star} =_{ext} b^{\tilde\star} :_{ext} A^\star\ [\Gamma^\star]
\end{aligned}
$$

The interpretation of **emTT**-types and terms constructors is defined as follows:

$$
\begin{aligned}
&(\,\Sigma_{x\in A}B(x)\ [\Gamma])^\star \ :\equiv \Sigma_{x:A^\star}\, B(x)^\star\ [\Gamma^\star]\\
&(\langle a, b\rangle)^\star \ :\equiv\ (a^{\tilde\star}, b^{\tilde\star})\\
&(\mathrm{El}_\Sigma(d, c))^\star \ :\equiv\ \mathsf{ind}_\Sigma(d^{\tilde\star}, x.y.c(x, y)^{\tilde\star})\\
&\mathsf{pr_S}((\Sigma_{x\in A}B(x)\,)^\star) \ :\equiv\ \mathfrak{s}_\Sigma(A^\star, \lambda x : A^\star.B(x)^\star, \mathsf{pr_S}(A^\star), \lambda x : A^\star.\mathsf{pr_S}(B(x)^\star))
\end{aligned}
$$

$$
\begin{aligned}
&(\Pi_{x\in A}B(x)\ [\Gamma])^\star \ :\equiv\ \Pi_{x:A^\star}\, B(x)^\star\ [\Gamma^\star]\\
&(\lambda x.b(x))^\star \ :\equiv\ \lambda x.b(x)^{\tilde\star}\\
&(\mathsf{Ap}(f, a))^\star \ :\equiv\ f^{\tilde\star}(a^{\tilde\star})\\
&\mathsf{pr_S}((\Pi_{x\in A}B(x))^\star) \ :\equiv\ \mathfrak{s}_\Pi(A^\star, \lambda x : A^\star.B(x)^\star, \lambda x : A^\star.\mathsf{pr_S}(B(x)^\star))
\end{aligned}
$$

$$
\begin{aligned}
&(\mathsf{N}_0\ [\Gamma])^\star \ :\equiv\ \mathbf{0}\ [\Gamma^\star]\\
&(\mathsf{emp}_0(c))^\star \ :\equiv\ \mathsf{ind}_0(c^{\tilde\star})\\
&\mathsf{pr_S}((\mathsf{N}_0)^\star) \ :\equiv\ \mathfrak{s}_0
\end{aligned}
$$

$(\mathsf{N}_1\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathbf{1}\ [\Gamma^\blacktriangledown]$

$(\star)^\blacktriangledown\ :\equiv\ \star$

$(\mathrm{El}_{\mathsf{N}_1}(t,c))^\blacktriangledown\ :\equiv\ \mathsf{ind}_1(\tilde{t^\blacktriangledown}, \tilde{c^\blacktriangledown})$

$\mathsf{pr}_\mathsf{S}((\mathsf{N}_1)^\blacktriangledown)\ :\equiv\ \mathfrak{s}_1$

$(A + B\ [\Gamma])^\blacktriangledown\ :\equiv\ A^\blacktriangledown + B^\blacktriangledown\ [\Gamma^\blacktriangledown]$

$(\mathsf{inl}(a))^\blacktriangledown\ :\equiv\ \mathsf{inl}(\tilde{a^\blacktriangledown})$

$(\mathsf{inr}(b))^\blacktriangledown\ :\equiv\ \mathsf{inr}(\tilde{b^\blacktriangledown})$

$(\mathrm{El}_+(c, d_A, d_B))^\blacktriangledown\ :\equiv\ \mathsf{ind}_+(\tilde{c^\blacktriangledown}, x.d_A(x)^\sim, y.d_B(y)^\sim)$

$\mathsf{pr}_\mathsf{S}((A+B)^\blacktriangledown)\ :\equiv\ \mathfrak{s}_+(A^\blacktriangledown, B^\blacktriangledown, \mathsf{pr}_\mathsf{S}(A^\blacktriangledown), \mathsf{pr}_\mathsf{S}(B^\blacktriangledown))$

$(\mathsf{List}(A)\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathsf{List}(A^\blacktriangledown)\ [\Gamma^\blacktriangledown]$

$(\epsilon)^\blacktriangledown\ :\equiv\ \mathsf{nil}$

$(\mathsf{cons}(\ell, a))^\blacktriangledown\ :\equiv\ \mathsf{cons}(\tilde{\ell^\blacktriangledown}, \tilde{a^\blacktriangledown})$

$(\mathrm{El}_{\mathsf{List}}(c, d, l))^\blacktriangledown\ :\equiv\ \mathsf{ind}_{\mathsf{List}}(\tilde{c^\blacktriangledown}, \tilde{d^\blacktriangledown}, x.y.z.l(x,y,z)^\sim)$

$\mathsf{pr}_\mathsf{S}((\mathsf{List}(A))^\blacktriangledown)\ :\equiv\ \mathfrak{s}_{\mathsf{List}}(A^\blacktriangledown, \mathsf{pr}_\mathsf{S}(A^\blacktriangledown))$

$(A/R\ [\Gamma])^\blacktriangledown\ :\equiv\ A^\blacktriangledown/R^\blacktriangledown\ [\Gamma^\blacktriangledown]$

$([a])^\blacktriangledown\ :\equiv\ [\tilde{a^\blacktriangledown}]$

$(\mathrm{El}_Q(p, c))^\blacktriangledown\ :\equiv\ \mathsf{ind}_\mathsf{Q}(\tilde{p^\blacktriangledown}, \tilde{c^\blacktriangledown})$

$(\mathbf{true} \in R(a,b)\ [\Gamma])^\blacktriangledown\ :\equiv\ p : R(a,b)^\blacktriangledown\ [\Gamma^\blacktriangledown]$ for some term p

$\mathsf{pr}_\mathsf{S}((A/R)^\blacktriangledown)\ :\equiv\ \mathfrak{s}_Q(A^\blacktriangledown, R^\blacktriangledown, \mathsf{pr}_\mathsf{S}(A^\blacktriangledown), \mathsf{pr}_\mathsf{P}(R^\blacktriangledown), r^\blacktriangledown)$ for some term r

$(\mathcal{P}(1)\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^\blacktriangledown]$

$([A])^\blacktriangledown\ :\equiv\ (||A^\blacktriangledown||, \mathsf{pr}_\mathsf{P}(||A^\blacktriangledown||))$

$(\mathbf{true} \in A \leftrightarrow B\ [\Gamma])^\blacktriangledown\ :\equiv\ p : ||A^\blacktriangledown|| \leftrightarrow ||B^\blacktriangledown||\ [\Gamma^\blacktriangledown]$ for some term p

$\mathsf{pr}_\mathsf{S}((\mathcal{P}(1))^\blacktriangledown)\ :\equiv\ \mathfrak{s}_{\mathsf{Prop}_0}$

$(A \to \mathcal{P}(1)\ [\Gamma])^\blacktriangledown :\equiv\ A^\blacktriangledown\ \to\ \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^\blacktriangledown]$

$(\lambda x.b(x))^\blacktriangledown\ :\equiv\ \lambda x.b(x)^{\tilde{\blacktriangledown}}$

$(\mathsf{Ap}(f,a))^\blacktriangledown :\equiv\ f^{\tilde{\blacktriangledown}}(a^{\tilde{\blacktriangledown}})$

$\mathsf{pr}_\mathsf{S}((A\ \to\ \mathcal{P}(1))^\blacktriangledown) :\equiv \mathfrak{s}_\Pi(A^\blacktriangledown, \lambda : A^\blacktriangledown.\mathsf{Prop}_{\mathcal{U}_0}, \mathfrak{s}_{\mathsf{Prop}_0}\ )$

$(\bot\ [\Gamma])^\blacktriangledown\ :\equiv\ ||\mathbf{0}||\ [\Gamma^\blacktriangledown]$

$(\mathsf{true} \in C\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathsf{ind}_{\bot^\blacktriangledown}(c^{\tilde{\blacktriangledown}}) : C^\blacktriangledown\ [\Gamma^\blacktriangledown]$ for some term c

$\mathsf{pr}_\mathsf{P}((\bot)^\blacktriangledown)\ :\equiv\ \mathfrak{p}_{||\ ||}(\mathbf{0})$

$\mathsf{pr}_\mathsf{S}((\bot)^\blacktriangledown)\ :\equiv\ \mathfrak{s}_{coe}((\bot)^\blacktriangledown, \mathsf{pr}_\mathsf{P}((\bot)^\blacktriangledown))$

$(A \vee B\ [\Gamma])^\blacktriangledown\ :\equiv\ A^\blacktriangledown \vee B^\blacktriangledown\ [\Gamma^\blacktriangledown]$

$(\mathsf{true} \in A \vee B\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathsf{inl}_\vee(a^{\tilde{\blacktriangledown}}) : A^\blacktriangledown \vee B^\blacktriangledown\ [\Gamma^\blacktriangledown]$ for some term a

$(\mathsf{true} \in A \vee B\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathsf{inr}_\vee(b^{\tilde{\blacktriangledown}}) : A^\blacktriangledown \vee B^\blacktriangledown\ [\Gamma^\blacktriangledown]$ for some term b

$(\mathsf{true} \in C\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathsf{ind}_\vee(d^{\tilde{\blacktriangledown}}, x.c_1(x)^{\tilde{\blacktriangledown}}, y.c_2(y)^{\tilde{\blacktriangledown}}) : C^\blacktriangledown\ [\Gamma^\blacktriangledown]$ for some terms c_1, c_2, d

$\mathsf{pr}_\mathsf{P}((A \vee B)^\blacktriangledown)\ :\equiv\ \mathfrak{p}_\vee(A^\blacktriangledown, B^\blacktriangledown)$

$\mathsf{pr}_\mathsf{S}((A \vee B)^\blacktriangledown)\ :\equiv\ \mathfrak{s}_{coe}((A \vee B)^\blacktriangledown, \mathsf{pr}_\mathsf{P}((A \vee B)^\blacktriangledown))$

$(A \wedge B\ [\Gamma])^\blacktriangledown :\equiv\ ||A^\blacktriangledown \times B^\blacktriangledown||\ [\Gamma^\blacktriangledown]$

$(\mathsf{true} \in A \wedge B\ [\Gamma])^\blacktriangledown\ :\equiv\ (a^{\tilde{\blacktriangledown}},_\wedge b^{\tilde{\blacktriangledown}}) : ||A^\blacktriangledown \times B^\blacktriangledown||\ [\Gamma^\blacktriangledown]$ for some terms a, b

$(\mathsf{true} \in A\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathsf{pr}_{1\,\wedge}(c^{\tilde{\blacktriangledown}}) : A^\blacktriangledown\ [\Gamma^\blacktriangledown]$ for some term c

$(\mathsf{true} \in B\ [\Gamma])^\blacktriangledown\ :\equiv\ \mathsf{pr}_{2\,\wedge}(c^{\tilde{\blacktriangledown}}) : B^\blacktriangledown\ [\Gamma^\blacktriangledown]$ for some term c

$\mathsf{pr}_\mathsf{P}((A \wedge B)^\blacktriangledown)\ :\equiv\ \mathfrak{p}_{||\times||}(A^\blacktriangledown, B^\blacktriangledown)$

$\mathsf{pr}_\mathsf{S}((A \wedge B)^\blacktriangledown)\ :\equiv\ \mathfrak{s}_{coe}((A \wedge B)^\blacktriangledown, \mathsf{pr}_\mathsf{P}((A \wedge B)^\blacktriangledown))$

$(A \to B\ [\Gamma])^\blacktriangledown\ :\equiv\ ||A^\blacktriangledown \to B^\blacktriangledown||\ [\Gamma^\blacktriangledown]$

$(\mathsf{true} \in A \to B\ [\Gamma])^\blacktriangledown\ :\equiv\ \lambda_\to x.b^{\tilde{\blacktriangledown}} : ||A^\blacktriangledown \to B^\blacktriangledown||\ [\Gamma^\blacktriangledown]$ for some term b

$(\mathsf{true} \in B\ [\Gamma])^\blacktriangledown\ :\equiv\ f^{\tilde{\blacktriangledown}}_\to(a^{\tilde{\blacktriangledown}}) : B^\blacktriangledown\ [\Gamma^\blacktriangledown]$ for some terms a, f

$\mathsf{pr}_\mathsf{P}((A \to B)^\blacktriangledown)\ :\equiv\ \mathfrak{p}_{||\to||}(A^\blacktriangledown, B^\blacktriangledown)$

$\mathsf{pr}_\mathsf{S}((A \to B)^\blacktriangledown)\ :\equiv\ \mathfrak{s}_{coe}((A \to B)^\blacktriangledown, \mathsf{pr}_\mathsf{P}((A \to B)^\blacktriangledown))$

$$
\begin{aligned}
&(\exists_{x\in A}B(x)\ [\Gamma])^{\blacktriangledown} \ :\equiv\ \exists_{x:A^{\blacktriangledown}}\, B(x)^{\blacktriangledown}\ [\Gamma^{\blacktriangledown}]\\
&(\mathsf{true}\in \exists_{x\in A}B(x)\ [\Gamma])^{\blacktriangledown} \ :\equiv\ (a^{\tilde{\blacktriangledown}},_{\exists}\, b^{\tilde{\blacktriangledown}}) : \exists_{x:A^{\blacktriangledown}}\, B(x)^{\blacktriangledown}\ [\Gamma^{\blacktriangledown}] \text{ for some terms } a, b\\
&(\mathsf{true}\in C\ [\Gamma])^{\blacktriangledown} \ :\equiv\ \mathsf{ind}_{\exists}(d^{\tilde{\blacktriangledown}}, x.y.c(x,y)^{\tilde{\blacktriangledown}}) : C^{\blacktriangledown}\ [\Gamma^{\blacktriangledown}] \text{ for some terms } c, d\\
&\mathsf{pr}_{\mathsf{P}}((\exists_{x\in A}B(x))^{\blacktriangledown}) \ :\equiv\ \mathfrak{p}_{\exists}(A^{\blacktriangledown}, \lambda x : A^{\blacktriangledown}.B(x)^{\blacktriangledown})\\
&\mathsf{pr}_{\mathsf{S}}((\exists_{x\in A}B(x))^{\blacktriangledown}) \ :\equiv\ \mathfrak{s}_{coe}((\exists_{x\in A}B(x))^{\blacktriangledown},\ \mathsf{pr}_{\mathsf{P}}((\exists_{x\in A}B(x))^{\blacktriangledown}))
\end{aligned}
$$

$$
\begin{aligned}
&(\forall_{x\in A}B(x)\ [\Gamma])^{\blacktriangledown} \ :\equiv ||\Pi_{x:A^{\blacktriangledown}}\, B(x)^{\blacktriangledown}||\ [\Gamma^{\blacktriangledown}]\\
&(\mathsf{true}\in \forall_{x\in A}B(x)\ [\Gamma])^{\blacktriangledown} \ :\equiv\ \lambda_{\forall}x.b(x)^{\tilde{\blacktriangledown}} : ||\Pi_{x:A^{\blacktriangledown}}\, B(x)^{\blacktriangledown}||\ [\Gamma^{\blacktriangledown}] \text{ for some term } b\\
&(\mathsf{true}\in B(a)\ [\Gamma])^{\blacktriangledown} \ :\equiv\ f^{\tilde{\blacktriangledown}}_{\forall}(a^{\tilde{\blacktriangledown}}) : B(a)^{\blacktriangledown}\ [\Gamma^{\blacktriangledown}] \text{ for some terms } a, f\\
&\mathsf{pr}_{\mathsf{P}}((\forall_{x\in A}B(x))^{\blacktriangledown}) \ :\equiv\ \mathfrak{p}_{||\Pi||}(A^{\blacktriangledown}, \lambda x : A^{\blacktriangledown}.B(x)^{\blacktriangledown})\\
&\mathsf{pr}_{\mathsf{S}}((\forall_{x\in A}B(x))^{\blacktriangledown}) \ :\equiv\ \mathfrak{s}_{coe}((\forall_{x\in A}B(x))^{\blacktriangledown},\ \mathsf{pr}_{\mathsf{P}}((\forall_{x\in A}B(x))^{\blacktriangledown}))
\end{aligned}
$$

$$
\begin{aligned}
&(\mathrm{Eq}(A,a,b)\ [\Gamma])^{\blacktriangledown} :\equiv\ ||\mathrm{Id}_{A^{\blacktriangledown}}(a^{\tilde{\blacktriangledown}}, b^{\tilde{\blacktriangledown}})||\ [\Gamma^{\blacktriangledown}]\\
&(\mathsf{true}\in \mathrm{Eq}(A,a,a)\ [\Gamma])^{\blacktriangledown} \ :\equiv\ |\mathsf{refl}_{a^{\tilde{\blacktriangledown}}}| : ||\mathrm{Id}_{A^{\blacktriangledown}}(a^{\tilde{\blacktriangledown}}, a^{\tilde{\blacktriangledown}})||\ [\Gamma^{\blacktriangledown}] \text{ for some term } a\\
&\mathsf{pr}_{\mathsf{P}}((\mathrm{Eq}(A,a,b))^{\blacktriangledown}) \ :\equiv\ \mathfrak{p}_{||\ ||}(A^{\blacktriangledown}, a^{\tilde{\blacktriangledown}}, b^{\tilde{\blacktriangledown}}, \mathrm{Id}_{A^{\blacktriangledown}}(a^{\tilde{\blacktriangledown}}, b^{\tilde{\blacktriangledown}}))\\
&\mathsf{pr}_{\mathsf{S}}((\mathrm{Eq}(A,a,b))^{\blacktriangledown}) \ :\equiv\ \mathfrak{s}_{coe}((\mathrm{Eq}(A,a,b))^{\blacktriangledown},\ \mathsf{pr}_{\mathsf{P}}((\mathrm{Eq}(A,a,b))^{\blacktriangledown}))
\end{aligned}
$$

Remark 4.3.54. A distinctive feature of the extensional theory **emTT** is that propositions are inhabited just by one canonical element called '**true**'. In the previous definition, this canonical element is interpreted by the corresponding canonical elements at the intensional level: e.g., if the proposition is a conjunction, then **true** will amount to a pair and so on. However, we could have given a different definition consisting in a single clause for judgements including **true**, instead of giving a clause for each constructor. Indeed, we may have simply required that $(\mathbf{true})^{\blacktriangledown}$ would amount to a generic proof-term p in **HoTT**. This alternative definition would allow us to avoid the specification of the interpretation for each constructor, since the generic clause would work for each of them. However, the drawback then would have been that we would have been committed to show how to recover all the intensional canonical proof-terms in the proof of the Validity theorem.

By inspecting the previous definition, it turns out that we can simplify our notation by means of the following definition:

Definition 4.3.55. Let $(-)^{\blacklozenge}$ be the multi-functional interpretation from **emTT**-raw syntax to **HoTT**-raw syntax defined as follows:

- $(\mathcal{J})^{\blacklozenge} :\equiv (\mathcal{J})^{\blacktriangledown}$ if $\mathcal{J}$ is a type judgement

- $(\mathcal{J})^{\blacklozenge} :\equiv (\mathcal{J})^{\tilde{\blacktriangledown}}$ if $\mathcal{J}$ is a term judgement

In what follows, I will systematically make use of this new notation.
The second step consists in the definition of the interpretation $(-)^{\mathcal{D}}$ of **emTT**-judgements into the category $\mathsf{Set}_{mf_{/\cong_c}}$. Informally, the interpretation of a dependent type A under a context Γ into a suitable category $\mathcal{C}$ should amount to a sequence of morphisms in $\mathcal{C}$ going from A to the terminal object 1. For instance, if $\Gamma :\equiv [x_1 : B_1, \ldots, x_n : B_n]$ and $\Gamma \vdash A \; type$, then the interpretation of A $[\Gamma]$ will be

$$[\![A]\!] \to [\![B_n]\!] \to [\![B_{n-1}]\!] \to \ \ldots \ \to [\![B_1]\!] \to 1$$

Namely, if Γ is empty and thus A is a closed type, then its interpretation in $\mathcal{C}$ is given by the morphism $[\![A]\!] \to 1$. Finally, a term $a : A$ is interpreted as a *section* of the morphism interpreting A.
In the light of this remark, we have to allow for the possibility of representing dependent types as morphisms in the category in order to get a proper definition of the interpretation $(-)^{\mathcal{D}}$ and this motivates the following definition:

Definition 4.3.56. Let Γ be a context in **HoTT**. Then we define the *indexed closure* of Γ, $Sig(\Gamma)$, by induction over the length of the context. Moreover, $Sig(\Gamma)$ comes equipped with projection maps $\pi_i^n(z)$ for $z : Sig(\Gamma)$ and $i \in \{1, \ldots, n\}$.

- If $\Gamma :\equiv x : A$, then $Sig\ (\Gamma) :\equiv A$ and $\pi_1^1(z) :\equiv z$

- If $\Gamma :\equiv \Delta, x : A$ of length $n+1$, then

$$Sig\ (\Gamma) :\equiv (\Sigma_{z:Sig(\Delta)}\ A\ [\pi_1^n(z)/x_1, \ldots, \pi_n^n(z)/x_n])$$

and, for any $w : \Sigma_{z:Sig(\Delta)}\ A\ [\pi_1^n(z)/x_1, \ldots, \pi_n^n(z)/x_n]$,

$$\pi_i^{n+1}(w) :\equiv \pi_i^n(\pi_1(w))$$

for $i \in \{1, \ldots, n\}$ and

$$\pi_{n+1}^{n+1}(w) :\equiv \pi_2(w)$$

Notation. In the following, we will denote $a\ [\pi_1^n(z)/x_1, \ldots, \pi_n^n(z)/x_n]$, for $z : Sig(\Gamma)$, with $\overline{a}$.
We define the interpretation $(-)^{\mathcal{D}} : \mathbf{emTT} \ \rightarrow \ \mathsf{Set}_{mf_{/\cong_c}}$ on the top of the partial interpretation $(-)^{\blacklozenge}$ as follows:

Definition 4.3.57 (Interpretation of **emTT**-judgements in $\mathsf{Set}_{mf_{/\cong_c}}$)**.** The interpretation $(-)^{\mathcal{D}} : \mathbf{emTT} \rightarrow \mathsf{Set}_{mf_{/\cong_c}}$ is defined in the following way:

- An **emTT**-type judgement is interpreted as a projection map:

$$(\ A\ type\ [\Gamma]\ )^{\mathcal{D}} \ :\equiv \ [\pi_1] : [Sig(\Gamma^{\blacklozenge}, A^{\blacklozenge})] \rightarrow [Sig(\Gamma^{\blacklozenge})]$$

  that amounts to derive $A^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$, equipped with canonical transports, in **HoTT**.

- An **emTT**-type equality judgement is interpreted as equality of type interpretations in $\mathsf{Set}_{mf_{/\cong_c}}$ and hence as equality of arrows:

$$(\ A = B\ type\ [\Gamma]\ )^{\mathcal{D}} \ :\equiv \ (A\ type\ [\Gamma])^{\mathcal{D}} =_{\mathsf{Set}_{mf_{/\cong_c}}} (B\ type\ [\Gamma])^{\mathcal{D}}$$

  that amounts to derive $A^{\blacklozenge}\ [\Gamma^{\blacklozenge}] =_{ext} B^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$ and hence $A^{\blacklozenge} =_{\mathcal{U}_1} B^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$ in **HoTT**.

- An **emTT**-term judgement is interpreted as a section of the projection interpreting the corresponding type:

$$(\ a \in A\ [\Gamma]\ )^{\mathcal{D}} \ :\equiv \ [\langle z, \overline{a}^{\blacklozenge} \rangle] : [Sig(\Gamma^{\blacklozenge})] \rightarrow [Sig(\Gamma^{\blacklozenge}, A^{\blacklozenge})]$$

  that amounts to derive $a^{\blacklozenge} : A^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$ in **HoTT**.

- An **emTT**-term equality judgement is interpreted as the equality of terms interpretations in $\mathsf{Set}_{mf_{/\cong_c}}$:

$$(\ a = b \in A\ [\Gamma]\ )^{\mathcal{D}} \ :\equiv \ (a \in A\ [\Gamma])^{\mathcal{D}} =_{\mathsf{Set}_{mf_{/\cong_c}}} (b \in A\ [\Gamma])^{\mathcal{D}}$$

  that amounts to derive $a^{\blacklozenge} =_{A^{\blacklozenge}} b^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$, for some $a^{\blacklozenge} : A^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$ and $b^{\blacklozenge} : A^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$, in **HoTT**.

Notation. Let us introduce some notational conventions concerning substitution in types and terms. Given a context $\Gamma \ :\equiv \ \Delta', x_n : A_n, \Delta''$ with $\Delta'' \ :\equiv \ x_{n+1} : A_{n+1}, \ldots, x_m : A_m$, for any term $a : A_n \ [\Delta']$ and for any type B *type* $[\Gamma]$, we denote the substitution of x_n with a in B as

$$B[a/x_n] \textit{ type } [\Delta', \Delta''_a]$$

instead of the extended form

$$B[a/x_n][x'_i/x_i]_{i\in\{n+1,\ldots,m\}} \textit{ type } [\Delta', \Delta''_a]$$

where

$$\Delta''_a \ :\equiv \ x'_{n+1} : A'_{n+1}, \ldots, x'_m : A'_m$$

and

$$A'_j \ :\equiv \ A_j[a_n/x_n][x'_i/x_i]_{i\in\{n+1,\ldots,j-1\}}$$

for $j \in \{n+2, \ldots, m\}$ if $n+2 \leqslant m$, otherwise $A'_{n+1} \ :\equiv \ A_{n+1}[a_n/x_n]$. Finally, if Δ'' is the empty context, then Δ''_a is empty as well. Also, the same notation is adopted for terms.

Lemma 4.3.58 (Substitution)**.** *For any* **emTT***-judgement* B *type* $[\Gamma]$ *interpreted in* $\mathsf{Set}_{mf_{/\cong_c}}$ *as*

$$[\pi_1] : [Sig(\Gamma^\blacklozenge, y : B^\blacklozenge)] \to [Sig(\Gamma^\blacklozenge)]$$

with $\Gamma \ :\equiv \ \Delta', x_n \in A_n, \Delta''$ *and for every* **emTT***-judgement* $a \in A_n \ [\Delta']$ *interpreted as* $[\langle z, \overline{a}^\blacklozenge\rangle] : [Sig(\Delta'^\blacklozenge)] \to [Sig(\Delta'^\blacklozenge, x_n \in A_n^\blacklozenge)]$*, we have that*

$$(\ B[a/x_n] \textit{ type } [\Delta', \Delta''_a]\ )^{\mathcal{D}}$$

is equal in $\mathsf{Set}_{mf_{/\cong_c}}$ *to*

$$[\pi_1] : [Sig(\Delta'^\blacklozenge, \Delta''^\blacklozenge_a, y \in B^\blacklozenge[a^\blacklozenge/x_n])] \to [Sig(\Delta'^\blacklozenge, \Delta''^\blacklozenge_a)]$$

Similarly, for any **emTT***-judgement* $b \in B \ [\Gamma]$*, where* B *and* Γ *are exactly as specified above, and which is interpreted as* $[\langle z, \overline{b}^\blacklozenge\rangle] : [Sig(\Gamma^\blacklozenge)] \to [Sig(\Gamma^\blacklozenge, y : B^\blacklozenge)]$*,*

$$(\ b[a/x_n] \in B[a/x_n] \ [\Delta', \Delta''_a]\ )^{\mathcal{D}}$$

is equal in $\mathsf{Set}_{mf_{/\cong_c}}$ *to:*

$$[\langle z, \overline{b^\blacklozenge[a^\blacklozenge/x_n]}\ \rangle\ ] : [Sig(\Delta'^\blacklozenge, \Delta''^\blacklozenge_a)] \to [Sig(\Delta'^\blacklozenge, \Delta''^\blacklozenge_a, y \in B^\blacklozenge[a^\blacklozenge/x_n])]$$

Proof. The lemma is established by induction over the interpretation of raw terms and raw types and relies upon the Lemma 4.3.34. □

Theorem 4.3.59 (Validity)**.** *If A type $[\Gamma]$ is derivable in* **emTT***, then $(\ A$ type $[\Gamma]\ )^{\mathcal{D}}$ is well-defined.*
If $a \in A\ [\Gamma]$ is derivable in **emTT***, then $(\ a \in A\ [\Gamma]\ )^{\mathcal{D}}$ is well-defined.*
If A type $[\Gamma]$, B type $[\Gamma]$ and $A = B\ [\Gamma]$ are derivable in **emTT***, then $(\ A = B\ [\Gamma]\ )^{\mathcal{D}}$ is well-defined.*
If $a \in A\ [\Gamma]$, $b \in A\ [\Gamma]$ and $a = b\ \in A\ [\Gamma]$ are derivable in **emTT***, then $(\ a = b \in A\ [\Gamma]\ )^{\mathcal{D}}$ is well-defined.*
Therefore, **emTT** *is valid with respect to the interpretation $(-)^{\mathcal{D}}$.*

Proof. The proof is by induction over the derivation of judgements. H-sets in Set_{mf} form a Π-pretopos and, as a result, they possess enough structure to interpret **emTT**-type constructors.
emTT-conversions are correctly interpreted by canonical isomorphisms and, for the definitions given above, we can coerce terms along canonical isomorphisms. In particular, the *conversion* rule

$$\frac{a \in A\ [\Gamma] \qquad A = B\ type\ [\Gamma]}{a \in B\ [\Gamma]}\ (\mathsf{conv})$$

is interpreted as follows: by induction hypothesis, both $(a \in A\ [\Gamma])^{\mathcal{D}}$ and $(A = B\ type\ [\Gamma])^{\mathcal{D}}$ are well-defined. The first amounts to the derivation of $a^{\blacklozenge} : A^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$, for some $a^{\blacklozenge}$ and some $A^{\blacklozenge}\ type\ [\Gamma^{\blacklozenge}]$, in **HoTT**; the latter, instead, amounts to the derivation of $A^{\blacklozenge} =_{ext} B^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$, for some canonical isomorphism $\mu : A^{\blacklozenge} \to B^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$ and for some $A^{\blacklozenge}\ type\ [\Gamma^{\blacklozenge}], B^{\blacklozenge}\ type\ [\Gamma^{\blacklozenge}]$, in **HoTT**. Then, by Univalence, $A^{\blacklozenge} =_{\mathcal{U}_i} B^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$ follows. Therefore, the interpretation of the conclusion $(a \in B\ [\Gamma])^{\mathcal{D}}$ is well-defined, since $\mu(a) : B^{\blacklozenge}\ [\Gamma^{\blacklozenge}, a : A^{\blacklozenge}]$ is derivable in **HoTT** and, moreover, such isomorphism is unique up to propositional equality. Further, it must be observed that the validity of β-rules also depends on the Substitution Lemma stated above.
The power collection of the singleton $\mathcal{P}(1)$ is interpreted as $\mathsf{Prop}_{\mathcal{U}_0} : \mathcal{U}_1\ [\Gamma^{\blacklozenge}]$ together with a proof $\mathsf{pr_S}((\mathcal{P}(1)^{\blacklozenge})) : \mathsf{isSet}((\mathcal{P}(1)^{\blacklozenge}))$. The introduction rule

$$\frac{A\ prop_s\ [\Gamma]}{[A] \in \mathcal{P}(1)\ [\Gamma]}\ \mathcal{P}(1)\text{-I}$$

is interpreted as follows: by induction hypothesis, $(A\ prop_s\ [\Gamma])^{\mathcal{D}}$ is well-defined and this means that $||A^{\blacklozenge}||\ :\ \mathcal{U}_0\ [\Gamma^{\blacklozenge}]$ together with $\mathsf{pr_P}(||A^{\blacklozenge}||)\ :$

$\mathsf{isProp}(||A^\blacklozenge||)$ is derivable in **HoTT**. Therefore the conclusion is immediately valid, since it boils down to derive $(||A^\blacklozenge||, \mathsf{pr_P}(||A^\blacklozenge||)) : \mathsf{Prop}_{\mathcal{U}_0}\ [\Gamma^\blacklozenge]$.
The two rules

$$\frac{\mathbf{true} \in A \leftrightarrow B\ [\Gamma]}{[A] = [B] \in \mathcal{P}(1)\ [\Gamma]}\ \mathcal{P}(1)\text{-}\mathsf{eq} \qquad \frac{[A] = [B] \in \mathcal{P}(1)\ [\Gamma]}{\mathbf{true} \in A \leftrightarrow B\ [\Gamma]}\ \mathcal{P}(1)\text{-}\mathsf{eff}$$

are shown to be valid as follows. In the case of $\mathcal{P}(1)$-eq, we know by induction hypothesis that $(\mathbf{true} \in A \leftrightarrow B\ [\Gamma])^{\mathcal{D}}$ is well-defined and hence there exists a proof-term p such that $p : ||A^\blacklozenge|| \leftrightarrow ||B^\blacklozenge||\ [\Gamma^\blacklozenge]$ is derivable and $(\mathbf{true})^\blacklozenge \ :\equiv\ p$. But then the conclusion is valid, since by Propositional Extensionality we can conclude that $(||A^\blacklozenge||, \mathsf{pr_P}(||A^\blacklozenge||)) =_{\mathsf{Prop}_{\mathcal{U}_0}} (||B^\blacklozenge||, \mathsf{pr_P}(||B^\blacklozenge||))\ [\Gamma^\blacklozenge]$ holds. $\mathcal{P}(1)$-eff, instead, trivially follows from the definition of the partial interpretation $(-)^\blacklozenge$.
The quotient set constructor of **emTT** is interpreted by **HoTT**-set quotients. In particular, the *effectiveness* rule

$$\frac{a \in A\ [\Gamma] \qquad b \in A\ [\Gamma] \qquad [a] = [b] \in A/R\ [\Gamma] \qquad A/R\ \mathit{set}\ [\Gamma]}{\mathbf{true} \in R(a,b)\ [\Gamma]}\ \mathcal{Q}\text{-}\mathsf{eff}$$

is validated as follows: by induction hypothesis, $(a \in A\ [\Gamma])^{\mathcal{D}}$, $(b \in A\ [\Gamma])^{\mathcal{D}}$, $([a] = [b] \in A/R\ [\Gamma])^{\mathcal{D}}$ and $(A/R\ \mathit{set}\ [\Gamma])^{\mathcal{D}}$ are all well-defined. Then this amounts to the fact that there exist terms $a^\blacklozenge, b^\blacklozenge$ in **HoTT** such that the judgements $a^\blacklozenge : A^\blacklozenge\ [\Gamma^\blacklozenge]$, $b^\blacklozenge : A^\blacklozenge\ [\Gamma^\blacklozenge]$, $[a^\blacklozenge] =_{A^\blacklozenge/R^\blacklozenge} [b^\blacklozenge]\ [\Gamma^\blacklozenge]$ are derivable and also the judgement $A^\blacklozenge/R^\blacklozenge : \mathcal{U}_0\ [\Gamma^\blacklozenge]$ together with a proof $\mathsf{pr_S}(A^\blacklozenge/R^\blacklozenge) : \mathsf{isSet}(A^\blacklozenge/R^\blacklozenge)$ is derivable as well for some $A^\blacklozenge$ and $R^\blacklozenge$ in **HoTT**. Since set-quotients are effective in **HoTT**, then the interpretation of the conclusion is well-defined and the effectiveness rule is validated by our interpretation.
The reflection rule for propositional equality

$$\frac{\mathbf{true} \in \mathrm{Eq}(A,a,b)\ [\Gamma]}{a = b \in A\ [\Gamma]}\ \text{Eq-E}$$

is also valid according to our interpretation: by induction hypothesis, $(\mathbf{true} \in \mathrm{Eq}(A,a,b)\ [\Gamma])^{\mathcal{D}}$ is well-defined and then, for some term p and some terms $a^\blacklozenge, b^\blacklozenge : A^\blacklozenge$, the judgement $p : \mathrm{Id}_{A^\blacklozenge}(a^\blacklozenge, b^\blacklozenge)\ [\Gamma^\blacklozenge]$ is derivable in **HoTT**. Therefore, it trivially follows that $(a = b \in A\ [\Gamma])^{\mathcal{D}}$ is well-defined and that the conclusion is valid.
By definition of the interpretation $(-)^\blacklozenge$, *true* judgements, i.e. those judgements in which the canonical proof-term **true** occurs, are interpreted by

restoring a suitable proof-term at the intensional level: that $(\mathbf{true} \in A\ [\Gamma])^{\mathcal{D}}$ is well-defined means that we can derive in **HoTT** the judgement $p : A^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$ such that $(\mathbf{true})^{\blacklozenge} :\equiv p$.
By way of example, let us consider the following rule:

$$\frac{a \in A\ [\Gamma]}{\mathbf{true} \in \mathrm{Eq}(A, a, a)\ [\Gamma]}\ \text{Eq-I}$$

By induction hypothesis, $(a \in A\ [\Gamma])^{\mathcal{D}}$ is well-defined and hence $a^{\blacklozenge} : A^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$ is derivable in **HoTT** for some term $a^{\blacklozenge}$. It follows that the interpretation of the conclusion is also well-defined, since $|\mathsf{refl}_{a^{\blacklozenge}}| : ||\mathrm{Id}_{A^{\blacklozenge}}(a^{\blacklozenge}, a^{\blacklozenge})||\ [\Gamma^{\blacklozenge}]$ is derivable and $(\mathbf{true})^{\blacklozenge} :\equiv |\mathsf{refl}_{a^{\blacklozenge}}|$. □

4.4 Discussion

The aim of the present section is to discuss some relevant aspects of the interpretation of **emTT** in **HoTT** given above. In particular, the possibility of defining an alternative, less direct, interpretation and the fact that the interpretation of the previous section can be understood as a kind of realizability interpretation are discussed. Furthermore, I discuss the relationship with other interpretations of extensional type theory occurring in the literature, starting from Hofmann's proof of the conservativity of extensional type theory over a version of intensional type theory extended with function extensionality and uniqueness of identity proofs axioms [66].

4.4.1 A Different Interpretation

The interpretation of the extensional level **emTT** into the intensional one **mTT** given in [89] suggests that it should possible to define a different interpretation of **emTT** into **HoTT** factoring through the quotient model built in [89].
On a more abstract level of analysis, the fact that the model $Q(\mathbf{mTT})/\simeq$ in [89] is an instance of the categorical construction known as *elementary quotient completion* [91] provides other strong motivations in support of the possibility of defining this alternative interpretation. Indeed, such a construction satisfies a universal property w.r.t. *elementary doctrines* that are closed under stable effective quotients. An example is given by the elementary

doctrine of h-propositions indexed over the syntactic category of h-sets in **HoTT**.

First of all, we recall that an object in the category $Q(\mathbf{mTT})/\simeq$ is an equivalence class $[(A, \sim_A)]$, with two objects being equivalent if they are canonically isomorphic; a morphism in $Q(\mathbf{mTT})/\simeq$ is an equivalence class $[f] : [(A, \sim_A)] \to [(B, \sim_B)]$ induced by a morphism $f : (A', \sim_{A'}) \to (B', \sim_{B'})$, where A' is canonically isomorphic to A and B' is canonically isomorphic to B.

Further, the definition of setoid and dependent setoid given in **mTT** can be clearly stated also in **HoTT**. Then, the problem is to define a functor from $Q(\mathbf{mTT})/\simeq$ to $\mathsf{Set}_{mf_{/\cong_c}}$ that preserves canonical isomorphisms and introduce a mapping internal to **HoTT** that transforms (dependent) setoids to (dependent) h-sets. The functorial mapping should be based on the alternative interpretation of **mTT**-propositions in **HoTT** outlined in Def.4.3.24.

In what follows, we are not going to introduce the formal construction, but we just give some hints about how it could be carried out. First, the following fact should be established by stuctural induction: if $\mu_A^B : A \to B\ [\Gamma]$ is a canonical isomorphism in **mTT**, then $(\mu_A^B)^\bullet : A^\bullet \to B^\bullet\ [\Gamma^\bullet]$ is a canonical isomorphism in **HoTT**. The second step should amount to define an internal map from (dependent) setoids to (dependent) h-sets in **HoTT**. Such a map could be defined by making use of **HoTT**-quotients. For simplicity, we consider just the case A is a closed type: let us call this assignment $(-)^\dagger$ and given a setoid $(A, \sim_A)$, we have that

$$(A, \sim_A)^\dagger \ :\equiv\ A/\sim_A$$

and given a term $a : A$,

$$(a : A)^\dagger \ :\equiv\ [a] : A/\sim_A$$

If this mapping is correctly extended to the general dependent case and if it has been established that canonical isomorphisms are preserved by the interpretation $(-)^\bullet$, then it follows directly that canonical isomorphisms between dependent setoids in **mTT** are send to canonical isomorphisms between the corresponding h-sets in **HoTT**.

In the light of the above, a functor $\mathcal{F}$ from $Q(\mathbf{mTT})$ to $\mathsf{Set}_{mf_{/\cong_c}}$ could be defined as follows:

$$\mathcal{F}((A, \sim_A)) \ :\equiv\ [A^\bullet/\sim_{A^\bullet}]$$

and, given $f : (A, \sim_A) \to (B, \sim_B)$,

$$\mathcal{F}(f) \ :\equiv\ [\lambda z.\mathsf{ind}_{\mathcal{Q}}(z, x.f(x)^\bullet)]$$

Then, $\mathcal{F}_{\cong}$ would amount to the lifting $\mathcal{F}$ to the category $Q(\mathbf{mTT})/\simeq$. Finally, the last step would consist in defining an interpretation of **emTT**-judgements in $\mathsf{Set}_{mf_{/\cong_c}}$ that essentially corresponds to Def.4.3.57.
The main problem with the construction sketched above is that it does not guarantee that the resulting translation from **emTT** to **HoTT** respects the compatibility criterion. A positive answer could be obtained by proceeding as follows. A *category with families* (CwF), in the sense of [43], can be built out of $\mathsf{Set}_{mf_{/\cong_c}}$. Then, both the interpretations can be represented as functors going from a syntactic category built out of **emTT** to this CwF. Therefore, the problem of showing that the alternative interpretation yields a compatibility result reduces to show the existence of a natural isomorphism between the two functors.

4.4.2 True Judgements and Realizability

An interesting feature of the interpretation of **emTT** in **HoTT** is that it *restores computational content*. We know that propositions in **emTT** are definitionally proof-irrelevant: if P is a proposition and $p, q \in P$, then $p = q \in P$.
Proof-irrelevance can be regarded as the result of an abstraction process: in dealing with propositions, all the information encoded in the proof-terms is discarded apart from the fact that at least one of them exists and hence the proposition is inhabited. The idea is formalized also by adding a rule that states that there is at most one canonical proof-term inhabiting a given proposition. Therefore, for a proposition P, what interests us is just the provability of P, but not the many different proofs that may inhabit it. This provides a formalization of Martin-Löf's *true judgements* in [99, 100]: these are judgements of the form 'A *true*', that must be read intuitionistically as 'there exists a proof of A'. In **emTT** A *true* corresponds to $\mathbf{true} \in A$.
Furthermore, this way of conceiving propositions is the key to formalize constructions such as quotients or other basic set-theoretic notions in **emTT** without running in well-known problems (e.g. derivation of LEM). Then, it gives a substantial contribution to make **emTT** a practical set-theoretical framework.
If the propositions-as-monotypes interpretation is regarded as the outcome of an abstraction process that deletes computational information, then the interpretation in **HoTT** can be seen as a way of recovering this information and thus of giving computational meaning to the true judgements of **emTT**.

Under this respect, we argue that the interpretation is set up as a kind of realizability construction. However, the realizers are provided within a type-theoretical framework and thus they are typed. Let us look at the interpretation of the introduction and elimination rules for $\wedge$ to illustrate how the realizers are recovered:

- The introduction rule for the conjunction

$$\frac{\mathbf{true} \in A\ [\Gamma] \qquad \mathbf{true} \in B\ [\Gamma]}{\mathbf{true} \in A \wedge B\ [\Gamma]}\ \wedge\text{-I}$$

  is interpreted as follows: by induction hypothesis, $(\mathbf{true} \in A\ [\Gamma])^{\mathcal{D}}$ and $(\mathbf{true} \in B\ [\Gamma])^{\mathcal{D}}$ are well-defined, that means that the following judgements $p : ||A^{\blacklozenge}||\ [\Gamma^{\blacklozenge}]$ and $q : ||B^{\blacklozenge}||\ [\Gamma^{\blacklozenge}]$ are derivable in **HoTT** for some proof-terms p, q. Then the interpretation of the conclusion is also well-defined, since there exists a derivation of $|(p, q)| : ||A^{\blacklozenge} \times B^{\blacklozenge}||\ [\Gamma^{\blacklozenge}]$.

  The elimination rules for the conjunction are the projections and thus we can consider only one case, since the other is established in the same manner. The rule

$$\frac{\mathbf{true} \in A \wedge B\ [\Gamma]}{\mathbf{true} \in A\ [\Gamma]}\ \wedge\text{-E}_1$$

  is interpreted as follows: by induction hypothesis, $(\mathbf{true} \in A \times B\ [\Gamma])^{\mathcal{D}}$ is well-defined and amounts to derive $c : ||A^{\blacklozenge} \times B^{\blacklozenge}||\ [\Gamma^{\blacklozenge}]$. Then, the interpretation of the conclusion is valid, since we can derive $\mathsf{pr}_{1\,\wedge}(c^{\blacklozenge}) : A^{\blacklozenge}\ [\Gamma^{\blacklozenge}]$.

This case together with the interpretations of $\mathcal{P}(1)$ and Eq displayed above provides a good example of how the computational information is restored. Roughly speaking, if A *true* is obtained by identifying all the possible witnesses of A, then the interpretation restores one of these witnesses p, which can be regarded as a realizer encoding the computational content contained in A. Therefore, the validity of the interpretation has as outcome that true judgements are endowed with computational content and Martin-Löf's interpretation of them is extended to a many-sorted extensional local set theory like **emTT**. Finally, it should be noted that a similar result was already obtained in the context of the interpretation of **emTT** in **mTT** designed in [89]. I just observe that it can be obtained also in the case of the present interpretation.

4.4.3 A comparison with other interpretations

A definition of a class of isomorphisms similar to that given in Def.4.3.33 was introduced by Martin Hofmann in [66, 65]. The definition is by induction over the structure of the types and is given with the aim of establishing the conservativity proof of extensional type theory over intensional type theory extended with function extensionality and Axiom K [3]. They are formulated as follows:

$$\frac{\begin{array}{c}\Gamma \vdash f, g : \Pi_{(x:A)} B(x) \\ \Gamma \vdash p : \mathrm{Id}_{B(x)}(f(x), g(x))\end{array}}{\Gamma \vdash \mathsf{Ext}_{A,B}(f, g, p) : \mathrm{Id}_{\Pi_{(x:A)} B(x)}(f, g)} \text{ Ext-Form}$$

and

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash p : \mathrm{Id}_A(a, a)}{\Gamma \vdash \mathsf{IdUni}_A(a, p) : \mathrm{Id}_{\mathrm{Id}_A(a,a)}(p, \mathsf{refl}_A(a))} \text{ Id-Uni}$$

Furthermore, the formation and introduction rules for propositional equality are the usual ones, but the elimination rule is left out, since it is definable from transport and uniqueness of identity. Indeed, Hofmann introduces a primitive operator $\mathsf{Subst}_{A,P}$ that amounts to transport and is defined by the following rules:

$$\frac{\Gamma \vdash p : \mathrm{Id}_A(a, b) \qquad \Gamma, x : A \vdash P(x)\ \mathit{type} \qquad \Gamma \vdash m : P(a)}{\Gamma \vdash \mathsf{Subst}_{A,P}(a, b, p, m) : P(b)} \text{ Leibniz}$$

that is just the transport of m along the path p and comes with the computation rule

$$\frac{\Gamma \vdash \mathsf{Subst}_{A,P}(a, a, \mathsf{refl}_A(a), m) : P(a)}{\Gamma \vdash \mathsf{Subst}_{A,P}(a, a, \mathsf{refl}_A(a), m) = m : P(a)} \text{ Leibniz-Comp}$$

expressing that the transport along the reflexive path is the identity.
Extensional type theory ($\mathrm{TT_E}$) is not literally an extension of $\mathrm{TT_I}$, but it can be thought as such if the operators introduced in $\mathrm{TT_I}$ are shown to be definable in it. For this purpose, Hofmann introduces a *stripping* map $|-|$ from $\mathrm{TT_I}$ to $\mathrm{TT_E}$ that is defined as follows

[3] Actually, Hofmann refers to this principle as 'uniqueness of identity', which can be a bit confusing in light of the current terminology. However, it is well-known that the uniqueness of identity principle is equivalent to this axiom (see 1.5.5).

- $|\mathsf{Subst}_{A,P}(a,b,p,m)| \;:\equiv\; |m|$
- $|\mathsf{IdUni}_A(a,p)| \;:\equiv\; \mathsf{refl}_{\mathrm{Id}_{|A|}(|a|,|a|)}(\mathsf{refl}_{|A|}(|a|))$
- $|\mathsf{Ext}_{A,B}(f,g,p)| \;:\equiv\; \mathsf{refl}_{\Pi_{(x:|A|)}|B(x)|}(|f|)$

and naturally extends to contexts, types and terms. If an expression of ($\mathrm{TT_I}$) does not contain one of these operators, then the stripping map does nothing. In particular, this translation is trivially sound.
Formally, a type theory S is a conservative extension of a type theory T if, whenever a type of T is inhabited in S, then it is already inhabited in T. Then, the theorem that Hofmann wants to establish is the following: let A be a type under context Γ in $\mathrm{TT_I}$ and $a : |A|[|\Gamma|]$ in $\mathrm{TT_E}$ for some a, then there exists a term a' such that $a' : A\ [\Gamma]$ in $\mathrm{TT_I}$. Now saying that S is a conservative extension of T is logically equivalent to saying that there exists a model Q of S (and hence also of T) such that the interpretation of T in Q is surjective. Therefore, the problem amounts to give a suitable definition of such a model.
The notion of propositional isomorphism turns out to be useful exactly in the definition of the model. In [65], Q is taken to be a CwF where the objects are equivalence classes of contexts and two contexts are identified if they are canonically isomorphic. Then, types in context $[\Gamma]_\sim$ are equivalence classes of pairs $[(\Gamma, A)]_\sim$, with two such pairs being identified if they are canonically isomorphic, and terms of type $[(\Gamma, A)]_\sim$ are equivalence classes of triples, where two triples (Γ, a, A) and (Γ', a', A') are identified if a and a' are propositionally equal in A' under the action of suitable propositional isomorphisms.
Then, we can show that the interpretation of intensional type theory in Q is full, i.e. surjective, and thus conservativity follows. However, as Hofmann points out, the proof involves an essential application of the Axiom of Choice in the form of an arbitrary choice of a representative from each equivalence class. And apparently this application of the axiom cannot be avoided in a semantics given in a set-theoretical metatheory, where judgemental equality is interpreted as set-theoretic equality. The outcome is that Hofmann's construction does not give an algorithm that yields a corresponding judgement in the intensional theory from a typing derivation in the extensional one.
In the case of our interpretation, this problem is avoided. Given a typing derivation in **emTT**, the interpretation is crafted in such a way that we can always recover a corresponding term in **HoTT** without applying AC. In

general, we may have multiple possible translation values for each **emTT**-derivation, but the way the interpretation is defined guarantees that at least one of these values can always be computed and also that they are related by canonical isomorphisms. Furthermore, **emTT**-judgemental equality is interpreted as the existence of suitable canonical isomorphisms in **HoTT** and not as set-theoretical equality, since, contrary to Hofmann's model, the whole construction is carried out in a type-theoretic context. Therefore, our interpretation has a strong syntactic flavour and can be regarded as a translation between type-theories.
Finally, in our case a conservativity result is not achieved, but this was expected. First of all, **emTT** is not an extension of **HoTT**. And we also know that in **HoTT** the axiom of unique choice is derivable, whereas this principle cannot be derived in **MF**.
A first attempt to make Hofmann's interpretation effective was the work by Oury in [114], which then has been improved in [171]. There, a translation of the extensional calculus of constructions into the intensional one (extended with function extensionality and UIP axioms) is provided. The interpretation gives rise to a syntactical translation which associates to each extensional judgement a set of judgements in the intensional theory. Therefore, this interpretation is more similar to ours, because it is multi-functional and avoids any application of choice. On the other hand, the translations given in [114, 171] do not rely on canonical isomorphisms, but assume instead a notion of *heterogeneous* equality which links terms belonging to different types.

Chapter 5

The Constructive Conception of Predicativity

The aim of the present chapter is to shed light on different characterizations of predicativity that can be found in the literature and, especially, on the notion of *generalized* predicativity, which is often associated with Martin-Löf Type Theory and systems such as Constructive Zermelo-Fraenkel Set Theory **CZF**. The notion of generalized predicativity is taken to be grounded on the *constructive* conception of predicativity "which recognises a construction as predicative if it has a clear inductive structure" [115]. Such conception is meant to warrant a notion of predicatively admissible construction which is wider than that provided by the proof-theoretic conception as codified in the classical works by Feferman [45] and Schütte [139][1].

The structure of the chapter is the following: in the first section, an historical introduction to the problem of impredicative definitions is provided; then, I consider as case study the predicative system set up by Hermann Weyl in [164, 168] and analyze the foundational principles underlying this system as well its relation with Martin-Löf Type Theory. Finally, the last section is entirely devoted to the constructive conception of predicativity and to the analysis of the arguments that may support this conception.

[1]The results by Feferman and Schütte arise within the framework of a general approach to predicativity outlined by Kreisel. Furthermore, it is worth noting that Feferman has explored other potential analyses of the notion of predicativity in his later works. Crosilla discusses Feferman's contribution to predicativity in a broader context in [29].

5.1 The problem of impredicativity

The early debate about Russell's Paradox as well as other paradoxes shaking the foundations of mathematics at the beginning of the last century set the context for the appearance of the notion of predicativity.
It was Russell [133] who introduced the word 'predicative' to refer to those propositional functions that determine a class. However, at first he did not give a clear-cut criterion to distinguish between predicative and impredicative propositional functions. Such a criterion came out in the subsequent discussion with Poincaré. Indeed, around the same time [121], the French mathematician singled out two features as the hallmarks of all paradoxes: they are all based on *viciously circular* definitions and tacitly assume that infinite totalities are *completed*. For Poincaré predicative collections are those satisfying a certain *invariance* property: any extension of a predicative collection does not *disorder* the collection itself.
Around the same years, Russell gave slightly different definitions of the *Vicious Circle Principle* (VCP) and embarked on the project of building a formal system where such a principle is strictly observed [134]. The resulting system was the *ramified* theory of types. Roughly, VCP established that we cannot define an entity by quantifying over a totality that includes the entity itself. Without going into all the details of the Poincaré-Russell debate, two main conceptual points can be pinned down:

- The whole question of predicativity is primarily a problem about *definability*. Therefore, the whole discussion can be rephrased as follows: given a language L, what L-expressions count as legitimate definitions? Then, fixing a formal language is fundamental to get a precise notion of predicativity and it turns out that one of the main shortcomings of the early debate on predicativity, especially in Poincaré's case, was the fact that the formal details were left unspecified.

- A predicative class X must be conceived as something that is built in stages. Therefore, X has an inherently potential nature and in no way can be regarded as a completed totality.

With Zermelo's early axiomatization of set theory, it became clear that paradoxes could be avoided just by introducing a suitable restriction on the comprehension axiom without sacrificing impredicative definitions. Further, Zermelo explicitly advocated for the usefulness and the unproblematic nature

of impredicative definitions in mathematics. The successful development of Zermelo's proposal represented a serious setback for predicativism.[2]

On the other hand, another significant symptom of difficulty for predicative mathematics was the fact that Russell and Whitehead were forced to assume the controversial Axiom of Reducibility to give a satisfactory treatment of many mathematical notions in the ramified theory of types. Such an axiom did not seem to be justified on predicative grounds. Therefore, the main challenge was to show that predicative mathematics was a viable project and not just a philosophical chimera.

Hermann Weyl took up the challenge in his monograph *Das Kontinuum* [164]. Weyl articulated a coherent predicative standpoint and was able to reconstruct significant parts of mathematics on this basis. In particular, he realized that in order to recover some fundamental theorems of analysis it was enough to stick to the first levels of the ramified hierarchy and assume only an *arithmetical* comprehension axiom that allows to form subsets of $\mathbb{N}$ of the form $\{n \in \mathbb{N} \mid \phi(n)\}$ with the quantifiers of ϕ ranging only over the natural numbers. Under these assumptions, Weyl derived predicatively acceptable versions of the Least Upper Bound principle, the Bolzano-Weierstrass Theorem, the Heine-Borel Theorem as well as other basic results of real analysis.

However, in the years after the publication of *Das Kontinuum*, the interest in predicative mathematics declined[3] and Weyl himself abandoned his own attempt. The interest in predicativity flourished after the Second World War, when the technical tools developed in proof theory and recursion theory made possible to investigate this notion in a more detailed and fruitful manner.

Kleene in [74] investigated hierarchies of definitions of sets of natural numbers in the context of the ramified analytical hierarchy. In particular, he proved that the class of *hyperarithmetic* sets corresponds to the Δ^1_1-sets. Then Kreisel [78] suggested the identification of predicatively definable sets with hyperarithmetical ones. Wang in [161] had proposed a similar characterization. One of the shortcomings of this proposal was that it crucially relied upon the impredicative notion of well-ordering. Therefore, Kreisel [77] proposed to amend the characterization by imposing the requirement that a predicative ordinal is not only defined by a well-ordering relation in a given system of

[2] An early discussion and defense of impredicative definitions can also be found in [129].

[3] Nevertheless, it must be recalled that one of the most important achievement in set theory was inspired by predicative concepts and, in particular, by ramification. I am referring to Gödel's model of *constructible* sets and his proof of relative consistency of the Axiom of Choice and of the Continuum Hypothesis.

ramified analysis RA_α, but such a relation should be proved to be a well-ordering at a previous stage. The crucial step was taken, independently, by Feferman [45] and Schütte [139] who established that the least non-predicatively provable ordinal is Γ_0. Accordingly, Γ_0 was recognized as the limit ordinal of predicative theories and those theories whose proof-theoretic ordinal exceeds Γ_0 are not to be counted as predicative.
The Kreisel-Feferman-Schütte analysis gave a sharply defined bound on the notion of predicativity and hence a quantitative criterion that was at the same time philosophically motivated. In this sense, they provided an exact characterization of the classical notion of predicativity given the natural numbers.
Around the same years, a different conception of predicativity was appearing, motivated by a (broadly understood) constructive view of mathematics and mainly focused on the analysis of *inductive definitions*.
Roughly speaking, a set X is inductively defined if it satisfies the following three clauses:

1. canonical or initial elements of X are specified;
2. a way is given to build new elements of X from previously constructed ones;
3. a closure condition asserting that nothing else belongs to X is given.

More formally, an inductive definition is given by an operator

$$T : \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})$$

that is *monotone*:

$$X \subseteq Y \to T(X) \subseteq T(Y)$$

and moreover, given a set X, X is T-closed if $T(X) \subseteq X$. By the Knaster-Tarski Theorem, the following set exists:

$$A_T = \bigcap \{X \mid X \ T\text{-closed}\}$$

A_T is the smallest T-closed set. We have that $T(A_T) = A_T$ and for any X such that $T(X) = X$, $A_T \subseteq X$, which means that A_T is the least fixed point of T. A set X is said to be inductively defined if X is defined as the least fixed point of a monotone operator T.

The set-theoretic definition is blatantly impredicative. However, Wang in [162] suggested that an inductive definition is impredicative when it is given set-theoretically, but if it is assumed that the predicativity constraints do not make any commitment to explicit definitions, then the predicative status of inductive definitions can be safeguarded. Lorenzen and Myhill in [85] endorsed the predicative status of inductive definitions. They claimed that such definitions do not make any use of actual infinity thus resorting to one of the original insights related to the notion of predicativity. In particular, their analysis seems to justify even iterated inductive definitions and in this way extend the domain of predicative mathematics [4].

The point I want to stress is that all these ideas were related more or less explicitly with constructive mathematics. Then, the constructive conception of predicativity that was appearing presented some significant differences with the classical conception. In any event, also this new development of predicative ideas was still influenced by insights contained in the writings of Poincaré and Weyl.

Finally, we should also consider the hypothesis, explicitly formulated by Kreisel in [77], that the status of inductive definitions and, in general, the extension of predicative mathematics may depend on the underlying logic [5]. Kreisel was quite skeptical about the possibility of arguing in favour of the predicativity of inductive definitions within a classical setting. However, he sketched an argument aimed at a justification of their predicative character in the context of his *theory of constructions* [79]. The purported justification would be based on the intuitionistic proof-conditions and on the decidability of the notion of construction. Therefore, it would turn out to be a matter of *meaning*.

[4] A philosophical analysis of the conception of predicativity advocated by Lorenzen and Myhill and of its difference with the Kreisel-Feferman-Schütte conception has been given by Parsons in [118] .

[5] Kreisel made also the following intriguing historical remark:

> "Il me semble que, sans doute inconsciemment, Poincaré donnait un sense intuitionniste au probléme de la prédicativité, tandis que Russell lui donnait un sens classique. "([77], p.389)

5.2 Weyl's Classical Predicativity

Hermann Weyl's *Das Kontinuum* [164] can be regarded as the first systematic attempt to develop significant parts of mathematics on strictly predicative grounds. In particular, Weyl was concerned with the recasting of fundamental results from real analysis in a way that turns out to be predicatively acceptable. Among the results established in *Das Kontinuum*, there are *sequential* versions of the Least Upper Bound principle and the Heine-Borel Theorem, the existence of a maximum and a minimum for continuous functions on a closed interval, uniform continuity of continuous functions on a closed interval and the Fundamental Theorem of the Calculus. Then, the main merit of the book was to show that predicative mathematics was a viable project. Furthermore, Weyl's predicative reconstruction of the continuum took place within the wider context of a comprehensive philosophical view of the foundations of mathematics.
Weyl was deeply dissatisfied with the new set-theoretic foundations axiomatized by Zermelo and believed that it did not provide a satisfactory remedy to the paradoxes. At a more fundamental level, he did not subscribe to the conception of set underlying set theory, since it assumed a dangerous notion of actual infinity. According to Weyl, Cantor and Zermelo's fundamental error was to deal with infinite collections as if they were *completed* and *closed* entities:

> "The notion that an infinite set is a "gathering" brought together by infinitely many individual arbitrary acts of selection, assembled and then surveyed as a whole by consciousness is nonsensical; "inexhaustibility" is essential to infinite." [[168], p. 23]

Also, he reaffirmed his contention in a later work:

> "The deepest root of the trouble lies elsewhere: a field of possibilities open into infinity has been mistaken for a closed realm of things existing in themselves."[[170], p.243]

These admittedly suggestive passages nicely illustrate one of Weyl's main tenets: infinite totalities are inherently *potential* and *open*. Accordingly, the problem with impredicative definitions is that they pick out an object from an open collection by treating such a collection as if it were closed and completed. Predicative definitions may quantify only over those collections that Weyl calls

extensionally determinate, such as finite sets or the set of natural numbers. A collection like $\mathbb{R}$, instead, is not extensionally determinate and thus it is an example of open collection. A definition containing a quantification over an open collection must be counted as impredicative.
The affinity with Poincaré's thought concerns the privileged epistemic status assigned to the natural numbers and to the associated principle of induction:

> "The idea of iteration, i.e. of the sequence of the natural numbers, is an ultimate foundation of mathematical thought."([168], p.48)

The idea of iteration that yields the sequence of natural numbers is given in intuition and must be assumed as primitive. In particular, it follows that natural numbers cannot be defined in more elementary terms and hence are not *reducible* to other more fundamental concepts. Note the evident difference with other foundational projects such logicism or set-theoretic reductionism that aimed to give a definition of natural numbers in terms of other primitive notions. According to Weyl, all these purported definitions tacitly assume our understanding of the notion of natural number. Therefore, they do not achieve a real foundational reduction or explanation.

5.2.1 Interpretations of *Das Kontinuum*

In *Das Kontinuum*, Weyl presented a framework for predicative mathematics, that is discussed at some length especially in the first part of the book. However, some definitions are not stated with sufficient clarity and there is some confusion between syntactic and semantic aspects, that is hardly surprising since the book dates back to 1918. Therefore, Weyl's presentation leaves room for ambiguities and this makes difficult to single out the exact system Weyl had in mind.
The most influential reconstruction of Weyl's system is that proposed by Feferman in his essay [48], which has been reprinted in the collection [49]. Feferman discussed the relevance of Weyl's work for modern research in foundations of mathematics also in [50].
A basic tenet of Feferman's interpretation is that Weyl in *Das Kontinuum* was outlining a general program of *arithmetical* mathematics. In this sense, his foundational program anticipates to some extent the *Reverse Mathematics* program [144], that is a very active field of research in contemporary logic. By arithmetical mathematics Feferman basically means the following: Weyl

assumes the set of natural numbers $\mathbb{N}$ as a primitive domain and admits just those subsets of $\mathbb{N}$ that are definable by means of formulas quantifying only over $\mathbb{N}$. A formula ϕ which contains quantifiers ranging only over natural numbers and not sets thereof is called *arithmetical*. Then, Weyl accepts the comprehension axiom in the following form, that is known as *arithmetical comprehension*:

$$(\mathsf{aca}_0)\ \exists X\ \forall n\ (n \in X\ \leftrightarrow \phi(n))$$

where $\phi(n)$ is an arithmetical formula and X does not occur free in it.
Weyl's problem was to preserve a predicative account of mathematics without resorting to ramification in the sense of Russell, since the latter made quite cumbersome the formalization of standard mathematical notions. The insight that definable sets of the first level of the ramified hierarchy are sufficient for the predicative development of analysis provided the solution according to Feferman ([49], p.258). These sets amount exactly to the arithmetically definable ones.
Among the principles laid down in *Das Kontinuum*, Weyl assigned a special role to the *Principle of Iteration* (*Iterationsprinzip*) ([168], p.36). Weyl gave slightly different formulations of this principle, but the idea behind it seems to be the following: given a set-valued function f[6], we can introduce a sequence of functions $f^1, \ldots, f^n, \ldots$ that is given by the clauses

$$f^1(X) = f(X)$$

and

$$f^{n+1}(X) = f^n(f(X)).$$

The variants of this basic schema consist in allowing some additional parameters.
A contentious point of Feferman's interpretation is that such a principle is 'incoherent' with Weyl's foundational aims, since it destroys the arithmetical character of his system. The most elementary version of the principle is sufficient to introduce non-arithmetically definable objects. If we state the principle in the following form

$$\Phi^*(1, X)\ =\ \Phi(X)$$

[6]Weyl's original formulation is in terms of relations.

$$\Phi^*(n+1, X) \;=\; \Phi^*(n, \Phi(x))$$

(that is the familiar principle of iteration for set-valued functions, since $\phi^*(n, X) := \phi^n(X)$), then by assuming basic facts from computability theory and applying a diagonal argument it turns out that a non-arithemtical relation is definable ([49], p. 265). Therefore, Feferman concludes that the Principle of Iteration does not fit well with Weyl's program of arithmetical mathematics. According to Feferman, there are two possible remedies:

(α) A restriction of the Principle of Iteration in such a way that arithmetical definability is preserved.

(β) Endorse the full strength of the principle with the commitment to show that the resulting system is predicatively justified.

These two strategies lead to two different axiomatic theories that are called by Feferman, respectively, $\mathrm{K}^{(\alpha)}$ and $\mathrm{K}^{(\beta)}$.
Feferman acknowledges that both strategies can be pursued, but he opts for the first one. He does not spell out the reasons behind this choice, but I think that at least three motivations can be dug up from his discussion:

1. The first and most obvious one is that only (α) appears to be coherent with Weyl's foundational project as Feferman understands it. According to Feferman's interpretation, the Principle of Iteration is a defect of Weyl's theory, since he was not aware that this principle entails the possibility of defining non-arithmetical sets. This was an unintended consequence, which does not imply that Weyl was also willing to accept non-arithmetical definitions.

2. If Weyl's aim was to show that the purely arithmetical system suffices for the development of analysis, then it is reasonable to amend Weyl's system with a restricted version of the iteration principle to show that such it is the case.

3. Finally, Feferman interprets the closing passage of *Das Kontinuum* as providing evidence for the claim that Weyl conceived arithmetical

mathematics as all what we need for scientific applications[7]. The idea that weak systems, in particular systems avoiding strong set-theoretic assumptions, suffice for scientifically applicable mathematics has been thoroughly advocated by Feferman himself [8].

Then, the theory K^{α} represents a reformulation of Weyl's system that conjugates the fulfillment of modern standards of rigour with faithfulness to Weyl's original ideas ([49], p.268).
In the following, I just sketch the main features of K^{α} and refer to [49] for a detailed exposition of the theory.
The language of K^{α} includes three sorts of variables: numerical variables, functional variables and relational variables. Constants of each sort are denoted by special symbols: e.g. 0 stands for the numerical constant zero and *sc* for the functional constant representing the successor function. Furthermore, there are functional constants for all primitive recursive functions which are governed by the usual axioms. Similarly, there are relational constants for primitive recursive relations. The atomic formulas consist of equalities $s = t$, for s, t well-formed terms, and of formulas of the form $A(t_1, \ldots, t_k)$, where A is a k-ary relational symbol and $t_1, \ldots, t_k$ are terms. We can introduce the notation $(t_1, \ldots, t_k) \in A$ for the latter formulas. The set of well-formed formulas is inductively generated from atomic formulas, the connectives $\neg$ and $\vee$ and first and second-order existential quantifiers.
The logic of K^{α} is classical many-sorted predicate calculus with equality. The axiomatization includes, in addition to the conditions for primitive recursive functions, the familiar axioms for zero and the successor and the following form of induction axiom:

$$(\mathtt{ind}_0)\ 0 \in A \ \wedge\ \forall n\ (n \in A \ \rightarrow\ \mathtt{sc}(n) \in A) \ \rightarrow\ \forall n\ (n \in A)$$

It must be observed that in a system containing the arithmetical comprehension axiom, this induction principle suffices to derive all the instances of the induction scheme for arithmetical formulas. However, Feferman claims that it is not entirely straightforward to determine what form of induction Weyl

[7] Weyl has given fundamental contributions to mathematical physics and has been always interested in the relation between mathematics and physical sciences. Moreover, few months after *Das Kontinuum*, his exposition of the theory of relativity *Raum, Zeit, Materie* [165] was published.

[8] See, in particular, the last essay in the collection [49].

would have accepted. In particular, it is not clear whether he would have included the full induction scheme

$$(\mathtt{ind})\ \phi(0)\ \wedge \forall n\ (\phi(n)\ \rightarrow\ \phi(\mathtt{sc}(n)))\ \rightarrow\ \forall n\ \phi(n)$$

with ϕ arbitrary formula, which is proof-theoretically stronger.
In this system, the Principle of Iteration is restricted to *numerical-valued* functions and essentially follows from the axioms for primitive recursion. This prevents the possibility of defining non-arithmetical relations.
Further, there are two axioms for the explicit definition of functions and relations. The last two axioms of $\mathrm{K}^{(\alpha)}$ are forms of arithmetical comprehension, respectively, for relations and functions: if $\phi(n, m, X, f)$ is an arithmetical formula, then

$$(\mathtt{aca}_{\mathtt{rel}})\ \exists A\ \forall n\ (A(n) \leftrightarrow \phi(n, m, X, f))$$

and if $\phi(n, p, m, X, g)$ is an arithmetical formula, then

$$(\mathtt{aca}_{\mathtt{fun}})\ \forall n\ \exists! p\ \phi(n, p, m, X, g) \rightarrow \exists f\ \forall n, p\ (f(n) = p \leftrightarrow \phi(n, p, m, X, g))$$

In such a system, we can either dispense with relations in favour of the corresponding characteristic functions or dispense with functions in favour of their graphs seen as relations. The first fact follows from $(\mathtt{aca}_{\mathtt{rel}})$ together with the principle of explicit definition for relations, while the latter follows by taking as ϕ the (arithmetical) formula $(n, p)\ \in A$, for each relation symbol A.
Feferman remarks that we can drastically simplify $\mathrm{K}^{(\alpha)}$ by means of elementary coding techniques, which allow us to move from k-ary relations to unary ones and hence to sets. Since we know that also functions are dispensable, then we can recast the language of the system with only two sorts of variables: numerical and set variables. Accordingly, the two comprehension principles for relations and functions are then replaced by the single arithmetical comprehension axiom $(\mathtt{aca}_0)$. The resulting system is $\mathbf{ACA_0}$, one of the sub-systems of second-order arithmetic which has been investigated in the field of Reverse Mathematics [144]. Therefore, it follows that K^{α} is a conservative extension of Peano Arithmetic. Finally, note that if in $\mathbf{ACA_0}$, we replace the induction axiom $\mathtt{ind}_0$ with the full induction scheme $\mathtt{ind}$, then we get the system known as $\mathbf{ACA}$, that is no longer conservative over $\mathbf{PA}$. However, both systems

enjoy an arithmetical model, in the sense that functions and relations are interpretable as those functions and relations which are arithmetically definable over the standard natural number structure.

As observed above, a potentially problematic aspect of Feferman's interpretation is his rejection of the Principle of Iteration as stated by Weyl, because it entails the possibility of outstripping arithmetical definability and thus is in contrast with Weyl's arithmetical mathematics program. Then, it is not surprising that the main objections to Feferman's interpretation have concerned especially this point.

In a series of papers [2, 4, 3], R.Adams and Z.Luo have proposed an alternative formalization of Weyl's system. The system of *Das Kontinuum* is formalized as a *logic-enriched* type theory, i.e. a type theory with a primitive sort of propositions that is completely separated from types. The resulting theory is called $\mathbf{LTT_W}$ and, apart from some technical details, is arguably quite faithful to Weyl's original system. In fact, the syntax of $\mathbf{LTT_W}$ embodies some important distinctions which Weyl explicitly made in *Das Kontinuum*: e.g., the distinction between basic categories (types), the paradigmatic example being the category of natural numbers $\mathbb{N}$, and *ideal* categories, such as the category $\mathrm{Set}(A)$, which always exists for any category A. An object of the category $\mathrm{Set}(A)$ has the form $\{x^A \mid \phi(x^A)\}$, where x is a variable of category A and ϕ is an arithmetical proposition. Further, there is a corresponding axiom scheme

$$\forall y^A \ (y^A \in \{x^A \mid \phi(x^A)\} \ \leftrightarrow \ \phi[y^A/x^A] \ )$$

for any category A and any arithmetical proposition ϕ.

Adams and Luo raise an interesting point in connection with the notion of predicative definability assumed in *Das Kontinuum*. In the first chapter of the monograph, Weyl gives a definition of cardinality that makes use of third-order quantification and involves a non-arithmetical formula. Then, in the second chapter, he proves some elementary results about cardinalities: his arguments there contain applications of induction to non-arithmetical formulas. Therefore, Adams and Luo argue that these examples provide a compelling case against the equivalence of Weyl's system with $\mathbf{ACA_0}$, since he assumes an induction principle stronger than ($\mathtt{ind}_0$) and his system is not restricted to the second-order. In fact, $\mathbf{LTT_W}$ is provably stronger than $\mathbf{ACA_0}$ and is at least as strong as $\mathbf{ACA}$ [3].

The analysis provided by Adams and Luo casts doubts about the claim that

Weyl intended to pursue a program of arithmetical mathematics. On the other hand, it must be noted that already Feferman acknowledged that the whole system of *Das Kontinuum* is sensibly stronger than $\mathrm{K}^{(\alpha)}$ and hence than $\mathbf{ACA_0}$:

> "One often reads that the axiom system on which *Das Kontinuum* is based is that of Res-$(\Pi^0_\infty$-CA$)$[9] or an equivalent version (such as $\mathrm{K}^{(\alpha)}$). Indeed, this is what I had said on various occasions until looking more closely at the monograph for the purposes of the present essay. It is more accurate to say that it formalizes that part of Weyl's system which meets his aim of a purely arithmetical interpretation, though the system as a whole (reconstructed in something like $\mathrm{K}^{(\beta)}$, below) is stronger than Res-$(\Pi^0_\infty$-CA$)$." ([49], p.272)

However, Feferman still maintains that the third-order theory $\mathrm{K}^{(\beta)}$ should be sufficient to formalize Weyl's system.
The most penetrating objections to Feferman's interpretation of *Das Kontinuum* have been given by A.Avron in [9], which is the most updated reconstruction of Weyl's framework:

> "First, none of Feferman's formalizations of Weyl's system is really faithful to those of Weyl. Second, Weyl's work actually transcends what Feferman calls above 'arithmetical mathematics'. Third, I believe that Weyl's system is *superior* to its modern counterparts for the task of serving as a basis of a predicatively justified, natural development of classical analysis. Fourth, there are some important hidden ideas in *Das Kontinuum* that have so far been completely ignored in the literature on it, but will be uncovered and exploited below." ([9], p.3)

Avron gives a careful analysis of the philosophical tenets underlying Weyl's monograph and introduces a system **WA**, that is similar to the system $\mathbf{LTT_W}$, but is meant to be even more faithful to Weyl's original framework. The main difference between **WA** and the axiomatization given by Adams and Luo is that in the first each type or category comes equipped with an associated

[9]This is just another name for $\mathbf{ACA_0}$.

notion of equality, whereas in $\mathbf{LTT_W}$ equality is assumed as primitive only for the category of natural numbers and is defined for the others [10].

As we have seen, Feferman holds that $\mathrm{K}^{(\beta)}$ should represent a correct axiomatization of Weyl's system. However, Avron objects that Weyl did not make any commitment to any finite order, rather he allowed for sets of arbitrary order. Moreover, this would be the case even if all the mathematics developed in *Das Kontinuum* would turn out to be formalizable in $\mathrm{K}^{(\beta)}$.

Avron critically discusses also the status of the Principle of Iteration and the alleged incoherence of Weyl's formulation spotted by Feferman. Avron's remarks strike at the root of Feferman's interpretation, since they are directed against the idea that Weyl meant to pursue something like an arithmetical mathematics program and challenge the identification of first-level definability as conceived by Weyl with arithemtical definability.

Indeed, if we break this identification, then Feferman's argument is blocked and thus the apparent contrast of the Principle of Iteration with the other foundational principles assumed by Weyl vanishes.

Under this perspective, the crucial task is to understand correctly what Weyl meant by first-level definability. According to Avron, the Principle of Iteration is fundamental for Weyl, since its justification stems from the very idea of iteration that is associated to the 'absolute' and primitive category of the natural numbers and is given directly in intuition. Also, on a more pragmatic level, Weyl intended to enlarge the stock of basic operations of his system in order to make it more expressive. Therefore, including the Principle of Iteration among the basic operations makes the collection of first level sets much wider. In this sense, first level definability is identified with definability from the basic principles of the system, that are recognized as justified in some way. Finally, under this reading, also sets and relations over the natural numbers that are not arithmetically definable are included.

Given this different understanding of first-level definability, it follows that there is no particular problem with the Principle of Iteration and, as a result, the alleged ambiguity about the strength of the induction principle accepted in *Das Kontinuum* is resolved. It follows from the Principle of Iteration that the induction principle in *Das Kontinuum* amounts to the full induction scheme and hence is stronger than the restricted induction principle that we have in systems such as $\mathbf{ACA_0}$ and $\mathrm{K}^{(\alpha)}$.

[10] Avron's formalization seems to be more faithful to Weyl's idea that every category of objects must include among its basic relations the identity ([168], p. 48).

The reason appears to be the following: once the Principle of Iteration is included among the basic principles of the system and is held to be justified, then it can be used to define higher-order objects. Then, we need a suitable form of induction that can be applied to these higher-order entities. It would have been quite strange to admit the possibility of defining this kind of objects in the system and then assume only a restricted form of induction, that does not apply to them. As a byproduct, this interpretation makes fully intelligible the points in *Das Kontinuum* where induction is applied to non-arithmetical formulas.

5.2.2 A Forgotten Interpretation of *Das Kontinuum*

In the last section, I have discussed the interpretation of *Das Kontinuum* given by Feferman as well as the objections to it and the alternative formalizations proposed by Adams and Luo and by Avron. In any event, Feferman's work on *Das Kontinuum* is usually held to be the first systematic investigation of Weyl's work. However, an earlier study of *Das Kontinuum* is contained in Ettore Casari's book *Questioni di Filosofia della Matematica* [20]. The book was published in 1964 and it is likely the first attempt to formalize Weyl's views. However, the book was written in Italian and has never been translated in other languages, therefore it did not have a broad circulation outside Italy and is never mentioned in the literature about *Das Kontinuum*. The book is an introduction to the main issues and the main schools in the philosophy and foundations of mathematics and consists of two parts: the first is devoted to the 'platonistic conceptions of mathematics', while the second discusses the 'conceptualist conceptions of mathematics'. The latter are subdivided into two main families that correspond to predicativism and intuitionism. Each chapter deals with the dialectical relationship between philosophical ideas and the corresponding formal developments. In particular, Chapter XII is focused on Weyl's predicative mathematics in *Das Kontinuum*. Casari mentions only a paper by Grzegorczyk [57] as an earlier study specifically devoted to Weyl's work. Nevertheless, this work is mainly meant to be a sharper reformulation of Weyl's predicative analysis within the context of what is nowadays called the minimal ω-model of $\mathbf{ACA_0}$, which is the model in which all second order quantifiers are assumed to range over those subsets of the natural numbers which are arithmetically definable. A similar approach was pursued by Mostowski in a paper republished in [109]. It seems that there is a consistent temporal gap between the publication of *Das*

Kontinuum and the appearance of the first systematic studies devoted to it. A similar conclusion is also confirmed by the informative and detailed historical reconstruction provided in [37]. Weyl's ideas certainly were a major influence on the early logical investigations of predicativity, but there was no attempt to give a precise formalization of Weyl's predicative framework. Therefore, it is likely that the formalization proposed by Casari represents the first work explicitly aimed at an axiomatization of the system presented in *Das Kontinuum*, since the other formalizations, which I have reviewed in the previous subsection, are all posterior to it.

The formal reconstruction takes place after a careful discussion of Weyl's methodology and philosophical attitude. Casari points out that Weyl's attitude is distinctly *anti-formalistic* and this point cannot be glossed over in any purported formalization of Weyl's system[11] Casari does not discuss further the exact meaning of Weyl's anti-formalism, but I think that the question can be elucidated as follows. It might be tempting to conclude that Weyl's attitude was modelled on Poincaré's position. However, Weyl had already shown a strong interest in the research in modern logic [163] before *Das Kontinuum* and did not seem to share Poincaré's prejudice against formalization. Therefore, it is likely that his views about the matter were more articulated than those held by Poincaré.

I claim that Weyl's anti-formalism comprehends a *positive* part, that concerns the role of intuition in mathematical thought, and a *negative* part, that is mainly focused on the *existence-as-consistency thesis*. Moreover, they are placed in the context of a more general view about the relationship between informal mathematical thought and formalization.

According to Weyl, the intuition of the sequence of natural numbers provides the minimal data that are necessary for any kind of abstract mathematical reasoning. The very idea of iteration is presupposed also by the systematic presentation of elementary logical inference rules. In this perspective, the role of intuition in acquiring mathematical knowledge and grasping mathematical concepts is absolutely fundamental. Concepts such as the sequence of natural numbers that are immediately justified by intuition cannot be reduced to others which are wrongly supposed to be more fundamental. Our epistemology of mathematics has to make room for this aspect, which cannot be confined

[11] G.Longo in his review [83] of [48] highlights this fundamental aspect of Weyl's thought (also in connection with the issue of formalizing our intuitive understanding of the physical continuum) and claims that it is neglected in Feferman's reconstruction.

to a purely psychological or heuristic dimension. It is undeniable that these views are consonant with Poincaré's beliefs. Nevertheless, it must not be taken for granted that Weyl's conception of intuition is essentially similar to that of Poincaré. They agree about the fundamental status of the sequence of natural numbers and about the fact that it is justified on an intuitive ground, but probably they meant different things by intuition and Weyl's view was influenced by Husserl's philosophy.
The negative component, instead, consists of the dismissal of the principle, famously held by Hilbert, that if a set of axioms is proved to be consistent, then this suffices for guaranteeing the existence of the entities that the axioms assert to exist. As it is well-known, this principle was linked to a view of the axioms as mere stipulations or uninterpreted formal conditions. Weyl was quite critical of this view in *Das Kontinuum*:

> "Today it is frequently maintained that axioms are *stipulations* and that, for example, Fermat's proposition ("there are no whole numbers $x \neq 0, y \neq 0, z \neq 0, n > 2$ such that $x^n + y^n = z^n$") asserts merely that this judgement is a *consequence* of the arithmetical axioms. Thus the axioms define, so to speak, the meaning of "there is". (That exists whose existence can be derived from the axioms). But quite apart from the fact that such a "hypothetico-deductive game" is worthless (if the axioms are devoid of any meaning which has cognitive significance) this view is even logically untenable." ([168], p.18)

and then he concluded

> "The interpretation under consideration proves to be feasible only when one knows that the axioms are *consistent* and *complete* [...] But we do not *know* this."
>
> (*ibid.*, p.19)

Then, he deemed that *completeness* of the axioms under consideration was a necessary requirement to make this conception of the axiomatic method going through. However, it may be possible that there is a sentence P such that neither P nor its negation can be derived from the axioms. In particular, Weyl was entertaining the possibility that the axiomatization of arithmetic could be *incomplete*, as has been pointed out also elsewhere [84, 9].

Ruling out this possibility amounts to show that the set of axioms is complete and, in turn, a completeness proof requires to set up a formal specification of the language of the theory as well as of its axioms and its underlying logic. But, according to Weyl, elementary logical steps already presuppose the grasp of a certain idea of iteration, which is based on the same insight that gives us the sequence of natural numbers. Therefore, even if this program would turn out to be feasible, we would be still committed to interpret at least arithmetical axioms in a contentful way, that is what intuition allows us to do.

The reflections on the axiomatic method and on the relationship between mathematical practice and axiomatization have occupied a special place in Weyl's overall production.

In the mature essay "Axiomatic versus Constructive Procedures in Mathematics" Weyl analyses the historical development of the role of the axiomatic method in mathematics:

> "Whereas the axiomatic method was formerly used merely for the purpose of elucidating the foundations on which we build, it has now become a tool for concrete mathematical research " ([167], p.13)

In this text, written around 1953, Weyl introduces a distinction between *transcendental axiomatics* and *immanent axiomatics*. The former is mainly focused on foundational and meta-theoretical issues (e.g. consistency and independence proofs). It requires that we abstract from the intuitive meaning of the terms occurring in the axioms. The latter, instead, is focused on the formulation of fundamental facts that hold for mathematical objects that may occur in different contexts. Issues such as completeness or consistency are not the main concern in this case, since 'the purpose of this sort of axiomatics is to *understand*; and what it reveals is a limited number of interrelated *structures*' ([167]. A masterful example of immanent axiomatics is provided by Emmy Noether's mathematical work [12].

It is true that Weyl's later writings are characterized by a more balanced attitude towards formalistic tendencies. Nonetheless, he is still far from wholeheartedly accepting a formalistic view. In this respect, I think that his viewpoint on transcendental axiomatics and formalism can be summarised as follows: the problems with transcendental axiomatics arise exactly when the

[12] In this respect, see also the third essay in the collection [169].

formal aspect is totally privileged at the expense of the intuitive component of mathematical understanding. What happens is that the axiomatic characterisation leads to disregard the intuitive meaning of basic mathematical concepts (e.g. the concept of natural number). As a result, some important conceptual distinctions are lost and the privileged status of some concepts is neglected. This should be contrasted with the use of constructive or genetic procedures.
In the end, Weyl's mature reflection is an attempt to combine the benefits of the axiomatic method with those deriving from the adoption of constructive or genetic procedures, where primitive notions are acknowledged as intrinsically meaningful:

> "*Large parts of modern mathematical research are based on a dexterous blending of constructive and axiomatic procedures.* One should be content to note their mutual interlocking. But temptation is great, and not all authors have resisted it, to adopt one of these two views as the genuine primordial way of mathematical thinking to which the other merely plays a subservient role" ([167], p.18)

In [20], Casari provides a detailed account of the methodology that must be followed in the reconstruction of mathematical theories, as advocated by Weyl in *Das Kontinuum*.
Any mathematical theory is built up from a stock of initial data, which is called *the sphere of operation* (*Operationsbereich*)[13]. A sphere of operation $\mathfrak{O}$ consists of some basic categories (types) equipped with some primitive relations [14], among which there is the identity relation associated to any basic category.
More formally, a sphere of operation $\mathfrak{O}$ is given by a signature $(A_i, R_k, =_{A_i})$ where

- $A_1, \ldots, A_n$, for $i \in \{1, \ldots, n\}$, are the basic categories;
- $R_1, \ldots, R_m$, for $k \in \{1, \ldots, m\}$, are the primitive relations over those categories;

[13]Here we follow the English translation proposed in [168]. Casari translates 'Operationsbereich' in Italian as *ambito operativo*.

[14]Casari follows Weyl in distinguishing between primitive *properties* and primitive relations. However, what is meant by property in *Das Kontinuum* is simply a predicate and hence a unary relation, so we can simplify and talk only of relations.

- $=_{A_1}, \ldots, =_{A_n}$ are the identity relations associated to the basic categories.

Therefore, every object is taken to belong to a category, and hence mathematical objects as conceived by Weyl are typed, and relations are prior to functions, since the initial signature includes only relational symbols.
The philosophical considerations reported above about the privileged status of the sequence of natural numbers entail that there must be a sphere of operation corresponding to this structure. Such a sphere of operation is called *absolute* by Weyl, because it is presupposed in the development of any other mathematical field. The absolute sphere of operation is the following:

$$\mathcal{N} = (\mathbb{N}, \mathtt{S}, =_{\mathbb{N}})$$

It consists of the category of natural numbers, the successor relation and the identity over $\mathbb{N}$.
Having fixed a sphere of operation $\mathfrak{O}$, the construction of mathematical entities takes place by means of two procedures, called by Weyl, respectively, the *logical* and the *mathematical* process:

- *Logical Process.* Given the primitive relations $R_1, \ldots, R_m$, that range over some fixed basic categories, the logical process allows us to build new relations (the *derived* ones) ranging over the same categories. These new relations are obtained by finitely many applications of the following elementary operations: (1) negation, (2) conjunction, (3) disjunction, (4) existential quantification, (5) identification of variables and (6) substitution. Since the underlying logic is classical, then the other operations corresponding to the missing logical constants can be easily defined.

- *Mathematical Process.* The Logical Process forms new relations by means of some elementary logical operations with a purely combinatorial nature, but we need also a way to introduce *sets* into the framework. The Mathematical Process is what fulfils this purpose. It consists of a comprehension principle that build sets starting from the available relations. The sets thus obtained must be thought of as *ideal* entities. Further, an extensionality principle is assumed stating when two ideal entities are equal. Finally, it must be observed that these ideal entities deriving from the application of the Mathematical Process will belong to new categories, that are fully determined by the categories to which the

> variables of the corresponding relations belong. The primitive notion of membership ε is assumed to hold between tuples of objects belonging to the basic categories and the newly defined sets, under the assumption that the relevant type restrictions are respected.

According to Casari's interpretation, Weyl held that the combination of these two procedures explains how the mathematical entities of a certain theory $\mathbb{T}$ are generated. Let us denote the sphere of operation on which $\mathbb{T}$ is based by $\mathfrak{O}_{\mathbb{T}}$. In $\mathfrak{O}_{\mathbb{T}}$, there are fixed primitive relations $R_1^{\mathbb{T}}, \ldots, R_m^{\mathbb{T}}$ (ranging over the basic categories of $\mathfrak{O}_{\mathbb{T}}$), from which we can get new derived relations $S_1^{\mathbb{T}}, \ldots, S_n^{\mathbb{T}}$ by applying the operations of the Logical Process. Let $\mathfrak{R}_{\mathbb{T}}$ be the whole system including the primitive relations of $\mathfrak{O}_{\mathbb{T}}$ as well as the derived ones. Then, we can apply the Mathematical Process to any relation in $\mathfrak{R}_{\mathbb{T}}$ and as result we introduce new sets in the theory. The sets thus obtained and all the relations in $\mathfrak{R}_{\mathbb{T}}$ represent the objects of the *first level*. However, it is in the nature of the two processes that they can be iterated. It follows that we can apply the Logical Process to all the first-level relations (including also the membership relation) and thus we get the second-level relations. Then, we can build new sets from these relations by applying the Mathematical Process and thus we are led to the sets of second level. In this way, objects of any finite level can be built and the predicativity constraint is respected, since the principles of construction are applied only to objects that have been already constructed.
This examination of Weyl's *genetic* conception of mathematical theories leads Casari to make the following claim:

> "In *Das Kontinuum* Weyl actually presents two systems of predicative mathematics, one practically equivalent to the ramified theory of types without the axiom of reducibility and another one much weaker." ([20], p.166)[15]

This is the main thesis underlying Casari's axiomatization. The two systems are called by Casari $\mathfrak{W}$ and $\mathfrak{W}^*$. The first is the result of the unrestricted iteration of the two processes and is the stronger system. The latter, instead, results from a suitable restriction of the generative procedure: existential

[15]It is noteworthy to observe that the idea that Weyl's system is nothing but a version of ramified theory of types without the Reducibility Axiom appears also in Church [22] and Quine [127], even if these authors do not further develop this claim.

quantification and substiution can be applied only to the entities belonging to the basic categories. Then, in the case of the absolute sphere of operation, we can quantify only over natural numbers, but not over sets of them. The weaker system $\mathfrak{W}^*$ is not explicitly presented by Casari. Nevertheless, he makes the important claim that Weyl's reconstruction of analysis takes place within it.
In the following, I give a formulation of the system $\mathfrak{W}$ that, modulo some changes in the notation and some minor syntactical modifications, corresponds to Casari's original presentation.

Definition 5.2.1 (Type Structure of $\mathfrak{W}$)**.** The types of $\mathfrak{W}$ are inductively defined as follows:

1. $\mathtt{0}$ is a type;

2. if $\tau_1, \ldots, \tau_n$ are types, then $(\tau_1, \ldots, \tau_n)$ is a type;

3. Nothing else is a type.

The type $\mathtt{0}$ is the type of individuals, but since every sphere of operation presupposes the absolute sphere $\mathcal{N}$, then we can interpret this basic type as the type of natural numbers $\mathbb{N}$. The type $(\tau_1, \ldots, \tau_n)$ is the type of propositional functions with arguments in types $\tau_1, \ldots, \tau_n$. If we take the empty sequence, then $(\ )$ is the type of propositions. The type structure defined above gives only simple types, but we can assign a natural number [16] to any type in order to get a ramified type structure:

Definition 5.2.2 (Ramified Type Structure of $\mathfrak{W}$)**.** The ramified types of $\mathfrak{W}$ are inductively defined as follows:

1. $\mathtt{0}$ is a ramified type with order zero;

2. if $\tau_1, \ldots, \tau_n$ are ramified types with orders $k_1, \ldots, k_n$ respectively, then $(\tau_1, \ldots, \tau_n)$ is a ramified type with order k such that, for each $i \in \{1, \ldots, n\}$, $k_i < k$;

3. Nothing else is a ramified type.

[16] It is possible to generalize the definition to ordinals, but is not necessary for our purposes.

In general, the order of a type τ will be denoted by $|\tau| = n$, where n is a natural number, and the order of a term will be the same as the order of the type to which it belongs.
Casari observes ([20], p.172) that Weyl distinguished between predicative symbols and set symbols in his system. Such a distinction is meant to reproduce at a syntactic level to the distinction between the Logical and the Mathematical Process. Consequently, the language of $\mathfrak{W}$ is designed in such a way that this distinction is preserved. However, Casari points out that predicative symbols possess a kind of special status in Weyl's system: they are introduced as a special kind of variables, but for all practical purposes they behave as constants, since they cannot be quantified over and cannot be substituted. In general, predicative symbols are thought by Weyl as abbreviations for complex sentences. A predicative symbol P is introduced as follows

$$P^{(\tau_1,\ldots,\tau_n)}(x_1^{\tau_1},\ldots,x_n^{\tau_n}) \leftrightarrow \phi(x_1^{\tau_1},\ldots,x_n^{\tau_n})$$

Since they are just abbreviations for the corresponding condition, then it would be natural to assume that they have the same order of the latter. However, this is not the case in general, in the light of the definition of ramified types given above. For instance, if $|\phi(x^\tau)| = |\tau| = k$, then the order of P turns out to be necessarily strictly greater than k, since the type of P is (τ). The way-out is the introduction of the concept of 'predicative' type:

Definition 5.2.3 (Predicative ramified type)**.** If $\tau_1,\ldots,\tau_n$ are ramified types with order $k_1,\ldots,k_n$ respectively and at least one $k_i \neq 0$, then $(\tau_1,\ldots,\tau_n)$ is a predicative ramified type with order k such that $k_i \leqslant k$ for each $i \in \{1,\ldots,n\}$.

The amendment amounts to allowing for the possibility that k be equal to k_i. Therefore now we can legitimately assign to P the same order as τ.
The language of $\mathfrak{W}$ includes the usual logical constants [17], a membership symbol ε, the equality symbol $=_\tau$ for any type τ, the set-forming abstraction operator $[\ |\ ]$ and infinitely many individual variables for any type τ as well as infinitely many predicative variables for any predicative type σ. Further, we assume a special symbol $\mathtt{S}$ for the successor relation.
Notation. In the following, the expression '$a : A$' denotes that a is a term of type A and 'ϕ is a k-formula' means that the well-formed formula ϕ has order k for $k \in \mathbb{N}$.

[17] Casari assumes from the very beginning also the logical constants which are not taken as primitive by Weyl in *Das Kontinuum*.

Definition 5.2.4 (Terms & Formulas of $\mathfrak{W}$)**.** The terms and the formulas of $\mathfrak{W}$ are simultaneously defined as follows:

- Every variable $x : \tau$ is a term of type τ;
- If $P : (\tau_1, \ldots, \tau_n)$ is a predicative variable, $|P| = k$ and $t_1 : \tau_1, \ldots, t_n : \tau_n$, then $P(t_1, \ldots, t_n)$ is a k-formula;
- If $t_1 : \tau_1, \ldots, t_n : \tau_n$, $s : (\tau_1, \ldots, \tau_n)$ and $|(\tau_1, \ldots, \tau_n)| = k$, then $t_1, \ldots, t_n \; \varepsilon \; s$ is a k-formula;
- If $s, t : \tau$ and $|\tau| = k$, then $t \; =_\tau \; s$ is a k-formula;
- If $n, m : \mathtt{0}$, then $\mathtt{S}(n, m)$ is a 1-formula;
- If ϕ is a k-formula, then $\neg\phi$ is a k-formula;
- If ϕ is a k_1-formula and ψ is a k_2-formula then $\phi * \psi$ is a k-formula, where $* \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ and $k = \mathsf{max}(k_1, k_2)$;
- If $\phi(x^\tau)$ is a k_1-formula, $x : \tau$ and $|\tau| = k_2$, then $\mathcal{Q}x^\tau.\phi(x^\tau)$ is a k-formula, where $k = \mathsf{max}(k_1, k_2 + 1)$ and $\mathcal{Q} \in \{\forall, \exists\}$;
- If $\phi(x_1^{\tau_1}, \ldots, x_n^{\tau_n})$ is a k-formula and $|\tau_1| = k_1, \ldots, |\tau_n| = k_n$, then

  $$[\; x_1^{\tau_1}, \ldots, x_n^{\tau_n} \mid \; \phi(x_1^{\tau_1}, \ldots, x_n^{\tau_n}) \;] : (\tau_1, \ldots, \tau_n)$$

  is a term with order $j = \mathsf{max}(k, k_1 + 1, \ldots, k_n + 1)$.

The deductive apparatus of $\mathfrak{W}$ consists of an axiomatization of many-sorted classical logic with equality plus the following axioms:

(a) Comprehension:

$$t_1, \ldots, t_n \; \varepsilon \; [x_1^{\tau_1}, \ldots, x_n^{\tau_n} \mid \phi(x_1^{\tau_1}, \ldots, x_n^{\tau_n})] \; \leftrightarrow \; \phi(t_1/x_1^{\tau_1}, \ldots, t_n/x_n^{\tau_n})$$

provided that $t_1 : \tau_1, \ldots, t_n : \tau_n$.

(b) Extensionality:

$$\forall x_1^{\tau_1}, \ldots, x_n^{\tau_n}.(\phi(x_1^{\tau_1}, \ldots, x_n^{\tau_n}) \; \leftrightarrow \; \psi(x_1^{\tau_1}, \ldots, x_n^{\tau_n})) \; \rightarrow$$

$$[\; x_1^{\tau_1}, \ldots, x_n^{\tau_n} \mid \phi(x_1^{\tau_1}, \ldots, x_n^{\tau_n})] \; = \; [\; x_1^{\tau_1}, \ldots, x_n^{\tau_n} \mid \psi(x_1^{\tau_1}, \ldots, x_n^{\tau_n})]$$

(c) If $\phi(x_1^{\tau_1}, \ldots, x_n^{\tau_n})$ is a k-formula, then

$$P^{(\tau_1,\ldots,\tau_n)}(x_1^{\tau_1}, \ldots, x_n^{\tau_n}) \leftrightarrow \phi(x_1^{\tau_1}, \ldots, x_n^{\tau_n})$$

with $|P| = k$, for P predicative symbol.

(d) $\forall x^0\ \exists y^0\ \mathtt{S}(x^0, y^0)$

(e) $\forall x^0\ \forall y^0\ \forall z^0\ (\mathtt{S}(x^0, y^0)\ \wedge\ \mathtt{S}(x^0, z^0)\ \rightarrow\ y^0 = z^0)$

(f) $\exists!\ y^0\ \forall x^0\ \neg\mathtt{S}(x^0, y^0)$

(g) $\forall x^0\ \forall y^0\ \forall z^0\ (\mathtt{S}(y^0, x^0)\ \wedge\ \mathtt{S}(z^0, x^0)\ \rightarrow\ y^0 = z^0)$

(h) Full Induction: for any k-formula ϕ with variables in $\mathtt{0}$,

$$\phi(0) \wedge (\forall n, m : \mathtt{0}\ \mathtt{S}(n, m) \wedge \phi(n) \rightarrow \phi(m)) \rightarrow \forall n : \mathtt{0}\ \phi(n)$$

where 0 is a new term of type $\mathtt{0}$ satisfying axiom (f).

The full induction scheme is justified by the Principle of Iteration, since it represents the inferential counterpart of this fundamental principle of definition ([168], p.39).
The formulation of the Principle of Iteration in $\mathfrak{W}$ is quite complex and Casari remarks that the way this principle is stated by Weyl involves some ambiguities. However, he maintains that the meaning of the principle is clear: 'justifying inductive definitions within any category' ([20], p. 179).
Casari conceives the principle as a principle of definition concerning predicative symbols: it yields a sequence of inductively defined properties $P_1, \ldots, P_n$ with arbitrary type. Given a sequence of formulas $\phi_1, \ldots, \phi_n$ with order k and with the following variables free:

$$\phi_1(x_{11}^{\tau_{11}}, \ldots, x_{1m}^{\tau_{1m}}, z_1^{(\tau_{11},\ldots,\tau_{1m})}, \ldots, z_n^{(\tau_{n1},\ldots,\tau_{nm})}, y^0)$$

$$\vdots$$

$$\phi_n(x_{n1}^{\tau_{n1}}, \ldots, x_{nm}^{\tau_{nm}}, z_1^{(\tau_{11},\ldots,\tau_{1m})}, \ldots, z_n^{(\tau_{n1},\ldots,\tau_{nm})}, y^0)$$

For each i such that $1 \leqslant i \leqslant n$, the whole set of free variables of ϕ_i is denoted by v_i. Let $t_1, \ldots, t_n$ be the abstraction terms (with order k, according our definition) that are obtained as follows:

$$\begin{aligned} t_1 &:\equiv [x_{11}^{\tau_{11}}, \ldots, x_{1m}^{\tau_{1m}} | \phi_1] \\ &\vdots \\ t_n &:\equiv [x_{n1}^{\tau_{n1}}, \ldots, x_{nm}^{\tau_{nm}} | \phi_n] \end{aligned}$$

Then, for predicative symbols $P_1, \ldots, P_n$ with suitable types and order k, the following expressions are axioms:

$$\begin{aligned} P_1(1, v_1) &\leftrightarrow \phi_1 \\ &\vdots \\ P_n(1, v_n) &\leftrightarrow \phi_n \end{aligned}$$

$$\begin{aligned} P_1(\mathsf{succ}(m), v_1) &\leftrightarrow P_1(m, v_1[t_1/z_1, \ldots, t_n/z_n, m/y]) \\ &\vdots \\ P_n(\mathsf{succ}(m), v_n) &\leftrightarrow P_n(m, v_n[t_1/z_1, \ldots, t_n/z_n, m/y]) \end{aligned}$$

For the sake of clarity, I have assumed a primitive functional symbol for the successor. Namely, this should entail the formulation of the corresponding term-formation clause as well as a suitable reformulation of the corresponding axioms. Finally, the variable y ranging over the basic type of natural numbers could be missing.

The variables occurring in the expressions ϕ_i and, accordingly, in the predicative symbols P_i, might be of arbitrarily complex types. The Principle of Iteration as formulated by Casari allows for the introduction of inductively defined predicates ranging over arbitrary types and having arbitrary orders. In this sense, it provides a way to introduce higher-order objects within the system. If this formulation of the principle is taken to be correct, then it seems right to conclude that Weyl would not have accepted any restriction to the full induction principle. Indeed, such a principle is necessary to establish facts about the higher-order objects that are introduced by the Principle of Iteration. As a result, the ambiguity about the strength of the induction principle accepted in *Das Kontinuum* remarked by Feferman disappears and we can make sense of the claim that full induction is *justified* by the Principle of Iteration.

Casari's interpretation of the Principle of Iteration and of its role in Weyl's system should be compared with the analysis proposed by Avron [9]. They both agree on the reconstruction of the relation between the iteration principle and the induction axiom as conceived by Weyl. Further, they do not hold that

this principle is incoherent with the general foundational aims underlying *Das Kontinuum*. Rather, it is dictated by Weyl's philosophical views. Accordingly, Weyl's notion of predicative definability is held to be more extensive than arithmetical definability. However, it would be a mistake to neglect the differences between these two interpretations, that in fact result in different axiomatizations.

Casari justifies the introduction of predicative symbols by appealing to the different natures of the Logical and the Mathematical processes. It is such a distinction that sets the ground for the introduction of predicative symbols as distinct from set symbols. Then, the formulation of the Principle of Iteration as a scheme involving the former implies that Casari assumes such a principle as part of the Logical Process. However, Weyl presents this principle as a 'specifically mathematical one' ([168], p.38) and hence as distinguished from the other *logical* principles of constructions. Avron ([9], p.51) explicitly points out this aspect in his analysis.

More generally, Casari's system $\mathfrak{W}$ is motivated by the description given by Weyl of the unrestricted iteration of the two generative processes. System $\mathfrak{W}$ is the only formalization of Weyl's system, among all that which can be found in the literature, which relies upon ramification.

Weyl himself acknowledged that his generative procedure bore a resemblance to Russell's ramified hierarchy:

> "Russell's theory of logical types corresponds to the formation of levels mentioned in § 6 and is motivated by his 'vicious-circle principle' " ([168], p.48).

On the other hand, it is also true that Weyl explicitly rejected a ramified version of analysis that would lead to real numbers and real-valued functions of different orders:

> "A "hierarchical" version of analysis is artificial and useless [. . .] In other words, the only natural strategy is *to abide by the narrower iteration procedure*" (*ibid.*, p.32)

Therefore, the ramified version of analysis is replaced by a treatment based on the narrower iteration procedure. In this procedure, substitution and quantification are restricted to terms of the basic category of natural numbers. Under the assumption of this restriction, we stick to the first levels of the ramified hierarchy and thus the articulation into levels is superfluous.

At least *prima facie*, these remarks are in harmony with the thesis that in *Das Kontinuum* there are two systems: the unrestricted procedure leads to $\mathfrak{W}$, which is ramified, whereas the narrow one leads to $\mathfrak{W}^*$, which is not ramified. Furthermore, the development of real analysis takes place in $\mathfrak{W}^*$. This is essentially the interpretation advocated by Casari.
A problem for this interpretation is again posed from the Principle of Iteration. In particular, in section §8 of *Das Kontinuum* (*Definitive Formulation of the Foundations. Introduction of Ideal Elements*) Weyl seems to hold that the introduction of the Principle of Iteration entitles us to dispense altogether with ramification:

> "We see now, given that the principles of substitution and iteration are to be added, that we can no longer adhere to the notion of a production of relations and corresponding sets in separate *levels*." (*ibid.*, p.40)

Further, the paragraph ends with a withdrawal of all the remarks made in the previous sections §4-7. The interpretation provided by Avron is that the Principle of Iteration is meant by Weyl to do the work of the iteration of the two fundamental constitutive processes, that is what yields the ramified structure. In light of the passage quoted above, Avron's interpretation appears to be more faithful to Weyl's intent and indeed is mainly based on this paragraph of Weyl's book.
While Casari is surely right in maintaining that ramification is not needed in the case of the restricted procedure and thus in the reconstruction of analysis, however Weyl's rejection of ramification does not seem to be confined to this special case: the full system presented by Weyl is not ramified.
The reason suggested by the previous quote is a trade-off between the iteration of the unrestricted procedure and the Principle of Iteration, as advocated by Avron. Including the Principle of Iteration allows for the introduction of higher-order objects and thus makes superfluous the iteration of the Logical and Mathematical processes, since they lose their raison d'être. Further evidence for the correctness of this reading comes from a later publication by Weyl, in which he describes the role of the Principle of Iteration in his system as follows:

> "In this system iteration plays the role which in set theory was played by the uninhibited application of quantifiers." ([166], p.9)

At the same time, it should be noted that in no place does Weyl reject the full iteration of the two processes as if it were in contrast with his foundational principles. In particular, the unrestricted iteration of the processes and the resulting organization of sets into orders still counts as predicative in Weyl's sense. Weyl simply prefers the system with the Principle of Iteration, since it allows a more natural formalization of mathematical notions without compromising the expressive strength of the system. In this respect, we can conclude that the system $\mathfrak{W}$ is not the most faithful reconstruction of the system presented in *Das Kontinuum*, but it is a quite accurate formalization of the Logical and Mathematical processes as conceived by Weyl.
Other peculiar features of the system $\mathfrak{W}$, when compared to the other systems proposed as formalization of Weyl's original framework, is the treatment of extensionality and the absence of functional variables.
All the interpreters agree on the fact that the system of *Das Kontinuum* is extensional to some extent. Casari makes the following interesting remark about this question:

> "As for the principle of extensionality, the faithful translation of that proposed by Weyl appears to be weaker than the usual version." ([20], p.177)

Recall that the axiom of extensionality in $\mathfrak{W}$ is formulated as follows:

$$(\texttt{ext})\ \forall x^{\tau}(\phi(x^{\tau})\ \leftrightarrow\ \psi(x^{\tau}))\ \rightarrow [\ x^{\tau}\mid \phi(x^{\tau})]\ =\ [\ x^{\tau}\mid \psi(x^{\tau})]$$

On the other hand, the 'usual version' of extensionality Casari is referring to is the following:

$$(\texttt{Ext})\ \forall x^{\tau}\ (x^{\tau}\varepsilon\ w^{(\tau)} \leftrightarrow x^{\tau}\varepsilon\ z^{(\tau)})\ \rightarrow\ w^{(\tau)} = z^{(\tau)}$$

Then, Casari's claim is that there is no way of deriving `Ext` from `ext` in $\mathfrak{W}$ unless we assume $[\ x^{\tau}\mid x^{\tau}\varepsilon\ z^{(\tau)}\ ] = z^{(\tau)}$, which apparently cannot be derived in $\mathfrak{W}$.
Let us call η[18] the axiom asserting $[\ x^{\tau}\mid x^{\tau}\varepsilon\ z^{(\tau)}\ ] = z^{(\tau)}$, then the following holds:

[18]The name is inspired by the fact that it corresponds to a kind of η-rule. Indeed, if we regard the set-forming operator as λ-*abstraction* and the membership ε as *application*, then we get $\lambda x.z(x) = z$, that is exactly the rule called η in the literature about λ-calculus and type theory. Moreover, there are other examples in the literature of exensionality axioms whose derivation involves exactly this rule.

Proposition 5.2.5. $\mathfrak{W} + \eta \vdash \texttt{Ext}$.

Proof. We can restrict to the case with one free variable, since the generalization is straightforward. Let $\phi(w)$ be the formula $x^\tau \varepsilon\ w^{(\tau)}$ and let $\psi(z)$ be $x^\tau \varepsilon\ z^{(\tau)}$. Let us assume $\forall x^\tau (x^\tau \varepsilon\ w^{(\tau)} \leftrightarrow x^\tau \varepsilon\ z^{(\tau)})$, then by (ext)

$$[\ x^\tau \mid x^\tau\ \varepsilon\ w^{(\tau)}\ ] = [\ x^\tau \mid x^\tau\ \varepsilon\ z^{(\tau)}\ ]$$

and by applying the axiom η to both sides of the equality, we get $w^{(\tau)} = z^{(\tau)}$. Therefore, by discharging the initial assumption, we get a proof of (Ext). □

Moreover, we can prove also the following:

Proposition 5.2.6. $\mathfrak{W} + \texttt{Ext} \vdash \eta$.

Proof. $x\ \varepsilon\ [\ x \mid x\ \varepsilon\ z\ ] \leftrightarrow x\ \varepsilon\ z$ holds as instance of the comprehension axiom, then by applying Ext we get $[\ x \mid x\ \varepsilon\ z\ ] = z$. □

Remark 5.2.7. It must be observed that principles similar to the axiom η are assumed both in Avron's **WA** and Adams and Luo's $\mathbf{LTT_W}$.

The language of $\mathfrak{W}$ does not contain function variables and, as I remarked above, this is another important difference between system $\mathfrak{W}$ and the other systems proposed as formalization of the system of *Das Kontinuum*.
Namely, functions may be introduced in $\mathfrak{W}$ as functional relations in a pretty standard manner, but Weyl quite explicitly contrasts his notion of function with this standard definition ([168], p.33). According to the former, a function is always determined by a *rule* and thus it has a strong operational or constructive flavour. Avron has thoroughly analyzed Weyl's conception of functions (see section 4.3 in [9]). There, a compelling case is made for including functions among the basic objects of *Das Kontinuum* mathematical universe. Accordingly, any faitfhul formalization of Weyl's system must include functional variables as well as term formation conditions involving functions. Further, Avron remarks that Weyl's notion of function is restricted to those functions having as their codomain a category of sets, i.e. a type of the form $\mathrm{Set}(\tau_1 \times \ldots \times \tau_n)$. It follows that functions mapping into the basic category of natural numbers must always be treated as relations. Finally, functions are extensional, in the sense that two functions are equal if they yield equal values for every element of the domain. Weyl's remarks on his notion of function are very scattered; nevertheless, there seems to be a tension

between the idea that a function is always determined by a rule and the assumption that functions are extensional.
Under this reading, it seems to be a defect of system $\mathfrak{W}$ not including function variables in the language. On the other hand, it must be noted that Casari's discussion is limited to functions having the natural numbers as codomain (and defined by means of the Principle of Iteration), which are correctly treated as relations. While there is no attempt in [20] to give a formal treatment within $\mathfrak{W}$ of functions having other types as codomains (e.g. the type of real numbers).

5.3 The Iteration of the Logico-Mathematical Process

The aim of the present section is to analyze in a more modern fashion the result of the iteration of the two generative processes, that represents the core of Casari's reading of *Das Kontinuum*, and to investigate the resulting theory within the context of modern type theory.

The Logico-Mathematical Process reconsidered. According to Weyl, every mathematical theory is based on a system consisting of a primitive domain of objects and some relations, which must include the equality relation over the domain. Further, the natural number system must be taken for granted, since it is the absolute sphere of operation that is presupposed by any other. The Logico-Mathematical Process consists in constructing new formulas out from these primitive data by means of the standard logical operations and then in forming the corresponding sets by applying the comprehension principle and hence by *abstraction*. We assume that the language contains the logical symbols $\bot, \wedge, \vee, \rightarrow, \exists, \forall$. The most conservative choice is to start from the system $\langle \mathbf{N}, \mathsf{succ}, =_{\mathbf{N}} \rangle$. The formulas obtained by applying to this system the logical operations are called formulas of level 0. In such formulas, quantifiers range only over $\mathbf{N}$. The type of formulas of level 0 is Ω_0. Given a formula ϕ in Ω_0 containing a free variable $x : \mathbf{N}$, we can form the abstraction term $[x : \mathbf{N} \mid \phi]$, that is of type $\mathsf{Set}_0(\mathbf{N})$. Such a type must be understood as the type of sets of natural numbers. In general, whenever A is a type, we can always form the type $\mathsf{Set}_n(A)$ for a given $n \in \mathbb{N}$.
Sets are conceived as *ideal* entities in this framework. They are essentially

obtained by abstraction from a certain formula ϕ and arise from looking at ϕ in an *extensional* way. This justifies the following *extensionality axiom*:

$$X =_{\mathsf{Set}_n(A)} Y \leftrightarrow (\forall z : A)(z \ \varepsilon \ X \leftrightarrow z \ \varepsilon \ Y)$$

The membership symbol 'ε ' is assumed as primitive and takes as arguments terms of type A and $\mathsf{Set}(A)$. If we assume an intensional view and consider the conditions determining the corresponding sets, then it arises from taking an extensional perspective on the application of these conditions to arguments of type A.

The introduction of sets of level 0 means that we have new types in the system to which we may apply the logical operations. The new formulas that are obtained in this way are the formulas of level 1 and we can introduce the corresponding type Ω_1. The sets of level 1 are built by abstraction from these formulas. The process can be iterated for every n[19] and thus we have a sequence of types Ω_n and $\mathsf{Set}_n(A)$ for each $n \in \mathbb{N}$.

Observing the unfolding of the Logico-Mathematical Process highlights the intensional nature of the conception of the mathematical universe underlying it. The mathematical universe consists primarily of properties and relations that are intensional entities. Sets, instead, are regarded as entities that are *derived* from them and are not taken to be ontologically primitive. The extensional point of view, though convenient for the mathematical practice, is derived from the intensional one and entities considered in extensions are not fundamental in the resulting description of the mathematical universe: they are introduced only after that the corresponding properties have been constructed.

The iteration of the Logico-Mathematical Process follows quite faithfully a '*constructional metaphor*' ([128], p.243): all the entities are generated in stages starting from a primitive system of objects and relations. Moreover, we can explicitly keep track of the stage at which an entity is introduced thanks to the ramification device and the process ensures that the principles of construction are applied only to entities that are already available. Impredicative definitions are avoided since at any stage n the construction process is concerned only with entities of the previous stages. From this perspective, the ramified hierachy resulting from the iteration of the Logico-Mathematical Process

[19]It would be possible to consider also transfinite iterations of the process. I refrain from doing so, because it is not necessary for my purposes and also for avoiding the complications arising from the need of showing that the resulting system is predicatively justified.

is regarded as an idealized temporal process of construction that at each stage n makes use only of 'material' coming from stages earlier than n. As a consequence, the idea that the predicative mathematical universe is *open-ended* is faithfully respected: quantification over the universe is meaningless, because it would amount to quantifying over entities that have not been constructed yet and hence to considering the collection of entities generated by the process as a *closed* totality.

Remark 5.3.1. Sometimes a kind of constructional metaphor is invoked to show that the axioms of ZF are justified after all on the basis of the *iterative conception*. However, in this case the value of the metaphor is quite limited, if not misleading. The constructivist attitude towards the iterative conception forces us to interpret the metaphor in a literal way: at a stage t, a new set can be constructed only by reference to those sets that have been constructed at earlier stages. This makes it difficult to justify all the axioms of ZF and, in particular, the separation axiom. The formula ϕ appearing in the latter is unbounded and might well vary over the whole hierarchy of sets: the separation axiom is an impredicative principle. Under the assumption of the constructivist standpoint, only a restricted form of separation can be justified, which (unsurprisingly) is a predicative reformulation of the original principle. Similar problems arise in connection with the powerset axiom. Therefore, it is doubtful whether the constructional metaphor has any value in this context. On the other hand, the ramified hierarchy allows us to take the metaphor literally and in fact the ramified comprehension principle is immediately justified on this reading, after that the rules of term formation and the conventions over levels are made explicit. Then, the constructional metaphor maintains an epistemological value: it provides a motivation for the axioms of the system and for their correctness.

Remark 5.3.2. It would be legitimate to include also *finite* types among the basic types of the generative process. They are surely extensionally determinate, according to Weyl's criteria, and quantification over them is trivially unproblematic and meaningful. Therefore, we may include types such as the unit type **1** or the type of Booleans **2**.

The systems $\mathfrak{LM}^i$ *and* $\mathfrak{LM}$. A natural question arising in connection with the Logico-Mathematical Process is whether the logic underlying it should be intuitionistic or classical. As it is well-known, Weyl in *Das Kontinuum* maintained that classical quantification over extensionally determinate collections

is meaningful and legitimate. From this perspective, statements containing quantifiers which range over the integers have a definite truth-value. However, the device of ramification allows us to quantify also over sets of higher level without violating the predicative nature of the system. Then, classical quantification could be safely extended also to higher types. On the other hand, there could be also valid reasons to opt for intuitionistic logic; in particular, if it is held that it supports a wider notion of predicativity. As we will see, the intuitionistic system resulting from the Logico-Mathematical Process is, at least in some respects, more flexible than its classical counterpart.
In the following, I present the two systems resulting from the formalization of the Logico-Mathematical Process as described in the opening of this section: the intuitionistic system $\mathfrak{LM}^i$ and the classical system $\mathfrak{LM}$.

Definition 5.3.3 ($\mathfrak{LM}^i$-Types)**.** The ground types are

- $\mathbf{N}$
- for each $n \in \mathbb{N}$, Ω_n

and types are closed under $\times$ and $\mathsf{Set}_n(-)$, i.e.

- if A and B are types, then $A \times B$ is a type
- if A is a type, then $\mathsf{Set}_n(A)$ is a type.

Definition 5.3.4 (Orders)**.** The map ℓ associating to each type symbol of $\mathfrak{LM}^i$ a natural number, which is the level or the order of the types, is defined as follows:

- $\ell(\mathbf{N}) = 0$
- $\ell(\Omega_n) = n + 1$
- $\ell(A \times B) = \mathsf{max}(\ell(A), \ell(B))$
- $\ell(\mathsf{Set}_n(A)) = \mathsf{max}(\ell(A), n + 1)$

Definition 5.3.5 ($\mathfrak{LM}^i$-Terms)**.** The terms are formed according to the following rules:

- for any type symbol A, countably infinite variables $x_1, x_2, x_3, \ldots$ of type A.
- $0 : \mathbf{N}$ and if $a : \mathbf{N}$, then $\mathsf{succ}(a) : \mathbf{N}$.
- if $a : A, b : B$, then $(a, b) : A \times B$ and if $d : A \times B$, then $\mathsf{pr}_1(d) : A$ and $\mathsf{pr}_2(d) : B$.
- if $a, b : A$ and $\ell(A) \leqslant n$, then $a =_A b : \Omega_n$.
- $\bot : \Omega_n$.
- If $p, q : \Omega_n$, then $p \wedge q : \Omega_n$.
- If $p, q : \Omega_n$, then $p \vee q : \Omega_n$.
- If $p, q : \Omega_n$, then $p \to q : \Omega_n$.
- If $x : A$ and $p : \Omega_n$ and $\ell(A) \leqslant n$, then $(\exists x : A)p : \Omega_n$.
- If $x : A$ and $p : \Omega_n$ and $\ell(A) \leqslant n$, then $(\forall x : A)p : \Omega_n$.
- If $x : A$ and $p : \Omega_n$, then $[x : A \mid p] : \mathsf{Set}_n(A)$.
- If $a : A$ and $D : \mathsf{Set}_n(A)$, then $a \; \varepsilon \; D : \Omega_m$ (where $n \leqslant m$).

Terms of type Ω_n are called *formulas* of order n. Orders are assumed to be cumulative: if $D : \mathsf{Set}_n(A)$, then $D : \mathsf{Set}_{n+1}(A)$ and if $p : \Omega_n$, then $p : \Omega_{n+1}$. The rules and the axioms of $\mathfrak{LM}^i$ are the following: the logical rules are assumed to include structural rules and the identity axioms; I omit the type in giving the logical rules

Logical Rules

$$\frac{\Gamma \vdash p \qquad \Gamma \vdash q}{\Gamma \vdash p \wedge q} \qquad \frac{\Gamma \vdash p \wedge q}{\Gamma \vdash p} \qquad \frac{\Gamma \vdash p \wedge q}{\Gamma \vdash q}$$

$$\frac{\Gamma, p \vdash q}{\Gamma \vdash p \to q} \qquad \frac{\Gamma \vdash p \to q \qquad \Gamma \vdash p}{\Gamma \vdash q}$$

$$\frac{\Gamma \vdash p}{\Gamma \vdash p \vee q} \qquad \frac{\Gamma \vdash q}{\Gamma \vdash p \vee q} \qquad \frac{\Gamma \vdash p \vee q \qquad \Gamma, p \vdash r \qquad \Gamma, q \vdash r}{\Gamma \vdash r}$$

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash p}$$

$$\frac{\Gamma \vdash p}{\Gamma \vdash (\forall x : A)p}\,(x \text{ not free in } \Gamma) \qquad \frac{\Gamma \vdash (\forall x : A)p}{\Gamma \vdash p\,[t/x]}\,(t : A)$$

$$\frac{\Gamma \vdash p\,[t/x]}{\Gamma \vdash (\exists x : A)p}\,(t : A) \qquad \frac{\Gamma \vdash (\exists x : A)p \qquad \Gamma, p \vdash q}{\Gamma \vdash q}\,(x \text{ not free in } \Gamma \text{ and in } q)$$

$$\frac{}{\vdash t =_A t} \qquad \frac{\Gamma \vdash s =_A t \qquad \Gamma \vdash p\,[s/x]}{\Gamma \vdash p\,[t/x]}$$

Axioms for $\times$

- $x : A, y : B \vdash \mathsf{pr}_1((x, y)) =_A x$.
- $x : A, y : B \vdash \mathsf{pr}_2((x, y)) =_B y$.
- $z : A \times B \vdash (\mathsf{pr}_1(z), \mathsf{pr}_2(z)) =_{A\times B} z$.

Axioms for $\mathbf{N}$

- $x : \mathbf{N} \vdash \mathsf{succ}(x) =_{\mathbf{N}} 0 \to \bot$.
- $x : \mathbf{N}, y : \mathbf{N} \vdash \mathsf{succ}(x) =_{\mathbf{N}} \mathsf{succ}(y) \to x =_{\mathbf{N}} y$.
- $x : \mathbf{N}, y : \mathbf{N} \vdash x =_{\mathbf{N}} y \to \mathsf{succ}(x) =_{\mathbf{N}} \mathsf{succ}(y)$.
- $p : \Omega_n \vdash p(0) \wedge (\forall x : \mathbf{N})(p(x) \to p(\mathsf{succ}(x))) \to (\forall x : \mathbf{N})p(x)$.

Axioms of Extensionality and Comprehension

- $X : \mathsf{Set}_n(A), Y : \mathsf{Set}_n(A) \vdash X =_{\mathsf{Set}_n(A)} Y \leftrightarrow (\forall z : A)(z \;\varepsilon\; X \leftrightarrow z \;\varepsilon\; Y)$
- $p : \Omega_n, q : \Omega_n \vdash (p =_{\Omega_n} q) \leftrightarrow (p \leftrightarrow q)$
- $\Gamma, z : A \vdash z \;\varepsilon\; [x : A \mid p] \leftrightarrow p[z/x]$
  where Γ contains the free variables of p.

The classical system $\mathfrak{LM}$ is obtained by extending the logical axioms given above with the excluded middle $(\forall p : \Omega_n)\ p \vee \neg p$ or an equivalent classical principle.

Remark 5.3.6. It should be noted that, contrary to other presentations of ramified type systems, the unicity of type property fails, because we have assumed the cumulativity conditions given above. On the other hand, it still holds that every type is inhabited, as can be checked by induction over the structure of types.

Remark 5.3.7. The system does not contain a primitive function type. The notion of function is given by total functional relations. Given $D : \mathsf{Set}_i(A), E : \mathsf{Set}_j(B)$, then $F(D, E) : \mathsf{Set}_k(A \times B)$ is said to be a map from D to E if the following conditions are satisfied:

(i) $(\forall x : A)(\forall y : B)((x, y)\ \varepsilon\ F \to (x, y)\ \varepsilon\ D \otimes E)$
where $D \otimes E\ :\equiv\ [(x, y) : A \times B \mid x\ \varepsilon\ D \wedge y\ \varepsilon\ E]$

(ii) $(\forall x : A)(\forall y : B)(\forall y' : B)((x, y)\ \varepsilon\ F \wedge (x, y')\ \varepsilon\ F \to y =_B y')$

(iii) $(\forall x : A)(x\ \varepsilon\ D \to (\exists y : B)(y\ \varepsilon\ E \wedge (x, y)\ \varepsilon\ F))$

The interpretation in HoTT. I am going to show that the system $\mathfrak{LM}^i$ can be soundly interpreted in **HoTT** with propositional truncation as primitive type and a cumulative hierarchy of univalent universes $\mathcal{U}_i$, for each external natural number i. Moreover, the interpretation extends to $\mathfrak{LM}$ if LEM is assumed in **HoTT**.
The idea underlying the interpretation is that any $\mathfrak{LM}^i$-type A is interpreted as a pair (A°, s_A), where $A^\circ : \mathcal{U}_{\ell(A)}$ and $s_A : \mathsf{isSet}(A^\circ)$. Therefore, the level of the ramified hierarchy is sent to the corresponding universe level. Accordingly, a term $t : A$ in $\mathfrak{LM}^i$ is interpreted as $t^\circ : A^\circ$. In particular, terms of type Ω are interpreted as pairs: $(p : \Omega_n)^\circ$ amounts to $p^\circ : \mathsf{Prop}_{\mathcal{U}_n}$ and thus as a pair, where the first component is a type in $\mathcal{U}_n$ and the second is a proof that this type is an h-proposition.

Definition 5.3.8. We define an interpretation map $(-)^\circ$ from $\mathfrak{LM}^i$-types to **HoTT**-types as follows:

- $(\mathbf{N})^\circ :\equiv (\mathbb{N}, s_{\mathbb{N}})$

- $(\Omega_n)^\circ :\equiv (\mathsf{Prop}_{\mathcal{U}_n}, s_{\mathsf{Prop}_{\mathcal{U}_n}})$
- $(A \times B)^\circ :\equiv (A^\circ \times B^\circ, s_{A\times B})$
- $(\mathsf{Set}_n(A))^\circ :\equiv (A^\circ \to \mathsf{Prop}_{\mathcal{U}_n}, s_{A^\circ\to\mathsf{Prop}_{\mathcal{U}_n}})$

The interpretation extends to terms as follows:

- $(x : A)^\circ :\equiv x : A^\circ$
- $(0 : \mathbf{N})^\circ :\equiv 0 : \mathbb{N}$
- $(\mathsf{succ}(a)) : \mathbf{N})^\circ :\equiv \mathsf{succ}(a^\circ) : \mathbb{N}$
- $(a\ \varepsilon\ b)^\circ :\equiv b^\circ(a^\circ)$
- $(a =_A b)^\circ :\equiv (Id^\circ_A(a^\circ, b^\circ), p_{\mathrm{Id}})$
- $(\bot)^\circ :\equiv (\mathbf{0}, p_{\mathbf{0}})$
- $(p \wedge q)^\circ :\equiv (\mathsf{pr}_1(p^\circ) \times \mathsf{pr}_1(q^\circ), p_\times)$
- $(p \vee q)^\circ :\equiv (||\mathsf{pr}_1(p^\circ) + \mathsf{pr}_1(q^\circ)||, p_\vee)$
- $(p \to q)^\circ :\equiv (\mathsf{pr}_1(p^\circ) \to \mathsf{pr}_1(q^\circ), p_\to)$
- $((\exists x : A)p)^\circ :\equiv (||\Sigma_{(x:A^\circ)}\mathsf{pr}_1(p^\circ)||, p_\exists)$
- $((\forall x : A)p)^\circ :\equiv (\Pi_{(x:A^\circ)}\mathsf{pr}_1(p^\circ), p_\Pi)$
- $((a, b) : A \times B)^\circ :\equiv \langle a^\circ, b^\circ\rangle : A^\circ \times B^\circ$
- $(\mathsf{pr}_1(a, b) : A)^\circ :\equiv \mathsf{pr}_1(\langle a^\circ, b^\circ\rangle) : A^\circ$
- $(\mathsf{pr}_2(a, b) : B)^\circ :\equiv \mathsf{pr}_2(\langle a^\circ, b^\circ\rangle) : B^\circ$
- $([x : A \mid p])^\circ :\equiv \lambda x : A^\circ.p^\circ : A^\circ \to \mathsf{Prop}_{\mathcal{U}_n}$

For the sake of brevity, the proof-terms witnessing h-propositions above have been written without spelling out their arguments: e.g., $p_\times$ stands for $p_\times(\mathsf{pr}_1(p^\circ), \mathsf{pr}_1(q^\circ), \mathsf{pr}_2(p^\circ), \mathsf{pr}_2(q^\circ))$ and $p_\vee$ for $p_\vee(\mathsf{pr}_1(p^\circ), \mathsf{pr}_1(q^\circ))$. These proof-terms are provided by the lemmas stated in 4.3.1. For proof-terms of h-sets instead we have adopted a notation where the arguments appear as subscripts. The derivation of these proof-terms also relies on 4.3.1.

Theorem 5.3.9. *The interpretation* $(-)^\circ$ *of* $\mathfrak{LM}^i$ *in* **HoTT** *is valid.*

Proof. The proof is by structural induction. The logical rules are trivially validated, so it remains to prove the validity of the extra-logical axioms. The axioms for product types are interpreted as follows:

$$(\mathsf{pr}_1((x,y)) =_A x)^\circ :\equiv (\mathrm{Id}^\circ_A(\mathsf{pr}_1(\langle x,y\rangle), x), p_{\mathrm{Id}})$$

and then we have to show that the first component is inhabited. But this trivially follows. The axiom $(\mathsf{pr}_2((x,y)) =_B y)$ is similarly validated. Then, there remains the axiom $(\mathsf{pr}_1(z), \mathsf{pr}_2(z)) =_{A\times B} z$. The first component of the interpretation of this axiom amounts to

$$\mathrm{Id}_{A^\circ\times B^\circ}(\langle \mathsf{pr}_1(z), \mathsf{pr}_2(z)\rangle, z)$$

and we know from Thm. 2.6.2 in [155] that this is equivalent to

$$\mathrm{Id}_{A^\circ}(\mathsf{pr}_1(\langle \mathsf{pr}_1(z), \mathsf{pr}_2(z)\rangle), \mathsf{pr}_1(z)) \times \mathrm{Id}_{B^\circ}(\mathsf{pr}_2(\langle \mathsf{pr}_1(z), \mathsf{pr}_2(z)\rangle), \mathsf{pr}_2(z))$$

and hence the result immediately follows.
Also, the verification of the axioms for natural numbers is straightforward.
In the case of set extensionality, the first component of the interpretation amounts to

$$X =_{A^\circ\to \mathsf{Prop}_{\mathcal{U}_n}} Y \leftrightarrow (\Pi_{(z:A^\circ)} X(z) \leftrightarrow Y(z))$$

Then, by function extensionality

$$X =_{A^\circ\to \mathsf{Prop}_{\mathcal{U}_n}} Y \leftrightarrow \Pi_{(z:A^\circ)} X(z) =_{\mathsf{Prop}_{\mathcal{U}_n}} Y(z)$$

that, in turn, by propositional extensionality is equivalent to

$$\Pi_{(z:A^\circ)} X(z) \leftrightarrow Y(z)$$

and thus the axiom is validated.
The axiom of propositional extensionality for Ω_n is trivially validated by the fact that propositional extensionality holds in the model.
Also, the cumulativity conditions over Ω and $\mathsf{Set}(A)$ immediately holds in the model.
Finally, we have to check the comprehension axiom. The first component of the interpretation is

$$\lambda x : A^\circ(\mathsf{pr}_1(p^\circ(x)(z))) \leftrightarrow \mathsf{pr}_1(p^\circ[z/x])$$

that is clearly valid in the model. □

The extension of **HoTT** h-propositions with the excluded middle

$$\Pi_{P:\mathcal{U}_i}(\mathsf{isProp}(P) \to P + \neg P)$$

allows us to interpret also the classical system $\mathfrak{LM}$:

Proposition 5.3.10. *The interpretation* $(-)^\circ$ *of* $\mathfrak{LM}$ *in* **HoTT** *is valid.*

Proof. The proof is completely similar to that of 5.3.9. We need to check only the exlcuded middle, but this is validated by the definition of the interpretation and the assumption of excluded middle for h-propositions. □

The relation with $\mathfrak{W}$. It is natural to ask what is the relationship between the system $\mathfrak{LM}$ and the system $\mathfrak{W}$ proposed by Casari. The type structure of the latter is easily seen to be interpretable in $\mathfrak{LM}$:

- $[\![0^0]\!] :\equiv \mathbf{N}$
- $[\![(\tau_1^{m_1}, \ldots, \tau_n^{m_n})^m]\!] :\equiv \mathsf{Set}_m([\![\tau_1^{m_1}]\!] \times \ldots \times [\![\tau_n^{m_n}]\!])$

Namely, $[\![()^m]\!] :\equiv \Omega_m$. The interpretation extends to terms and formulas in a straigthforward way. The axiom of comprehension is immediately seen to be valid, while the interpretation of axiom of extensionality of $\mathfrak{W}$ is validated by extensionality and comprehension in the target theory.
However, I do not maintain Casari's distinction between 'predicative' symbols and set symbols, because this makes the formalization smoother.

The restricted process. The 'narrow procedure' described by Weyl consists in restricting quantification only to natural numbers and hence blocking the iteration of the Logico-Mathematical process to higher levels. The resulting version of $\mathfrak{LM}$, that is called $\mathfrak{LM}_0$, has some interest. In particular, I will show that the theory $\mathbf{ACA}_0$ is interpretable in it.
There exists a wide consensus in the literature on *Das Kontinuum* about the fact that predicative analysis as developed by Weyl can be formalised in a system with the same strength as $\mathbf{ACA}_0$. According to Casari's interpretation discussed above, the development of predicative analysis takes place within the system resulting from the narrow procedure. Therefore, it would be desirable to show that $\mathfrak{LM}_0$ has the same strength as $\mathbf{ACA}_0$. I will show that there exists a translation from $\mathbf{ACA}_0$ to $\mathfrak{LM}_0$, but it is likely that a translation going into the other direction can be defined as well. This would

establish the claim about the equivalent strength of the two theories. A necessary preliminary remark is that the restriction over quantification that characterises the restricted process must not be understood as a constraint on the formation rules of the language. We can still form a formula such $\forall x : A.\phi$, where A is a type with level higher than $\mathbb{N}$ (e.g. $\mathsf{Set}_0(\mathbb{N})$). The constraint is that such a formula does not yield any set, i.e. the mathematical process cannot be applied to it. Rather, the only formulas that yield sets are the arithmetical ones. Therefore, the restriction is meant to apply to the comprehension principle. The type structure of $\mathfrak{LM}_0$ is defined as follows:

Definition 5.3.11. The types of $\mathfrak{LM}_0$ are the following:

$$\mathbb{N} \qquad \mathsf{Set}_0(\mathbb{N}) \qquad \Omega_0$$

In the following, I will omit the index 0 by putting $\mathsf{Set}(\mathbb{N}) :\equiv \mathsf{Set}_0(\mathbb{N})$. Further, I assume primitive symbols for the sum and the multiplication of natural numbers.

Definition 5.3.12. The terms of $\mathfrak{LM}_0$ are the following:

- For each type A, variables of A are terms.
- $0 : \mathbb{N}$ and if $a : \mathbb{N}$, then $\mathsf{succ}(a) : \mathbb{N}$. If $a, b : \mathbb{N}$, then $a + b, a \cdot b : \mathbb{N}$.
- If $a, b : \mathbb{N}$, then $a =_{\mathbb{N}} b : \Omega_0$.
- If $p, q : \Omega_0$, then $p \square q : \Omega_0$, where $\square \in \{\wedge, \vee, \rightarrow\}$.
- If $p : \Omega_0$, then $\neg p : \Omega_0$.
- If $x : \mathbb{N}$ and $p : \Omega_0$, then $\mathcal{Q}x : \mathbb{N}.p : \Omega_0$, where $\mathcal{Q} \in \{\exists, \forall\}$.
- If $x : \mathbb{N}$ and $p : \Omega_0$, then $\{x : \mathbb{N} \mid p\} : \mathsf{Set}(\mathbb{N})$.
- If $a : \mathbb{N}$ and $X : \mathsf{Set}(\mathbb{N})$, then $a \; \varepsilon \; X : \Omega_0$.

The type Ω_0 corresponds to the type of arithmetical formulas. In order to interpret arbitrary formulas in the language of $\mathbf{ACA}_0$, we introduce a notion of *large* formulas, which include those formulas not contained in Ω_0:

Definition 5.3.13. The large formulas of $\mathfrak{LM}_0$ are defined as follows:

- Every arithmetical formula is a large formula, that is if $p : \Omega_0$, then $\tilde{p}$ is a large formula.

- Large formulas are closed under the usual connectives.

- If $x, y : A$, then $x =_A y$ is a large formula.

- If $x : A$ and P is a large formula, then $\mathcal{Q}x : A.P$ is a large formula.

The axioms of $\mathfrak{LM}_0$ comprehend an axiomatisation of many-sorted classical logic and the usual Peano axioms for $0, \mathsf{succ}, +$ and $\cdot$. Then, there is the induction axiom, for $X : \mathsf{Set}(\mathbb{N})$,

$$(0 \; \varepsilon \; \mathbb{N} \wedge \forall n : \mathbb{N} \; (n \; \varepsilon \; X \rightarrow \mathsf{succ}(n) \; \varepsilon \; X)) \rightarrow \forall n : \mathbb{N} \; (n \; \varepsilon \; X)$$

and the extensional principles for formulas of Ω_0 and set-like terms of $\mathsf{Set}(\mathbb{N})$. The comprehension principle is restricted to arithmetical formulas and thus to terms of $\mathsf{Set}(\mathbb{N})$.
For the presentation of $\mathbf{ACA}_0$, I follow the standard reference [144], except for the fact that I do not take the order relation as primitive symbol. The language of second order arithmetic Z_2 comprehends variables of two sorts: number variables $i, k, m, n \ldots$ and set variables $X, Y, Z \ldots$ The numerical terms are specified as follows: every number variable is a term; the constants 0 and 1 are terms and if s, t are terms, then $s + t$ and $s \cdot t$ are terms. The atomic formulas consist of equalities between terms $s = t$ and of those formulas having the form $t \in X$, where X is a set variable. Formulas are built up form the atomic formulas by means of the connectives $\wedge, \vee, \neg, \rightarrow, \leftrightarrow$ and the quantifiers $\forall n, \exists n$ ranging over numerical terms and the quantifiers $\forall X, \exists X$ ranging over set variables. The axioms of $\mathbf{ACA}_0$ are the universal closures of the following formulas in the language just described:

- Basic Axioms:

$$\begin{array}{ll} n + 1 \neq 0 & m + 1 = n + 1 \rightarrow m = n \\ m + 0 = m & m + (n + 1) = (m + n) + 1 \\ m \cdot 0 = 0 & m \cdot (n + 1) = (m \cdot n) + m \end{array}$$

- Induction Axiom:

$$(0 \in X \wedge \forall n(n \in X \rightarrow n + 1 \in X)) \rightarrow \forall n(n \in X)$$

- Arithmetical Comprehension Axiom:

$$\exists X \forall n (n \in X \leftrightarrow \phi(n))$$

where $\phi(n)$ is arithmetical and X does not occur free in ϕ.

It is straightforward to define a translation from the language of $\mathbf{ACA}_0$ to the language of $\mathfrak{LM}_0$ which sends numerical terms to terms of type $\mathbb{N}$, arithmetical formulas to terms of type Ω_0 and second-order quantifiers to large formulas with $\mathsf{Set}_0(\mathbb{N})$ as domain of quantification.

Definition 5.3.14 (Translation from $\mathbf{ACA}_0$ to $\mathfrak{LM}_0$)**.** The mapping $(-)^\tau$ from $\mathbf{ACA}_0$-expressions to $\mathfrak{LM}_0$-expressions is defined as follows:

$$\begin{aligned} (x)^\tau &:\equiv x : \mathbb{N} \\ (0)^\tau &:\equiv 0 : \mathbb{N} & (t+1)^\tau &:\equiv \mathsf{succ}(t^\tau) : \mathbb{N} \\ (t+s)^\tau &:\equiv t^\tau + s^\tau : \mathbb{N} & (t \cdot s)^\tau &:\equiv t^\tau \cdot s^\tau : \mathbb{N} \end{aligned}$$

In the case of arithmetical formulas, we define

$$\begin{aligned} (t = s)^\tau &:\equiv t^\tau =_{\mathbb{N}} s^\tau : \Omega_0 \\ (t \in X)^\tau &:\equiv t^\tau \ \varepsilon \ X^\tau : \Omega_0 \\ (\neg p)^\tau &:\equiv \neg p^\tau : \Omega_0 \\ (p \square q)^\tau &:\equiv p^\tau \square q^\tau : \Omega_0 \\ (\mathcal{Q}x.p)^\tau &:\equiv \mathcal{Q}x : \mathbb{N}.p^\tau : \Omega_0 \end{aligned}$$

The translation naturally extends to arbitrary formulas, which are interpreted as large formulas. Then we give only the following clauses:

$$\begin{aligned} (\forall X.P)^\tau &:\equiv \forall X : \mathsf{Set}(\mathbb{N}).P^\tau \\ (\exists X.P)^\tau &:\equiv \exists X : \mathsf{Set}(\mathbb{N}).P^\tau \end{aligned}$$

Finally, we can establish the soundness of the translation:

Theorem 5.3.15. *If* $\boldsymbol{ACA}_0 \vdash \phi$*, then* $\mathfrak{LM}_0 \vdash (\phi)^\tau$*.*

Proof. The proof is a straightforward induction over the derivation of ϕ. $\square$

Palmgren's IRTT. The system $\mathfrak{LM}^i$ bears a clear relationship to the internal language of elementary toposes. First, we may extend our type structure with the unit type $\mathbf{1}$, where $\ell(\mathbf{1}) = 0$. Namely, the unit type contains a canonical element $\star$ and the relevant axiom is

$$x : \mathbf{1} \vdash x =_{\mathbf{1}} \star$$

Then, let us consider the intuitionistic simple theory of types introduced in [82] , which includes the following types $\mathbf{1}, \Omega, N$ that are closed under $\times$ and $P(-)$. Such a theory provides a syntactic presentation of the internal theory of an elementary topos. We can define a map from $\mathfrak{LM}^i$ into this theory that 'erases' all the level annotations, thus collapsing the predicative hierarchy. The result is exactly the intuitionistic simple theory of types by Lambek and Scott. In the light of the work of [14], the system $\mathfrak{LM}^i$ can be regarded as an example of ramified local set theory. In this respect, the work by Palmgren [117] is especially relevant. There a system **IRTT** is introduced to which the system $\mathfrak{LM}^i$ is quite close, modulo some modifications. For instance, Palmgren does not include a primitive sequence of types Ω_n and defines formulas and terms by mutual inductions. It must be remarked that, in the presence of the unit type, these types of propositions can be dispensed with also in $\mathfrak{LM}^i$. In fact, propositions can be represented as elements of $\mathsf{Set}_k(\mathbf{1})$ in a way similar to what happens in an elementary topos. A major part of Palmgren's paper is devoted to the study of the category of *ramified local sets*: the objects are triple (A, X, n) where A is a type, $X : \mathsf{Set}_n(A)$ and a morphism between (A, X, n) and (B, Y, m) is a functional relation $F : \mathsf{Set}_k(A \times B)$. The equality of morphisms F, G is defined as

$$(\forall x : A)(\forall y : B)((x, y) \; \varepsilon \; F \leftrightarrow (x, y) \; \varepsilon \; G)$$

and identity and composition are defined as expected.
For many constructions considered by Palmgren, it appears to be necessary to consider not only the level of a type A, but also the level of the associated equality that can be recursively defined as follows:

- $\ell_{eq}(\mathbf{1}) = 0$
- $\ell_{eq}(\mathbb{N}) = 0$

- $\ell_{eq}(\Omega_n) = n$
- $\ell_{eq}(A \times B) = \mathsf{max}(\ell_{eq}(A), \ell_{eq}(B))$
- $\ell_{eq}(\mathsf{Set}_n(A)) = \mathsf{max}(\ell_{eq}(A), n)$

Then it immediately follows that for any type A, $\ell_{eq}(A) \leqslant \ell(A)$.
A striking feature of Palmgren's theory is that it includes a restricted version of the Reducibility axiom which is called *Functional Reducibility* and is formulated as follows: for any level r

$$(\forall X : \mathsf{Set}_m(A))(\forall Y : \mathsf{Set}_n(B))(\forall F : \mathsf{Set}_r(A \times B))$$

$$[F : (A, X, m) \mapsto (B, Y, n) \Rightarrow (\exists G : \mathsf{Set}_k(A \times B)(\forall z : A \times B)(z \,\varepsilon\, F \leftrightarrow z \,\varepsilon\, G)]$$

where $k = \mathsf{max}(\ell_{eq}(B), m, n)$ and $F : (A, X, m) \mapsto (B, Y, n)$ stands for F being a functional relation from (A, X, m) to (B, Y, n). Therefore, the axiom asserts that for any functional relation F of level r, there exists a functional relation G of level k, which is bounded by the level of the sets and the level of equality over B, that is extensionally equivalent to it. Further, such a map is unique by extensionality. What the axiom achieves is the closure of levels under exponentiation, since it bounds the level of a map on the basis of the levels of its domain and codomain. For instance, if we consider the local set $\mathsf{Nat} = (\mathbb{N}, \{x : \mathbb{N} \mid T\})$, where $\{x : \mathbb{N} \mid T\} : \mathsf{Set}_0(\mathbb{N})$ and $F : \mathsf{Set}_k(\mathbb{N} \times \mathbb{N})$ such that F is a map from Nat to Nat, then the k will be necessarily greater than 0. However, by applying the functional reducibility axiom, we get an extensionally equivalent map G with level 0. In general, for any fixed level m, it is possible to show that set of maps between two objects of this level is not closed under exponentiation. Functional reducibility provides a way to restore this closure property.
Moreover, Palmgren shows that the system with the functional reducibility axiom enjoys an interpretation in a setoid model built over a version of Martin-Löf type theory, whose ordinal is Γ_0. Therefore, the axiom is predicatively acceptable. I conjecture that the axiom should be valid also in the **HoTT**-model.

Remark 5.3.16. The main difference between the two model constructions is that the interpretation in **HoTT** preserves what can be thought of as an adequacy criterion, that is the distinction between propositions and sets is preserved. Indeed, this seems an essential point underlying the ramified

theory. In this sense, I think that the h-prop interpretation is more adequate on a conceptual level than the Curry-Howard interpretation of logic if we take into account the way propositions are considered in the ramified system.

As we will get to see in the next section, it should be expected that the closure under exponentiation achieved by the means of the functional reducibility axiom is not stable under the change of logic from a predicative standpoint, that is, if we take the classical extension of the system, then the axiom becomes fully impredicative in analogy to what happens in other constructive systems that contain exponentiation of functional relations. Indeed, in the last section of [117], it is proved that, under the assumption of the excluded middle, the Full Reducibility Axiom[20] is derived. Therefore, the classical system $\mathfrak{LM}$ cannot be extended with this kind of axiom.
A major motivation behind the introduction of **IRTT** is that it should give rise to a notion of predicative ramified topos. Nonetheless, such structure is not axiomatised in [117] nor, as far as I know, elsewhere. Then, the problem of providing a suitable axiomatic description of ramified toposes is still open.

5.4 Constructive Predicativity

One suggestion stemming from the previous sections is that the constructive conception of predicativity might turn out to be wider than its classical counterpart. There are axioms and principles that are predicatively justified only within a constructive framework. Thus, the notion of predicativity might be sensitive to the underlying logic.
Both classical and constructive predicativists agree on the rejection of the *power set* axiom as it appears to be strongly impredicative compared to other axioms. This axiom states that, given a set x, we can form the set $\mathcal{P}(x)$ of all the subsets of x. Even in the presence of *bounded* separation, i.e. the restricted version of separation where the formula ϕ contains only bounded quantifiers, it enables us to quantify over subsets. For instance, we could impredicatively specify a set a as the set of all x in b such that for any subset z of b a certain condition $\phi(x, z)$ holds. Then, the circular character of the definition is immediately visible, because the definiendum a itself is among the subsets z.

[20] $(\forall X : \mathsf{Set}_k(A))(\exists Y : \mathsf{Set}_0(A))(\forall x : A)(x \in X \leftrightarrow x \in Y)$.

A standard objection to the power set axiom on a predicative ground is that, granted the existence of $\mathbb{N}$, the axiom allows to form the sequence

$$\mathcal{P}(\mathbb{N}), \mathcal{P}(\mathcal{P}(\mathbb{N})), \ldots$$

and these infinite sets turn out to be particularly problematic. A first challenge is to make sense of statements involving such sets: they are examples of collections that are not invariant in Poincaré's sense and hence classical quantification over them is not well-determined[21]. In general, it is the application of the power set axiom to infinite sets that is questionable: for cardinality reasons prevent us from having an effective way of generating all the subsets of an infinite set and thus we do not have a definite conception of what is an arbitrary subset of an infinite set. The axiom simply allows the formation of the set of all these subsets, but it does not prescribe a *rule of construction* that we can follow. Moreover, it destroys the idea that sets are constructed from other sets already available at previous stages: when we gather all the subsets of a given set, it does not follow that they have already been recognized to exists at some previous stage.
The last remarks suggest that from a constructive and predicative point of view, the power set can be rejected simply because it does not correspond to any principle of construction or rule. In particular, the objection does not need to make appeal to cardinality issues. In my opinion, this is the idea underlying Myhill's objection to the power set axiom in the context of the constructive and predicative set theory **CST**:

> "Power set seems especially nonconstructive and impredicative compared with the other axioms: it does not involve, as the others do, putting together or taking apart sets that one has already constructed but rather selecting out of the totality of all sets those that stand in the relation of inclusion to a given set." ([110], p.351).

The point is reinforced by the result given by Myhill in the same paper that, under the assumption of Church's Thesis, even the power set of a singleton is not countable. Myhill's remark is that power set is already impredicative in the case of finite sets and thus cardinality arguments are not the main reasons

[21] On these grounds, Feferman has challenged the very idea that Continuum Hypothesis is a well-defined mathematical problem [51].

to reject this axiom. The result depends on the assumption of Church's Thesis and I cannot survey here the arguments that have been given in the literature on constructive mathematics in support for or against this principle. Nevertheless, it is interesting to observe that power set also implies that not every object can be described in a finite way, a principle that has been assumed in some constructivist schools.

For our purposes, the main point emerging from the previous discussion is that a crucial difference intervenes between collections that are generated by a rule and those that are not. In this respect, a basic insight of Weyl is vindicated.

Predicatively, the name 'sets' should be reserved for the first, while the latter can be called 'classes' or 'collections'. Then, crucially we distinguish between collections with an effective content and those without it. To a first approximation, the first are those which are inductively generated and hence their construction process can be understood as occurring from below. On the other hand, the distinction is similar to that between sets and categories proposed in [99]. There, a category (that must not be confused with the notion of category from category theory) is a totality for which we can grasp what an object of it is, even if it is not possible to fix rules that generate all the elements belonging to it. More precisely, we can recognize an object as an object of a certain category when such a object is given, but there is no exhaustive description of the way these objects are generated. It turns out that the power set of $\mathbb{N}$ is a natural example of category. In this sense, a category is inherently open-ended.

Also, the practice of constructive (predicative) mathematics in the last decades lends support to the idea of the relevance of this distinction. It is a fact that in predicative setting the power set of a non-empty set x, and hence the discrete topology on x, fails to be a set. However, it is an example of class in our sense. Namely, the classical definition of topological space is itself impredicative, since it involves quantifiers ranging over subsets and families of subsets. However, the approach known as *formal topology* allows us to deal with topological spaces in a purely constructive and predicative way [136].

The conclusion is that predicatively it must be rejected that any set has a power set, but it must be maintained that any set has a power class or collection. Therefore, it is useful to introduce explicitly this distinction in the language of the theory. The theory **MF** from Chapter 4 supports this distinction and hence it is faithful to the practice of constructive mathematics. The distinction between propositions and small propositions is just a byproduct

of the main distinction between sets and collections.
One aspect that makes constructive predicativity wider than classical predicativity is that in the case of the first we can safely assume the closure of functions under exponentiation: given two sets A, B, the collection B^A of functions from A to B is a set. Classically, this principle is equivalent to the power set axiom: it is enough to consider the set $\{0,1\}^A$ for any A and the subsets as characteristic functions. Constructively, instead, the axiom is weaker than the power set. The axiom was first considered by Myhill in [110] with the following motivation:

> "One could make the same, admittedly vague, objection to the existence of the set $A \to B$ of mappings of A into B but I do not think the situation is parallel − a mapping or function is a rule, a finite object which can actually be given" ([110], p.351)

The objection Myhill is referring to is the one about the non-constructive and impredicative nature of the power set. However, Myhill's justification does not stand up to scrutiny if understood in a too literal way and the subsequent development of his paper is intended to make, among other things, a more solid case for this justification.
Myhill's words echo Bishop's definition of *operation*. According to Bishop, an operation $f : A \to B$ is given by a rule that can be effectively executed into a finite number of steps. Such an f is an *extensional function* if $a_1 =_A a_2 \to f(a_1) =_B f(a_2)$. Since Myhill is working within a set-theoretic context, he defines a function as an extensional functional relation. Then, the computational content of the notion is lost. Nevertheless, Myhill appeals to the effective notion of operation in the introduction of the exponentiation axiom. The main problem, then, is to restore this effective content in order to make the argument supporting exponentiation go through. Myhill's solution is to provide a realizability interpretation of his system. Then, at the metalinguistic level, the functions of the system are seen to be effective and thus their algorithmic content is recovered. However, I want to make a case for preserving the distinction between (extensional) functions and operations within a given framework, working out some ideas already presented in Chapter 4.
In type-theoretic frameworks based on Martin-Löf type theory, the notion of function is taken as primitive. These type-theoretic functions can be legitimately regarded as operations in Bishop's sense. Then, the rules for

Π-types represent a generalization of the exponentiation axiom, since the universe is assumed to be closed under dependent products. In particular, they allow for the introduction of higher-order function types.
We know from the previous chapters that the axiom of unique choice AC! is a theorem in type theory, since it follows from the general AC. Further, we also know that even if AC is not derivable in **HoTT** when properly reformulated with truncated Σ, AC! is still a theorem in that setting. The same is true in elementary toposes. Recall that AC! can be expressed as follows:

$$\text{(AC!)}\ \forall_{(x:A)}\exists!_{(y:B)}\ F(x,y) \to \exists_{(f:A\to B)}\forall_{(x:A)}\ F(x,f(x))$$

where the antecedent is an abbreviation for the fact that the relation $F(x,y)[x : A, y : B]$ is *total*

$$\forall_{(x:A)}\exists_{(y:B)}F(x,y)$$

and *single-valued*

$$\forall_{(x:A)}\forall_{(y,y':B)}F(x,y) \wedge F(x,y') \to y =_B y'$$

Therefore, the validity of AC! amounts to asserting that any functional relation corresponds to a type-theoretic function and thus the two notions are identified. As a result, the closure under exponentiation of functional relations is automatically valid.
For the result mentioned above, if Martin-Löf type theory is extended with excluded middle, then the theory is fully impredicative and there cannot be a corresponding classical predicative theory. The outcome is that taking together AC! and the exponentiation axiom is predicatively justified only within constructive contexts and hence there a correspondence with classical predicative theories cannot be established.
A possible way-out is to drop exponentiation altogether, but then we are committed to show that the resulting systems retain some expressive strength and allow the formalization of significant parts of mathematics. Another solution is to block the internal validity of AC! and is what is pursued in a system such as **MF**. In **MF** the way the rules for the existential quantifier are given is what prevents the derivation of AC!: the crucial aspect is that propositions eliminate only in propositions and not in sets or collections. Then, quite interestingly, for systems such as **MF** it should be possible to show that they are compatible with classical predicative theories such as those introduced by Feferman and to maintain exponentiation for type-theoretic

functions. The argument given by Myhill for exponentiation naturally applies to this setting. The connection with classical predicativity turns out to be one of the benefits deriving from the preservation of the distinction between operations and functional relations within a given system.
Another sense in which constructive predicativity is held to be stronger than classical predicativity concerns the fact that *inductive* and even *generalized* inductive definitions are regarded to be predicatively admissible on this conception. This issue will be the focus of the present section. In this respect, the notion of *generalized* predicativity has been introduced to characterize those systems which include different forms of inductive definitions that are held to be predicatively justified. These systems usually are proof-theoretically stronger than predicative theories in the sense of Feferman-Schütte, since their proof-theoretic ordinals are greater than Γ_0. In Martin-Löf Type Theory, these strong forms of inductive definitions are assumed to be justified on the basis of the *meaning explanations* elaborated by Martin-Löf in [98]. In fact, they are meant to make more precise the sense in which these inductively defined objects exhibit a *well-founded* structure and thus can be considered as if they were *built up from below*. As far as I know, the connection between meaning explanations and generalised predicativity is only hinted at in the literature, but there is no thorough analysis of it.
Predicativity is often taken to be, more or less explicitly, a *relative* rather than an absolute notion. This means that articulating a conception of predicativity presupposes that we spell out not only a negative constraint such as the vicious circle principle, but also which notions must be taken for granted. It is precisely w.r.t. such notions that predicative definability is characterised. The obvious example is the notion of predicativity given the natural numbers, where the natural number structure is taken for granted. However, in light of the previous considerations, the logic that is assumed must also be specified. The case of the exponentiation axiom provides a clear example of the fact that the predicative status of certain objects might depend on the underlying logic. Then, we know that, for what concerns predicativity given the natural numbers, classical logic is usually assumed to hold over well-determined domains. On the other hand, generalized predicativity seems to be grounded on a constructive reading of logical constants and thus the background logic is intuitionistic. Moreover, in the constructive conception of predicativity, the paradigmatic role of natural numbers is extended to a general notion of inductive structure, which, under some assumptions, can be expressed in terms of well-founded trees. It results that generalised predicativism could

be understood as providing a notion of predicative definability relative to intuitionistic logic and general inductively generated sets.

5.4.1 Inductive Definitions and $\mathcal{W}$-types

As we have seen in section 1.2.1, inductive definitions of different kinds and complexity can be introduced in type theory. In a classical context, even an ordinary inductive definition looks circular when analysed set-theoretically. However, as I have already stressed in the opening of this chapter, inductive definitions have a strong constructive flavour since they can be regarded as ways of specifying rules of construction for certain objects. Moreover, we are not necessarily committed to a set-theoretical explanation of them. Then, there could be more direct explanations that make sense of their constructive and predicative appeal.

The most typical example of an ordinary inductive definition is given by the type of natural numbers $\mathbb{N}$. Other examples are the type of lists List and the type of binary trees $\mathsf{bintree}$. I recall that the Kreisel-Feferman-Schütte analysis is meant to capture a *relative* notion of predicativity, namely predicativity given the natural numbers. Then, the inductive definition of $\mathbb{N}$ is taken for granted and is not considered to be problematic. However, there are stricter notions of predicativity that challenge even the predicative status of the natural numbers structure and, in particular, of the induction principle over $\mathbb{N}$ [112]. I will not be concerned here with this strict notion of predicativity.

A first example of *generalized* inductive definition is provided by the type of constructive ordinals $\mathcal{O}$, which consists of the following rules:

$$\frac{}{\Gamma \vdash \mathcal{O} : \mathcal{U}}\ \mathcal{O}\text{-F}$$

$$\frac{}{\Gamma \vdash 0 : \mathcal{O}}\ \mathcal{O}\text{-I}_1 \qquad \frac{\Gamma \vdash a : \mathcal{O}}{\Gamma \vdash \mathsf{s}(a) : \mathcal{O}}\ \mathcal{O}\text{-I}_2 \qquad \frac{\Gamma \vdash f : \mathbb{N} \to \mathcal{O}}{\Gamma \vdash \mathsf{sup}(f) : \mathcal{O}}\ \mathcal{O}\text{-I}_3$$

$$\frac{\begin{array}{c}\Gamma, z : \mathcal{O} \vdash C(z) : \mathcal{U} \\ \Gamma \vdash c : \mathcal{O} \\ \Gamma \vdash d : C(0) \\ \Gamma, x : \mathcal{O}, y : C(x) \vdash e(x, y) : C(\mathsf{s}(x)) \\ \Gamma, h : \mathbb{N} \to \mathcal{O}, g : \Pi_{(n:\mathbb{N})} C(h(n)) \vdash f(h, g) : C(\mathsf{sup}(h))\end{array}}{\Gamma \vdash \mathsf{Ind}_{\mathcal{O}}(c, d, x.y.e, h.g.f) : C(c)}\ \mathcal{O}\text{-E}$$

The rule $\mathcal{O}$-E corresponds to transfinite induction, because it can be reframed as follows: given a predicate C defined over the constructive ordinals, let us assume that C holds for the initial ordinal, if $C(a)$ then C holds for the successor of a and for an enumeration of ordinals given by a map f, if C holds for any ordinal in the sequence, then C holds for the sup of the sequence. Then, we conclude that C holds for any ordinal in $\mathcal{O}$.
Systems including definitions such as the one displayed above usually do not count as predicative according to the Feferman-Schütte criterion. In general, the proof-theoretic strength of theories of inductive definitions is well beyond Γ_0: ID_1, the theory of a single iterated inductive definition, has already a proof-theoretic ordinal greater than Γ_0 and is the *Bachmann-Howard* ordinal $\psi_\Omega(\varepsilon_{\Omega+1})$. The same ordinal classifies CZF and Martin-Löf Type Theory with a single universe and the type $\mathbb{V}$ of iterative sets. The version of type theory including the type of *well-founded trees* $\mathcal{W}$ is even stronger proof-theoretically. $\mathcal{W}$-types represent a powerful example of generalized inductive definitions, since they can be understood in a precise sense as a generalization of all tree-like types. In fact, Peter Dybjer [44] has showed that, at least in extensional type theory[22], all strictly positive inductively defined types can be reduced to $\mathcal{W}$. Therefore, types such as $\mathbb{N}$ and $\mathcal{O}$ can be regarded as special instances of $\mathcal{W}$-types. The rules for the type of well-founded trees are the following:

$$\frac{\Gamma \vdash A : \mathcal{U} \qquad \Gamma, x : A \vdash B(x) : \mathcal{U}}{\Gamma \vdash \mathcal{W}_{(x:A)} B(x) : \mathcal{U}}\ \mathcal{W}\text{-F}$$

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash f : B[a/x] \to \mathcal{W}_{(x:A)} B(x)}{\Gamma \vdash \mathsf{sup}(a, f) : \mathcal{W}_{(x:A)} B(x)}\ \mathcal{W}\text{-I}$$

[22]However, see also the recent work by Hugunin [69] in the intensional setting.

$$\frac{\begin{array}{c}\Gamma, w : \mathcal{W}_{(x:A)}B(x) \vdash C(w) : \mathcal{U} \\ \Gamma \vdash c : \mathcal{W}_{(x:A)}B(x) \\ \Gamma, x : A, y : B(x) \to \mathcal{W}_{(x:A)}B(x), z : \Pi_{(v:B)}C(y(v)) \vdash d(x, y, z) : C(\mathsf{sup}(x, y))\end{array}}{\Gamma \vdash \mathsf{ind}_{\mathcal{W}}(c, x.y.z.d) : C(c)}\ \mathcal{W}\text{-E}$$

The computation rule $\mathcal{W}$-C is the following:

$$\mathsf{ind}_{\mathcal{W}}(\mathsf{sup}(a, f), x.y.z.d) \equiv d(a, f, \lambda v.\mathsf{ind}_{\mathcal{W}}(f(v), x.y.z.d)) : C(\mathsf{sup}(a, f))$$

An intuitive explanation of these rules can be given along the following lines: an element of $\mathcal{W}$ is a well-founded tree with the nodes represented by elements in A and the immediate predecessors of a given node indexed by $B(x)[x : A]$. Therefore, at each $x_0, x_1, x_2, \ldots$ in $B(a)[a : A]$ corresponds a well-founded tree indexed by one of these elements. Then the application of the introduction rule amounts to the construction of a new tree (with a new root) that collects all these well-founded trees indexed by elements in $B(a)$. The elimination rule amounts to a principle of transfinite induction over well-founded trees.
Another possible explanation of $\mathcal{W}$-types, which makes clear why they can be viewed as a generalization of inductive types, consists in taking A as the type of labels standing for constructors and $B(x)[x : A]$ as the type of the arities associated to constructors. For instance, we know that the type of natural numbers has two constructors that are zero and the successor function. The arity of the former is zero, while that of the latter is one. Then we can define the type of the natural numbers as $\mathcal{W}_{(x:\mathbf{2})}B(x)$, where $B(0_{\mathbf{2}}) = \mathbf{0}$ and $B(1_{\mathbf{2}}) = \mathbf{1}$.
Semantically, well-founded trees correspond to *initial algebras* for a special class of endofunctors [107].

Remark 5.4.1. There is another possible general representation of inductive types in type theory, which has been thoroughly investigated by P. Dybjer [42], and consisting in the formulation of a general *scheme*. In this case, it can be shown that the usual type constructors are particular instances of this scheme. This approach is a development of the line of research initiated in [95].

Finally, we know that type theory allows for even more complex kinds of inductive definitions, such as induction-induction and induction-recursion.

In particular, more powerful universe constructions can be introduced and has been argued that they are predicatively valid [142]. For our purposes, it would be sufficient to consider the case of generalized inductive definitions, without delving into these more complex cases.

5.4.2 On the Method of Meaning Explanations

The method of *meaning explanations* provides an intuitive, pre-mathematical semantics for the judgements of type theory. Roughly speaking, it is a kind of *operational* semantics that gives a direct insight about the validity of judgements. Meaning explanations were introduced by Martin-Löf in [98] and then fully worked out in [99, 100].

This informal semantics made its first appearance concurrently with the *extensional* version of type theory [98]. Indeed, the reflection rule for identity types is validated by meaning explanations. On the other hand, it must be remarked that Dummett's thoughts about a theory of meaning in terms of *verifications* and also his analysis of the notion of harmony and his distinction between demonstrations and canonical proofs have been a major influence on Martin-Löf's meaning explanations. Also, the parallel work by Prawitz on the notion of *proof-theoretic* validity has been another influence.

Meaning explanations as worked out by Martin-Löf make the traditional intuitionistic explanation of logical constants more precise and, at the same time, represent an extension of it. In fact, they provide a semantics for general mathematical constructions and are not limited to the case of logical constants. In this sense, they also have a wider application than traditional *proof-theoretic semantics* as developed by Dummett and Prawitz.

Operational Semantics. Meaning explanations provide an operational semantics in the sense that they are based on a primitive notion of computation. In the following, I will try to make more precise this remark presenting the semantics in a way that is inspired by the standard reference [113]. The general scheme underlying Martin-Löf's meaning explanations is the following: we start from a set $\mathcal{E}$ of raw expressions over a given signature σ. Then, they are subdivided into *open* and *closed* expressions. An expression $\alpha \in \mathcal{E}$ is said to be closed if it does not contain free variables, otherwise, it is open. Open expressions are *reduced* to closed expressions by substitution. Further, closed expressions are divided into *canonical* and non-canonical, where the latter can be regarded as *programs* whose execution leads to the former. A signature σ is given by a pair $(\mathfrak{C}, \mathfrak{V})$, where $\mathfrak{C}$ is a finite set of sorted constants and $\mathfrak{V}$ is

a set of sorted variables. We can introduce a theory of arities which specifies our signature and then consider the set of expressions over this signature. The sorted constants will stand for type symbols and term symbols, that are not distinguished at this level. The whole theory of arities can be seen as a simply typed λ-calculus with a single ground type ι and arrow types $\rightarrow$. For each closed expression α, we fix the name, the arity and the form in Table 5.1.
The definition of the reduction relation $\alpha \Downarrow v$, where α is an arbitrary closed expression and v is a canonical closed expression (a *value*), represents the core of the operational semantics. Set-theoretically, we can think of $\Downarrow$ as the subset of the cartesian product $\mathcal{E}^{cl} \times \mathcal{V}$, where the following inclusions hold $\mathcal{V} \subset \mathcal{E}^{cl} \subset \mathcal{E}$. Moreover, canonical forms are computed according to a *lazy evaluation* strategy: an expression is in canonical form if the left outermost constructor is canonical according to the previous table. The lazy evaluation strategy is quite different from the traditional applicative order evaluation, that consists, instead, in evaluating first the constituents of a given expression (*evaluation from within*): e.g., a closed expression of the form $\mathsf{succ}(t)$ must be counted as canonical according to our definition, even if the sub-expression t is not.

Definition 5.4.2. The reduction relation $\Downarrow$ is the smallest relation generated by the following clauses:

(i) $v \Downarrow v$ (v value);

(ii) If $f \Downarrow \lambda(b)$ and $b(a) \Downarrow v$, then $\mathsf{Ap}(f, a) \Downarrow v$;

(iii) If $c \Downarrow \langle a, b\rangle$ and $d(a, b) \Downarrow v$, then $\mathsf{ind}_\Sigma(c, d) \Downarrow v$;

(iv) If $a \Downarrow 0$ and $b \Downarrow v$, then $\mathsf{ind}_\mathbb{N}(a, b, f) \Downarrow v$;

(v) If $a \Downarrow \mathsf{succ}(t)$ and $f(t, \mathsf{ind}_\mathbb{N}(t, b, f)) \Downarrow v$, then $\mathsf{ind}_\mathbb{N}(a, b, f) \Downarrow v$;

(vi) If $c \Downarrow \mathsf{inl}(a)$ and $d(a) \Downarrow v$, then $\mathsf{ind}_+(c, d, e) \Downarrow v$;

(vii) If $c \Downarrow \mathsf{inr}(b)$ and $e(b) \Downarrow v$, then $\mathsf{ind}_+(c, d, e) \Downarrow v$;

(viii) If $c \Downarrow \mathsf{sup}(a, f)$ and $b(a, f, x.\mathsf{ind}_\mathcal{W}(f(x), b)) \Downarrow v$, then $\mathsf{ind}_\mathcal{W}(c, b) \Downarrow v$.

The meaning of the basic judgement forms is condensed in the next definition:

Name	Arity	Form
Π	$\iota \to (\iota \to \iota) \to \iota$	Canonical
λ	$(\iota \to \iota) \to \iota$	Canonical
Ap	$\iota \to \iota \to \iota$	Non-Canonical
Σ	$\iota \to (\iota \to \iota) \to \iota$	Canonical
$\langle\, \rangle$	$\iota \to \iota \to \iota$	Canonical
ind_Σ	$\iota \to (\iota \to \iota \to \iota) \to \iota$	Non-Canonical
$\mathbb{N}$	ι	Canonical
0	ι	Canonical
succ	$\iota \to \iota$	Canonical
$\mathsf{ind}_\mathbb{N}$	$\iota \to \iota \to (\iota \to \iota \to \iota) \to \iota$	Non-Canonical
$+$	$\iota \to \iota \to \iota$	Canonical
inl	$\iota \to \iota$	Canonical
inr	$\iota \to \iota$	Canonical
ind_+	$\iota \to (\iota \to \iota) \to (\iota \to \iota) \to \iota$	Non-Canonical
Id	$\iota \to \iota \to \iota \to \iota$	Canonical
refl	ι	Canonical
$\mathcal{W}$	$\iota \to (\iota \to \iota) \to \iota$	Canonical
sup	$\iota \to (\iota \to \iota) \to \iota$	Canonical
$\mathsf{ind}_\mathcal{W}$	$\iota \to (\iota \to (\iota \to \iota) \to (\iota \to \iota) \to \iota) \to \iota$	Non-Canonical
$\mathcal{U}$	ι	Canonical
$\hat{\Pi}$	$\iota \to (\iota \to \iota) \to \iota$	Canonical
$\hat{\Sigma}$	$\iota \to (\iota \to \iota) \to \iota$	Canonical
$\hat{\mathbb{N}}$	ι	Canonical
$\hat{+}$	$\iota \to \iota \to \iota$	Canonical
$\hat{\mathrm{Id}}$	$\iota \to \iota \to \iota \to \iota$	Canonical

Table 5.1: Primitive Constants

Definition 5.4.3. Let $\mathcal{J}$ be a well-formed judgement. Then the meaning explanations for $\mathcal{J}$ are the following:

1. *Case* $\mathcal{J} \;:\equiv\; \Gamma \vdash A\ type$.

   Sub-case $\Gamma \;:\equiv\; \cdot$: to know $A\ type$ is to know $A \Downarrow V$ (V value) and to know how canonical elements of V are formed as well as when two such elements are equal.

   Sub-case $\Gamma \;:\equiv\; \Delta, x : C$: to know that $A\ type$ under the context $\Delta, x : C$ is to know that $A[c/x, \vec{t}/\Delta]\ type$ for any closed terms $c, \vec{t}$ of appropriate types, and, if $c \equiv c' : C$, $\vec{t} \equiv \vec{u}$, then $A[c/x, \vec{t}/\Delta] \equiv A[c'/x, \vec{u}/\Delta]$. The notation $\vec{t}$ stands for a sequence of terms $(t_1, \ldots, t_n)$ and the notation $\vec{t}/\Delta$ means that each variable in Δ has been substituted by a term t_i of the same type. *Presupposition*: we assume to know the judgements in the context.

2. *Case* $\mathcal{J} \;:\equiv\; \Gamma \vdash a : A$.

   Sub-case $\Gamma \;:\equiv\; \cdot$: to know $a : A$ is to know that $A \Downarrow V$, $a \Downarrow v$ (v value) and $v : V$. *Presupposition*: we assume to know $A\ type$.

   Sub-case $\Gamma \;:\equiv\; \Delta, x : C$ to know $a : A$ under the context $\Delta, x : C$ is to know that $a[c/x, \vec{t}/\Delta]$ is an element of $A[c/x, \vec{t}/\Delta]$ for any closed terms $c, \vec{t}$ of appropriate types, and that if $c \equiv c' : C$, $\vec{t} \equiv \vec{u}$, then $a[c/x, \vec{t}/\Delta] \equiv a[c'/x, \vec{u}/\Delta]$. *Presupposition*: we assume to know the judgements in the contexts and the judgement $\Gamma \vdash A\ type$.

3. *Case* $\mathcal{J} \;:\equiv\; \Gamma \vdash A \equiv B\ type$.

   Sub-case $\Gamma \;:\equiv\; \cdot$: to know $A \equiv B\ type$ is to know that $A \Downarrow V, B \Downarrow V'$ and $V \equiv V'$. Moreover, we know that a canonical element in V is also a canonical element in V' and vice-versa and that equal canonical elements in V are equal canonical elements in V' and vice-versa. *Presupposition*: we assume to know $A\ type, B\ type$.

   Sub-case $\Gamma \;:\equiv\; \Delta, x : C$: to know that $A \equiv B\ type$ under the context $\Delta, x : C$ is to know $A[c/x, \vec{t}/\Delta] \equiv B[c/x, \vec{t}/\Delta]\ type$ for any closed terms $c, \vec{t}$ of appropriate types, and that if $c \equiv c', \vec{t} \equiv \vec{u}$, then $A[c/x, \vec{t}/\Delta] \equiv B[c'/x, \vec{u}/\Delta]\ type$. *Presupposition*: we assume to know the judgements in the context as well the judgements $\Gamma \vdash A\ type$ and $\Gamma \vdash B\ type$.

4. *Case* $\mathcal{J} \ :\equiv \ \Gamma \vdash a \equiv b : A$.

   Sub-case $\Gamma \ :\equiv \ \cdot$: to know $a \equiv b : A$ is to know that $a \Downarrow v$, $b \Downarrow v'$ and $v \equiv v'$. *Presupposition*: we assume to know $A \ type, a : A$ and $b : A$.

   Sub-case $\Gamma \ :\equiv \ \Delta, x : C$: to know that $a \equiv b : A$ under the context $\Delta, x : C$ is to know $a[c/x, \vec{t}/\Delta] \equiv b[c/x, \vec{t}/\Delta] : A[c/x, \vec{t}/\Delta]$ and that if $c \equiv c' : C$, $\vec{t} \equiv \vec{u}$, then $a[c/x, \vec{t}/\Delta] \equiv b[c'/x, \vec{u}/\Delta] : A[c/x, \vec{t}/\Delta]$. *Presupposition*: we assume to know the judgements in the context as well the judgements $\Gamma \vdash A \ type, \Gamma \vdash a : A$ and $\Gamma \vdash b : A$.

Remark 5.4.4. Judgements are divided into *categorical* and *hypothetical*. The first are closed judgements, i.e. they do not contain open assumptions and hence are made in the empty context. A hypothetical judgement is a judgement containing open assumptions and thus depending on a non-empty context Γ. In the previous definition, hypothetical judgements are reduced to categorical ones by performing appropriate substitutions. The meaning explanations work with closed expressions and hence to attach a meaning to a hypothetical judgement, we are forced to consider its closed substitutional instances. The fact that the validity of judgements with open assumptions depends on the closed instances attached to them is held to be a basic tenet of the proof-theoretic semantics tradition, that has been called *the priority of the categorical over the hypothetical* by P.Schroeder-Heister [138].

Remark 5.4.5. The meaning of a judgement '$\vdash A \ type$' is given by knowing how to form an arbitrary canonical element of A as well as when two such canonical elements are equal. The introduction rule for A specifies what is the form of a canonical element of A.
The explanation of the first form of judgement closely corresponds to Bishop's notion of set:

> "A set is defined by describing exactly what must be done in order to construct an element of the set and what must be done in order to show that two elements are equal." ([15], pp.6-7)

Remark 5.4.6. In the previous definition, substitution is handled quite informally along the lines of [99]. In general, if $\Gamma \ :\equiv \ x_1 : C_1, \ldots, x_n : C_n$ and $A \ type \ [\Gamma]$, then $A[c_1/x_1, \ldots, c_n/x_n]$ denotes the result of the simultaneous substitution of terms $c_1, \ldots, c_n$ for variables $x_1, \ldots, x_n$. The same convention is assumed for substitution into terms. Moreover, it is assumed that substitution renames variables to avoid variable clashes.

It is with respect to this operational semantics that the rules of extensional type theory are validated. We can reframe the previous definition by introducing a notion of *validity* for judgements:

Definition 5.4.7 (Validity Categorical Judgements)**.** Let $\mathcal{J}$ be a categorical judgement. Then $\mathcal{J}$ is valid if one of the following cases hold:

- if $\mathcal{J} :\equiv \cdot \vdash A\ type$, then $A \Downarrow V$ and the constituents of V are valid;
- if $\mathcal{J} :\equiv \cdot \vdash a : A$, then $a \Downarrow v$, $A \Downarrow V$ and $v : V$;
- if $\mathcal{J} :\equiv \cdot \vdash A \equiv B$, then $A \Downarrow V$, $B \Downarrow V'$, $V \equiv V'$ is derivable and the constituents of V, V' are valid;
- if $\mathcal{J} :\equiv \cdot \vdash a \equiv b : A$, then $a \Downarrow v$, $b \Downarrow v'$, $v \equiv v'$ is derivable and $A \Downarrow V$.

Definition 5.4.8 (Validity Hypothetical Judgements)**.** Let $\mathcal{J}$ be a hypothetical judgement and let us assume that $\Gamma :\equiv x_1 : C_1, \ldots, x_n : C_n$, $\vec{c}$ is a sequence of terms $c_1 : C_1, \ldots, c_n : C_n$ and $\vec{c}/\Gamma$ represents the substitution of the free variables in Γ with the terms of the sequence $\vec{c}$. Then $\mathcal{J}$ is valid if one of the following holds:

- if $\mathcal{J} :\equiv \Gamma \vdash A\ type$, then for any $\vec{c}/\Gamma$, $A[\vec{c}/\vec{x}]\ type$ is valid and if $\vec{c} \equiv \vec{d}$, then $A[\vec{c}/\vec{x}] \equiv A[\vec{d}/\vec{x}]$ is valid;
- if $\mathcal{J} :\equiv \Gamma \vdash a : A$, then for any $\vec{c}/\Gamma$, $a[\vec{c}/\Gamma] : A[\vec{c}/\Gamma]$ is valid and if $\vec{c} \equiv \vec{d}$, then $a[\vec{c}/\Gamma] \equiv a[\vec{d}/\Gamma] : A[\vec{c}/\Gamma]$ is valid, provided that $A\ type$ $[\Gamma]$ is valid;
- if $\mathcal{J} :\equiv \Gamma \vdash A \equiv B\ type$, then for any $\vec{c}/\Gamma$, $A[\vec{c}/\vec{x}] \equiv B[\vec{c}/\vec{x}]$ is valid and $\vec{c} \equiv \vec{d}$, then $A[\vec{c}/\vec{x}] \equiv B[\vec{d}/\vec{x}]$ is valid, provided that $A\ type$ $[\Gamma], B\ type$ $[\Gamma]$ are valid;
- if $\mathcal{J} :\equiv \Gamma \vdash a \equiv b : A$, then for any $\vec{c}/\Gamma$, $a[\vec{c}/\vec{x}] \equiv b[\vec{c}/\vec{x}] : A[\vec{c}/\vec{x}]$ is valid and if $\vec{c} \equiv \vec{d}$, then $a[\vec{c}/\vec{x}] \equiv b[\vec{d}/\vec{x}] : A[\vec{c}/\Gamma]$ is valid, provided that $a : A$ $[\Gamma], b : A$ $[\Gamma]$ and $A\ type$ $[\Gamma]$ are valid.

At this point, following Martin-Löf's informal explanations in [99], the following result can be easily derived:

Theorem 5.4.9 (Simple Minded Validity)**.** *If $\mathcal{J}$ is a derivable judgement, then $\mathcal{J}$ is valid.*

The proof is by structural induction and consists in establishing the correctness of all the rules of the theory. The name *simple minded* validity is inspired by the simple minded consistency in ([99], p.37), that is contrasted there with metamathematical consistency. If we understand meaning explanations in a purely informal way and we test the correctness of the rules with respect to them, then we end with a simple minded validity theorem, that implies the simple minded consistency result discussed by Martin-Löf:

Corollary 5.4.10 (Simple Minded Consistency)**.** *There is no derivable closed judgement of the form $\cdot \vdash c : \mathbf{0}$.*

Because let us suppose that such judgement is derivable, then its validity amounts to $c \Downarrow v$ for v canonical element of the empty type, but there is no such element.
Canonicity and Extensionality. Let us consider some rules to understand how the method works. In the case of Π-F, let us assume that the premisses are valid (in the empty context), then the conclusion immediately follows since $\Pi_{(x:A)}B(x)$ is canonical. If the context is not empty, then the result follows by induction hypothesis. Moreover, the meaning explanations validate also the congruence rules associated to type and term constructors. In particular, in the case of Π-I, the congruence rule is the ξ-rule and is valid w.r.t. the operational semantics. Finally, Π-E is interpreted as follows: by induction hypothesis, the judgements $\cdot \vdash f : \Pi_{(x:A)}B(x)$ and $\cdot \vdash a : A$ are valid, which means that $f \Downarrow \lambda x.b(x)$ and then the substitution $b[a/x]$ yields a term in $B[a/x]$. Then, $b(a) \Downarrow v$ and hence $\mathsf{Ap}(f, a) \Downarrow v$ by the computation rule.
The case of ℕ-E rule is useful to highlight some significant features of the meaning explanations. Let us assume that the premisses are valid and they are $C(x)$ *type*$[x : \mathbb{N}]$, $c : C(0)$, $d(y, z) : C(\mathsf{succ}(y))[y : \mathbb{N}, z : C(y)]$ and $a : \mathbb{N}$. In particular, the validity of $a : \mathbb{N}$ means that the computation of a yields either 0 or $\mathsf{succ}(m)$ for some $m : \mathbb{N}$.
If $a \Downarrow 0$, then $c \Downarrow v : C(0)$ that is the value of $\mathsf{ind}_{\mathbb{N}}(a, c, y.z.d(y, z)) : C(a)$ by the computation rule and $a \equiv 0$.
If $a \Downarrow \mathsf{succ}(m)$, then by substitution

$$d[m/y, \mathsf{ind}_{\mathbb{N}}(m, c, y.z.d(y, z))/z] : C(\mathsf{succ}(m))$$

that is valid and yields a canonical element v' that is also the value of $\mathsf{ind}_{\mathbb{N}}(a, c, y.z.d(y, z)) : C(a)$ by the computation rule and $a \equiv \mathsf{succ}(m)$. Then we continue with $m : \mathbb{N}$. If $m \Downarrow 0$, we proceed as in the first case and we are done; if $m \Downarrow \mathsf{succ}(m')$, then we proceed as in the second case until we reach the value 0.

The explanation of the rules for the natural numbers has the interesting consequence that *canonicity* holds: if c is a closed term of type $\mathbb{N}$, then c either reduces to 0 or to $\mathsf{succ}(\mathsf{succ}(\ldots \mathsf{succ}(0) \ldots))$. In general, the operational semantics guarantees that every derivable closed judgement is in canonical form. The canonicity theorem is the strongest result that we can achieve in the case of extensional type theory, where a normalization procedure is not available. However, many corollaries of normalization still hold in this case (e.g. consistency).

The induction rule for $\mathbb{N}$ is justified by an analogous induction principle over the meta-linguistic expressions interpreting the terms and the types of the system. This fact can be interpreted as an instance of *pragmatic* circularity in the sense of Dummett [40]. Dummett introduced the distinction between 'gross' and pragmatic circularity in his discussion of the justification of logical laws. The traditional skepticism about the possibility of achieving such a justification is summarised in the remark that any purported deductive justification of a fundamental logical law must assume the law itself at some point in the argument and hence is inevitably circular. Dummett's proposal is to escape the skeptical conclusion by a distinction between different kinds of circularity and by focusing on the *purpose* of the argument considered. The deductive justification of fundamental logical laws is not blamed for gross circularity that amounts to include the conclusion among the initial premisses. It is circular in the sense that at least one inferential step in it is an instance of the law that must be justified. There is wide agreement about the fact that gross circularity is vicious and cannot serve any epistemic purpose. Dummett's claim is that pragmatic circularity, instead, is not necessarily vicious and, in some circumstances, may play an effective epistemic role. If this is the case depends on the purpose of the argument in which it occurs. According to Dummett, there exists a significant distinction between *suasive* arguments and *explanatory* arguments. The former are aimed to persuade us that the conclusion is true. The latter, on the other hand, are intended to explain why the conclusion is true. The conclusion is taken as a *fact* in need for an explanation, not as a doubtful statement. The philosopher's perplexities about a logical laws fall under the latter case: there is no genuine

doubt about the truth of the conclusion, but the request is to show why the conclusion is true and hence the law is justified. Therefore, an explanatory argument is needed.

Dummett observes that if a suasive argument would be required, then any circular reasoning would not be effective. However, in the case of explanatory arguments, at least some form of circularity might turn out to be of some epistemic value and thus count as genuine explanations. The pragmatic circularity involved in the justification of logical laws is an example of that.

Meaning explanations as conceived by Martin-Löf can be legitimately counted as explanatory arguments: they are meant to explain the meaning of type-theoretic judgements and hence to justify the rules of inference they are built from. Then, it can be maintained that they involve instances of pragmatic circularity. Furthermore, meaning explanations result from an analysis of meta-mathematical normalisation and canonicity arguments: they unravel the pre-mathematical semantic content of these proofs. Under this perspective, it is not surprising that they *reflect*, so to speak, the rules they are meant to justify at the meta-level. In fact, normalisation and canonicity arguments are held to have a greater epistemic value if they are carried out in a meta-theory that is as close as possible to the theory under consideration. Then, the alleged circularity turns out to be a desirable feature from the point of view of the epistemic content.

Meaning explanations are tailored for the extensional version of type theory. The reflection rule is naturally validated by this semantics. In fact, the reduction relation applies to closed terms and in the case of propositional equality this immediately entails the corresponding definitional equality, since $p : \mathrm{Id}_A(a, b)$ if and only if $a \equiv b : A$. Also, the computation rule is validated, because the computation of p yields a reflexivity element and thus, in the homotopical jargon, the only paths are reflexivities. Therefore, the meaning explanations as they stand cannot make sense of the higher-dimensional structure that is generated by intensional identity types. This seems to be a major defect of the method and is essentially related to the insistence on the priority of the evaluation of closed expressions to canonical form. On the other hand, the task of designing meaning-explanations dealing directly with open expressions is not trivial. Moreover, they should be crafted in a way that would not be parasitic on normalization procedures, that are known to apply directly to open expressions. The demand is to show that such meaning explanations for open expressions represent a genuine semantics in Martin-Löf's sense and are not a purely meta-mathematical interpretation.

Finally, I want to examine the meaning explanations for the elimination rule of $\mathcal{W}$-types. An element $c : \mathcal{W}_{(x:A)}B(x)$ yields as value a canonical element of the form $\mathsf{sup}(a, f)$, where $a : A$ and $f : B(a) \to \mathcal{W}_{(x:A)}B(x)$. Then, for any x in $B(a)$ we have a collection of subtrees ranging over $B(a)$ that are the immediate predecessors of $\mathsf{sup}(a, b)$. The application of f to $x : B(a)$ yields a new element of the form $\mathsf{sup}(a_1, f_1)$ and we can iterate the computation until we reach a canonical element $\mathsf{sup}(a_n, f_n)$ such that the type $B(a_n)$ is empty. Then, the elimination rule is validated by the following reasoning: let us assume that the judgements in the premises are valid, i.e. $C(w)$ *type* $[w : \mathcal{W}_{(x:A)}B(x)]$, $c : \mathcal{W}_{(x:A)}B(x)$, $d(x, y, z) : C(\mathsf{sup}(x, y))[x : A, y : B(x) \to \mathcal{W}_{(x:A)}B(x), z : \Pi_{(v:B)}C(y(v))]$. Then, in particular, $c \Downarrow \mathsf{sup}(a, f)$. By substituting a for x, f for y and $\kappa :\equiv \lambda v.\mathsf{ind}_{\mathcal{W}}(f(v), x.y.z.d)$ for z, that is the proof that C holds for all the sub-trees of $\mathsf{sup}(a, f)$, we get

$$d(a, f, \kappa) : C(\mathsf{sup}(a, f))$$

and by the computation rule and $c \equiv \mathsf{sup}(a, f)$, it yields the same value as $\mathsf{ind}_{\mathcal{W}}(c, x.y.z.d) : C(c)$.

What is interesting in this kind of argument is to analyse what we are implicitly assuming. The elimination rule for $\mathcal{W}$-types is a principle of transfinite induction over well-founded trees asserting that if C is an inductive predicate, i.e. C holds for a canonical element $\mathsf{sup}(a, f)$ whenever C holds for all the sub-trees of this element that are enumerated by f, then C holds for an arbitrary element of $\mathcal{W}_{(x:A)}B(x)$. Such a principle correspond to a generalized form of *bar induction*. Indeed, we assume that the computation must end at some point, namely when we reach a value $\mathsf{sup}(a_n, f_n)$ such that $B(a_n)$ is empty and thus the whole structure is well-founded. That a similar assumption was behind the meaning explanation of $\mathcal{W}$-types was already observed by Troelstra and Van Dalen ([158], p.617):

> "Thus in order to show that $t_0 \in \mathsf{Tr}$, according to the semantics we have to see that an arbitrary path obtained by successive choices $v_1, v_2, v_3 \ldots$ terminates in an element $\mathsf{sup}(\mathbf{0}, t)$. On the other hand WI and WE tell us that Tr is inductively generated. Accepting this semantical interpretation as a justification of the rules WI and WE for Tr amounts to accepting bar induction. The whole story can be repeated for the general case: there the acceptance of WI and WE on the basis of semantical considerations of this

nature amounts to the acceptance of a generalized form of bar induction."

The type Tr they are referring to is the type of the countably branching trees. Such a type is defined as

$$\mathsf{Tr} \ :\equiv \ \mathcal{W}_{(x:\mathbf{2})}B(x)$$

where $B(x) \ :\equiv \ \Sigma_{(y:\mathbb{N})}\mathrm{Id}_{\mathbf{2}}(x, 1_2)$. Then, $B(0_2)$ is empty and thus for any type C, there exists a map $\lambda x.R(x) : B(0_2) \to C$, which guarantees that Tr is not empty, since we can form $\mathsf{sup}(0_2, \lambda x.R(x))$. On the other hand, $B(1_2)$ amounts to a copy of the natural numbers and given a countable sequence of trees $t : B(1_2) \to \mathsf{Tr}$, we can form a new tree by introducing $\mathsf{sup}(1_2, t)$. The elimination rule in this case corresponds to bar induction.
Brouwer's argument for bar induction has been heavily scrutinised and thus the literature on this topic is quite extensive. I want to limit my attention to a specific interpretation that might be relevant for the present case. In particular, I am referring to the analysis of the meaning of Π^1_1-sentences advocated by Martin-Löf in [94]. Classically, a well-founded tree is a tree with no infinite descending chain w.r.t. the order $\preceq$. It is known that such structures are codified by Π^1_1-sentences, in the sense that the following equivalence is classically derivable

$$T \text{ well-founded tree } \leftrightarrow \varphi$$

where φ is Π^1_1. Also, the principle of induction over these structures is trivially valid classically and corresponds to Bar Induction. Let us define the set of infinite sequences of natural numbers as follows: $Seq \ :\equiv \ \omega^\omega$. Let α be an element of Seq; the initial segment of α for a natural number n is denoted by $\overline{\alpha}(n) \ :\equiv \ \langle \alpha(0), \ldots, \alpha(n-1) \rangle$; the concatenation operation is denoted by $\alpha^\frown n$, for any n and any α. Martin-Löf defines A *bars* α as follows:

1. if $A(\alpha)$ holds, then A *bars* α
2. if for any n, A *bars* $\alpha^\frown n$, then A *bars* α.

Then a formula of the form $\forall\alpha\exists n A(\overline{\alpha}(n))$ holds iff the empty sequence $\langle \ \rangle$ is barred by A. In this context, the Brouwerian arguments that appear in the proof of the Bar Theorem are taken as arguments supporting the correctness of the above definition. In particular, A *bars* $\langle \ \rangle$ must be understood as 'there

exists a canonical proof of A *bars* $\langle\ \rangle$', the assumption being that a canonical proof provides a constructive explanation of what it means that a Π^1_1-sentence holds. More generally, bar induction is an instance of

$$\forall X.A(X) \to A(B)$$

where $A(X)$ is arithmetical. The constructive meaning corresponds to the existence of a procedure transforming any canonical proof of the antecedent into a canonical proof of $A(B)$. Therefore, the notion of canonical proof is fundamental to explain the meaning of Π^1_1 sentences and if this explanation is accepted, then a large class of mathematical objects can be regarded as constructively admissible: among them there are constructive ordinals and Borel sets [94].

5.4.3 Generalised Predicativity and its Limits

In the following, I want to analyse some ideas that might clarify in which sense generalised predicative constructive systems are predicative and review some conjectures about the limits of generalised predicative constructivism.
Meaning Explanations and Well-Foundedness. The claim I want to address is that for which the meaning explanations *show* that the types are built up by means that are predicatively acceptable. More precisely, the iterated evaluation of closed expressions to canonical form yields a direct insight into the well-founded structure of the types. Following Martin-Löf's semantical method, a canonical form theorem is obtained which states that any closed term of a given type is in canonical form. Moreover, this implies that the computation generated by the execution of an arbitrary closed term t must end at some point. The resulting picture of types is that of well-founded structures whose elements are built up from below by iterated applications of the relevant introduction rules. Further, the meaning explanations validate the induction principles relative to these structures asserting that in order to prove something about an arbitrary element of the structure is enough to prove it for the canonical elements.
Recently, Van Atten [7] has investigated the way inductive definitions are conceived in intuitionistic mathematics. He claims that intuitionistic inductive definitions are governed by the grammatical mood of the *imperative*. Van Atten's interpretation is also built on a similar remark by Heyting reported in [7], which links the centrality of constructions in intuitionistic mathematics with the imperative mood:

> "The adequate description of an intuitionist theory would be composed of a description of the results of constructions carried out in advance (or assumed), a description of the construction we wish to make, preferably in terms of imperatives, and observation of the result obtained [...] It is mostly the lack of a grammar of the imperative sufficiently linked to that of the indicative used to express the result of a construction, that makes ordinary mathematical language so deficient for intuitionism"

The role of imperative sentences must be read in the light of Brouwer's conception of definitions. A definition is regarded as a restriction on our acts of construction. An object is well-defined and *exists* if it is the result of acts of construction that are delimited by a given definition. The substantial aspect is that the need for an existence proof of the object is bypassed. It can be appreciated especially in the case of inductive definitions: a set X that is inductively defined in set theory is first explicitly characterized as the least set such that a certain condition holds and *then* it is proved to exist. Within an intuitionist framework, instead, an inductive definition is interpreted as a set of *instructions* or *commands* that prescribe precise acts of constructions. Then, the object resulting from these acts of construction is the inductively defined set and if this is infinite, then exists only as a potential totality. It is the definition itself which guarantees the existence of the object. Van Atten's analysis of the inductive definition of the set of natural numbers amounts to the following:

1. `build` 0!
2. `if` n `has been built`, `then build` $\mathsf{succ}(n)$!

and is assumed that we can iterate the acts prescribed by these commands. The connection with Martin-Löf's meaning explanations is quite transparent. Also, it appears that this analysis can be extended to the case of generalized inductive definitions. Let us consider the inductive clauses of constructive ordinals, they can be interpreted as follows:

1. `build` 0!
2. `if` α `has been built`, `then build` $\mathsf{succ}(\alpha)$!
3. `if` $\alpha_1, \ldots, \alpha_n$ `has been built`, `then build` $\mathsf{sup}(\alpha_n)$!

Meaning explanations could be seen as a refinement of this perspective: the introduction rules are interpreted as instructions to constructs elements in canonical form and the elimination rule is interpreted in terms of execution of non-canonical elements that ends with a canonical value and hence guarantees that the definition is minimal. Again, meaning explanations appear to be a way for making more precise the intuitionist semantical explanations and supply the latter with conceptual distinctions coming from the proof-theoretic tradition. Furthermore, the result of the acts of construction is syntactically represented by proof-objects. In this sense, it is assumed that canonical elements must have a linguistic representation, differently from the Brouwerian perspective.

Making sense of inductive definitions. One could legitimately question whether these explanations in terms of instructions really achieve anything. After all, one could say, they just take the inductive definitions in their unanalysed form and attach to them some verbal expressions. However, the definitions are not converted to an explicit form as it happens instead when they are analysed set-theoretically. *Prima facie*, it seems that something is made explicit, namely the relevant acts of construction, and thus the explanations are not entirely empty. Each clause is interpreted as an explicit command and if a definition is conceived as a way of putting constraints on our acts of construction, then we get that such constraints are spelled out. We could say that even more is achieved, if we consider the refined analysis arising from the meaning explanations and the whole discussion takes place within the context of intuitionistic type theory, which can be legitimately regarded as a formal theory of constructions. The introduction rules are meant to specify means to build canonical elements and the meaning of a type is explained in terms of these rules. Furthermore, the meaning explanations provide a justification for the associated induction principles expressed as elimination rules and the associated computation rules. In this way, the meaning of a judgement is analysed in operational terms (computation of closed expressions to values) that are naturally inclined to an instruction-like reading. Moreover, this semantics extends to inductive definitions of different complexity and not only to ordinary ones. Finally, a major innovation with respect to the intuitionist tradition is that constructions can be always linguistically represented and canonical means of constructions result in canonical elements that are *named* in the syntax.

The outcome of the previous discussion is that inductive definitions can be analysed in terms of instructions or commands and Martin-Löf's meaning

explanations are a refinement of this perspective. Moreover, the whole discussion is meant to guarantee that this analysis is really explicative and thus provides a more direct ways for understanding inductive definitions avoiding any detour through the set-theoretical analysis that is blatantly impredicative. The latter point is fundamental, because it is meant to vindicate the idea that inductive definitions can be understood as they stand. Another argument in support of this view has been given by Parsons [118] in the context of a discussion about the predicative status of generalised inductive definitions in Lorenzen's work. His claim is that, granted a *certain* conception of set, Lorenzen's view can be defended. The conception of set in question is the *logical* one: a set must be understood as the *extension of a corresponding predicate*. From an historical perspective, it appears that early predicativists such Poincaré and Weyl held precisely this view. The main point is that there exists a *priority* relation between the concept of set and that of predicate. Such a relation is an epistemic one and consists in the fact that set formation *presupposes* that the relevant predicate has been already understood. In this sense, according to the logical conception, predicates are prior to sets. As an aside, I want to point out that the idea that there exists an order of epistemic priority between concepts is also endorsed by Martin-Löf: the fundamental concepts of logic come equipped with a precise order of conceptual dependency and their explanation must respect this order. This idea of conceptual priority is the key to achieve a justification of the predicative character of generalised inductive definitions. Indeed, generalised inductive definitions are meant to introduce predicates and not sets. The introduction of these predicates does not involve any circular reasoning: objects falling under these predicates are given by iterated applications of the relevant introduction rules and there is no appeal to a totality including the predicates themselves. Sets are formed only *later* by applying an appropriate comprehension principle that presupposes that the corresponding predicates have been correctly introduced. It is only when the priority relation is not taken into account that inductive definitions turn out to be circular. In a set-theoretic framework, which rests on the iterative conception of set, the priority relation between sets and predicates is devoid of any meaning, since sets are not understood primarily as extensions of predicates.

I think that the whole argument can be carried over to the context of type theory. The relevant distinction is that between types and *iterative sets*, which closely correspond to sets as understood by set theorists. The argument is

made to go through by claiming that types are epistemically prior to sets and what generalised inductive definitions introduce are the former. Then, the set-theoretic analysis of inductive definitions is not the privileged explanation. A striking feature is that the priority claim can be formally substantiated in the type-theoretic context. Indeed, Aczel's interpretation of CZF in type theory [1] elucidates this point. The interpretation works by introducing a type of iterative sets $\mathbb{V}$, which is an instance of a $\mathcal{W}$-type. A notion of extensional equality can be defined between elements of $\mathbb{V}$ and also a membership relation. They are denoted by $=_{\mathbb{V}}$ and $\in_{\mathbb{V}}$ respectively and thus the set-theoretic axioms are interpreted in the structure $(\mathbb{V}, =_{\mathbb{V}}, \in_{\mathbb{V}})$. This structure is intended to simulate the set-theoretic universe and elements of $\mathbb{V}$ are just sets. Then, the type of iterative sets turns out to be just a type among the others and the notion of type appears to be more fundamental than that of sets. Finally, it is known that the interpretation can be extended to ZF if the law of excluded middle is assumed and also to ZFC if the extensional AC is assumed [103]. Therefore, the analysis applies also to classical theories under the assumption of non-constructive principles.

Actual Infinity and building up from below. A seminal paper for generalised predicative constructivism is the one by Lorenzen and Myhill [85]. There a classification is proposed of ways to form sets with the aim of singling out those ways which are predicatively acceptable. In particular, the strongest way that is predicatively admissible is what they call 'Method IV' and is exemplified by Lorenzen's operative approach. It amounts to the method of introducing sets by inductive definitions with the quantifiers ranging only over the denumerable totality of sets that have been already obtained. The predicative status of generalised inductive definitions is advocated by pointing out that such definitions are fully compatible with the rejection of actual infinity. Then, the key aspect of predicativity is taken to be the rejection of the actualist view of infinite totalities in compliance with Poincaré's early analysis. It follows that only Method V in the Lorenzen-Myhill classification must be counted as impredicative: it comprehends those definitions including unrestricted quantification over subsets. However, they claim that classical mathematics for the most part can be developed predicatively avoiding definitions of the latter kind, as soon as generalised inductive definitions are held to be predicatively acceptable.

Quite interestingly, Lorenzen and Myhill stressed the problem of providing a quantitative measure for generalised predicativity. In other words, they point out that it would be useful to obtain a sharp bound on the notion of

definability by generalised predicative means. The problem is still open in the context of type theory, where complex kinds of inductive definitions are allowed and are held to be predicative in some sense. Therefore, the challenge of identifying the limits of generalised constructive predicativism remains to be addressed.

The way inductive definitions are analysed in type theory appears to be faithful to the fundamental insight of Lorenzen and Myhill. The constructive appeal of inductive definitions is due to a large extent to the fact that sets arising from them can be seen as if they were built up from below [31]. In particular, this means that they are built in a step-by-step fashion and at each stage the construction applies only to those elements that have been already obtained without any circular reasoning. In this respect, they naturally fit into a *genetic* conception of sets, that appears to underlie the main predicativist schools. The metaphor of building up from below turns out to be quite powerful, since it suggests that inductive definitions could be interpreted in a way that is unobjectionable from a predicative standpoint. The set-theoretic analysis which converts the inductive clauses into an explicit definition destroys this image of step-by-step construction. On the other hand, the idea of interpreting inductive definitions in terms of instructions for forming objects and the refinement of this insight provided by the meaning explanations preserves the value of the metaphor. Moreover, they make clear the well-founded structure of inductively defined types. In general, all these constructions can be understood in a potential way and also the reflection process that leads to the introduction of a universe of types can be predicatively understood. The universe itself is, according to Martin-Löf, open-ended and hence we cannot assume at any stage the totality of types as a completed and closed totality. In this sense, the idea that the rejection of methods of definitions involving infinite actual totalities is what truly characterises the predicative conception appears to be preserved in type theory.

Limits of Generalised Predicativity. Finally, I want to discuss the problem of identifying the limits of generalised predicativity and review a conjecture by Rathjen about this issue formulated in [130].

The question of identifying the limits of generalised predicativism in the context of intuitionistic type theory is quite ambiguous. The possibility of iterating the reflection process by the introduction of different and powerful universe constructions entails that whenever a collection of types is given and a limit is identified, we can transcend it and get a stronger system. The variety

of universe constructions in the literature reinforces this point: e.g., Setzer's construction of a type-theoretic Mahlo universe provides a striking example, since it corresponds to a *non-monotonic* inductive definition. However, Setzer argues that the construction is predicative and can be justified on the basis of the meaning explanations method. Rathjen's attempt to identify the limits of intuitionistic type theory and thus, arguably, of generalised predicative constructivism rests on the assumption that we can classify all the methods and the principles that are held to be admissible by an idealized type-theorist:

> "Nonetheless, it appears to be possible to transcend the type theorists realm by adopting, as it were, a hypothetical 'eagles' point of view as in classical set theory and reflect from such an advanced position on all possible moves the type theorist can ever perform. This thought experiment should then allow one to delineate bounds for the realm which the type theorist will never be leaving, or, more precisely, it should enable the Cantorian set theorist to draw a line in his set-theoretic world beyond which the type theorist will never be able to reach." ([130], p.111)

Then, the idea is to adopt impredicative methods to circumscribe the limits of intuitionistic type theory. The idea of adopting an external point of view to analyse formally a given conception of predicativity has been fruitfully applied in ordinal proof theory. The main challenge then is to provide a classification of the basic principles underlying type-theoretic constructions. Rathjen tentatively suggests the following basic principles:

1. Predicativism: the collection of types is built in stages and when we specify what is an element of a given type we cannot make reference to the totality of types.

2. Meaning explanation of the judgement A *type* in terms of canonical elements and equal canonical elements.

3. Canonical elements must allow for a linguistic representation in a language including countably many basic symbols as well as elements of previously introduced types.

4. Every inductive definition is definable by means of Σ-formulas in the ambient set theory. Such formulas may include other sets as parameters,

that correspond to types which have already been obtained. This last principle is meant to provide an upper bound for inductive definitions.

Rathjen concludes, on the basis of this classification, that the limit of intuitionistic type theory is a suitably modified version of Kripke-Platek set theory extended with an axiom stating that every set belongs to a transitive set that is a Σ_1 elementary substructure of the set-theoretic universe. Then, every construction admissible in type theory can be simulated in this system. A different approach (but with a quite similar conclusion) has been pursued by Martin-Löf in [104]. There, the "second failure" of Hilbert's program is discussed. The first failure of course was represented by Gödel's incompleteness theorems. Martin-Löf's claim is that the recent developments of ordinal proof theory show that also the extension of Hilbert's program from finitary or combinatorial methods to more abstract constructive ones is doomed to fail. He maintains that what the most advanced results show is that the consistency of classical analysis cannot be established by purely constructive methods. Rather, even subsystems of classical analysis are not amenable to a constructive analysis. Martin-Löf reviews the most significant results in ordinal analysis and their connection with systems of type theory of different strength: e.g., it is known that type theory with $\mathcal{W}$-types and without universes is of the same strength as the theory of finitely iterated generalised inductive definitions and second-order arithmetic with Π^1_1-comprehension and Π^1_1-induction. The outcome of the analysis is that at the level of the subsystem of second order arithmetic with Π^1_2 comprehension a natural barrier to constructive methods is found. The proof-theoretic analysis of this system relies on heavy set-theoretic assumptions and requires a substantial change in the traditional methodology of ordinal analysis. These methods do not appear to be amenable to a constructive treatment, at least not with the constructively acceptable methods that are known. Then, the limits of current constructive methods appear at a level of complexity which is relatively low when compared with full second order arithmetic. Obviously, Martin-Löf is aware that the methods that are recognised as constructively meaningful are not fixed once and for all. Nonetheless, he is inclined to the pessimistic conclusion that the problem of consistency of classical analysis cannot be solved constructively and thus that impredicative methods are what really discriminates between classical and constructive mathematics. In this sense, the limits of generalised predicativity are close to what has been conjectured by Rathjen.

Conclusion

This work has explored the multifaceted aspects of fundamental foundational concepts such as extensionality and predicativity through the lens of Martin-Löf Type Theory and other related type theories. The subtle relationship between intensional and extensional aspects has been clarified and the results in Ch. 3 and Ch.4 reinforce the claim that type theory offers a privileged perspective on these issues. Further, the notion of generalized predicativity has been analyzed in the last Chapter and has been compared to the notion of predicativity given the natural numbers, especially in relation with Weyl's work. The study of the constructive notion of predicativity is still an underdeveloped subject, in particular from a philosophical standpoint. The arguments I discussed in Ch.5 are meant to provide an overview of the ideas which motivate this notion, but they should be further worked out. In general, the idea of focusing on the concepts of predicativity and extensionality has been also suggested by the fact that they are closely related to other fundamental notions in logic and foundations of mathematics. In this sense, the investigation of these concepts within the context of type theory sheds light on other important issues and can be pursued along different axes: it can lead to interesting mathematical results as well as to finer philosophical analyses. One of the main aims of the present work was to attempt to develop these different perspectives together and I hope that it was at least partially successful. I want to conclude by pointing out some open problems and possible directions of research, which are motivated by this work:

- Possible extensions of inverse diagram models to other type theories. In particular, it is possible to define such models over more general inverse categories and it would be interesting to figure out what are the syntactical counterparts of these constructions. The relation with parametricity should also be explored: e.g., the relation model of Ch.3 should be anti-classical, in the sense that the interpretation does not

valid classical logical principles.

- The general philosophical background which motivates the principle of invariance should be investigated in connection with relevant examples from the history of mathematics. Moreover, it would be of interest to consider areas of mathematics which at least *prima facie* are less prone to a 'structural' methodology.

- Another direction is to work out the alternative interpretation of **emTT** in **HoTT**, which I have outlined in the subsection 4.4.1. The main problem is to analyze the relationship with the direct interpretation of **emTT** in **HoTT** in order to conclude that this alternative construction gives the compatibility between the two theories. Further, the interpretation of **emTT** in **HoTT** defined in Ch.4 could be extended to a version of **emTT** including inductive and coinductive definitions as in [90] and by considering a novel formulation of **HoTT** with a *superuniverse*. Finally, I point out that the possibility of employing a notion of heterogeneous equality should be explored in light of the relation with the works [114, 171].

- In general, all the interpretations defined in Ch.3 and Ch.4 are partial and are based on the introduction of a raw-syntax. The possibility of adopting an *intrinsic* representation of the syntax instead, in the sense of [6], which would allow to avoid the introduction of the untyped syntax, could be investigated.

- We have seen that in type theory there are multiple possible formalizations of logical notions. In particular, the status of choice principles is related to the way the existential quantifier is represented. A possible line of research is to investigate the status of choice principles within the minimalist framework provided by **MF**. In this setting, it is possible to discriminate among different formulations of choice principles and assess the relations between them. Moreover, working out such approach would allow to make more fine-grained distinctions between mathematical concepts. A basic example is the distinction between operations and functional relations, which is preserved in **MF** since the axiom of unique choice is not derivable in it. On a more philosophical level, I believe that this approach would provide a more precise view of the constructive notion of existence.

- The theory $\mathfrak{LM}$ introduced in Ch.5 should be further investigated in different directions. First, the validity of Palmgren's functional reducibility axiom in the model of h-sets and h-propositions must be proved. Then, it would be possible to consider also other axioms which are naturally validated by the interpretation. Other models, in particular set-theoretical and categorical models, could be defined: singling out the categorical structure corresponding to the constructors of the theory might be an interesting research topic. Another open problem is to define the conjectured inverse translation of the sub-system $\mathfrak{LM}_0$ into $\mathbf{ACA}_0$. Finally, it could be also worthwile to attempt to formalize some mathematics in $\mathfrak{LM}$, e.g. basic constructions from real analysis.

- The connection between the meaning explanations for $\mathcal{W}$-types and a general form of bar induction is a fascinating topic. A first problem is to evaluate the arguments given in the literature to justify bar-induction in the light of this connection. An interesting open problem is to consider a possible extension of Martin-Löf's meaning explanations to *open* expressions which is not parasitic on a normalization argument, even if it could be derived from it. In particular, this would entail the possibility of defining meaning explanations tailored for the intensional type theory, which do not validate strong extensional rules such as the reflection rule.

Bibliography

[1] P. Aczel. The type theoretic interpretation of constructive set theory. In *Studies in Logic and the Foundations of Mathematics*, volume 96, pages 55–66. Elsevier, 1978.

[2] R. Adams and Z. Luo. Weyl's predicative classical mathematics as a logic-enriched type theory. In *International Workshop on Types for Proofs and Programs*, pages 1–17. Springer, 2007.

[3] R. Adams and Z. Luo. Classical predicative logic-enriched type theories. *Ann. Pure Appl. Log.*, 161(11):1315–1345, 2010.

[4] R. Adams and Z. Luo. Weyl's predicative classical mathematics as a logic-enriched type theory. *ACM Transactions on Computational Logic (TOCL)*, 11(2):1–29, 2010.

[5] A.Lindendbuam and A.Tarski. Über die Beschränktheit der Ausdrucksmittel deduktiver Theorien. *Ergebnisse eines mathematischen Kolloquiums*, 7:15–22, 1936.

[6] T. Altenkirch and A. Kaposi. Type theory in type theory using quotient inductive types. In *POPL*, pages 18–29. ACM, 2016.

[7] M. V. Atten. Intuitionistic inductive definitions. 14th French PhilMath Workshop (Nantes, 19-21 October), 2022.

[8] M. V. Atten and G. Sundholm. L.e.j. brouwer's 'unreliability of the logical principles': A new translation, with an introduction. *History and Philosophy of Logic*, 38(1):24–47, 2017.

[9] A. Avron. Weyl Reexamined: "Das Kontinuum" 100 Years Later. *Bull. Symb. Log.*, 26(1):26–79, 2020.

[10] S. Awodey. Structuralism, Invariance and Univalence. *Philosophia Mathematica*, 22(1):1–11, 2014.

[11] S. Awodey. Univalence as a principle of logic. *Indagationes Mathematicae*, 29(6):1497–1510, 2018.

[12] S. Awodey and M. Warren. Homotopy theoretic models of identity types. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 146, pages 45–55. Cambridge University Press, 2009.

[13] A. Bauer, J. Gross, P. L. Lumsdaine, M. Shulman, M. Sozeau, and B. Spitters. The Hott Library: A Formalization of Homotopy Type Theory in Coq. In *Proceedings of the 6th ACM SIGPLAN Conference on Certified Programs and Proofs*, CPP 2017, page 164–172, New York, NY, USA, 2017. Association for Computing Machinery.

[14] J. L. Bell. *Toposes and local set theories: an introduction*. Courier Corporation, 2008.

[15] E. Bishop. *Foundations of constructive analysis*, volume 60. McGraw-Hill, 1967.

[16] E. Bishop. *Schizophrenia in contemporary mathematics*. American Mathematical Society, 1973.

[17] R. Bocquet. External univalence for second-order generalized algebraic theories, 2022.

[18] S. Boulier, P. Pédrot, and N. Tabareau. The next 700 syntactical models of type theory. In *CPP*, pages 182–194. ACM, 2017.

[19] L. Brouwer. De onbetrouwbaarheid der logische principes. *Tijdschrift voor wijsbegeerte*, 2:152–158, 1908.

[20] E. Casari. *Questioni di Filosofia Della Matematica*. Feltrinelli, 1964.

[21] A. Church. A Formulation of the Simple Theory of Types. *J. Symb. Log.*, 5(2):56–68, 1940.

[22] A. Church. Comparison of Russell's resolution of the semantical antinomies with that of Tarski. *The Journal of Symbolic Logic*, 41(4):747–760, 1976.

[23] P. Clairambault and P. Dybjer. The Biequivalence of Locally Cartesian Closed Categories and Martin-Löf Type Theories. In L. Ong, editor, *Typed Lambda Calculi and Applications*, pages 91–106, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[24] C. Cohen, T. Coquand, S. Huber, and A. Mörtberg. Cubical Type Theory: A Constructive Interpretation of the Univalence Axiom. *FLAP*, 4(10):3127–3170, 2017.

[25] T. Coquand. Canonicity and normalization for dependent type theory. *Theor. Comput. Sci.*, 777:184–191, 2019.

[26] T. Coquand, S. Huber, and A. Mörtberg. On higher inductive types in cubical type theory. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 255–264, 2018.

[27] T. Coquand and G. Huet. The Calculus of Constructions. *Information And Computation*, 76(2-3):95–120, 1988.

[28] T. Coquand and C. Paulin. Inductively defined types. In *Conference on Computer Logic*, volume 417 of *Lecture Notes in Computer Science*, pages 50–66. Springer, 1988.

[29] L. Crosilla. Predicativity and Feferman. In G. Jäger and W. Sieg, editors, *Feferman on Foundations: Logic, Mathematics, Philosophy*, pages 423–447. Springer International Publishing, Cham, 2017.

[30] L. Crosilla. The entanglement of logic and set theory, constructively. *Inquiry*, 65(6):638–659, 2022.

[31] L. Crosilla. Predicativity and constructive mathematics. In *Objects, Structures, and Logics*, pages 287–309. Springer, 2022.

[32] L. Crosilla. Bishop's Mathematics: A Philosophical Perspective. In *Handbook of Bishop Constructive Mathematics*. CUP, forthcoming.

[33] P. Curien. Substitution up to isomorphism. *Fundam. Informaticae*, 19(1/2):51–85, 1993.

[34] H. B. Curry. Functionality in combinatory logic. *Proceedings of the National Academy of Sciences of the United States of America*, 20(11):584, 1934.

[35] H. B. Curry and R. Feys. *Combinatory logic*, volume 1. North-Holland Amsterdam, 1958.

[36] N. G. De Bruijn. The mathematical language AUTOMATH, its usage, and some of its extensions. In *Symposium on automatic demonstration*, pages 29–61. Springer, 1970.

[37] W. Dean and S. Walsh. The prehistory of the subsystems of second-order arithmetic. *The Review of Symbolic Logic*, 10(2):357–396, 2017.

[38] R. Diaconescu. Axiom of choice and complementation. *Proceedings of the American Mathematical Society*, 51(1):176–178, 1975.

[39] M. Dummett. The philosophical basis of intuitionistic logic. In *Logic Colloquium '73*, volume 80 of *Studies in Logic and the Foundations of Mathematics*, pages 5 – 40. Elsevier, 1975.

[40] M. Dummett. *The logical basis of metaphysics*. Harvard university press, 1991.

[41] M. A. E. Dummett. *Truth and Other Enigmas*. Cambridge, MA, USA: Harvard University Press, 1978.

[42] P. Dybjer. Inductive families. *Formal aspects of computing*, 6(4):440–465, 1994.

[43] P. Dybjer. Internal type theory. In S. Berardi and M. Coppo, editors, *Types for Proofs and Programs*, pages 120–134, Berlin, Heidelberg, 1996. Springer Berlin Heidelberg.

[44] P. Dybjer. Representing Inductively Defined Sets by Wellorderings in Martin-Löf's type theory. *Theor. Comput. Sci.*, 176(1-2):329–335, 1997.

[45] S. Feferman. Systems of predicative analysis I. *The Journal of Symbolic Logic*, 29(1):1–30, 1964.

[46] S. Feferman. A language and axioms for explicit mathematics. In *Algebra and logic*, volume 450 of *Lecture Notes in Mathematics*, pages 87–139, Fourteenth Summer Res. Inst., Austral. Math. Soc., Monash Univ., Clayton, 1974, 1975. Springer, Berlin.

[47] S. Feferman. Iterated Inductive Fixed-Point Theories: Application to Hancock's Conjecture. In G. Metakides, editor, *Patras Logic Symposion*, volume 109 of *Studies in Logic and the Foundations of Mathematics*, pages 171–196. Elsevier, 1982.

[48] S. Feferman. Weyl Vindicated: Das Kontinuum, 70 Years Later. In *Proceedings of the Cesena Conference on Logic and Philosophy of Science*, 1988.

[49] S. Feferman. *In the light of logic*. Oxford University Press, 1998.

[50] S. Feferman. The significance of Weyl's Das Kontinuum. In *Proof Theory*, pages 179–194. Springer, 2000.

[51] S. Feferman. Is the continuum hypothesis a definite mathematical problem. *Draft of paper for the lecture to the Philosophy Dept., Harvard University*, pages 2011–2012, 2011.

[52] J. E. Fenstad (ed.). *Proceedings of the Second Scandinavian Logic Symposium*, volume 2. North-Holland Publishing Company, 1971.

[53] N. Gambino and R. Garner. The identity type weak factorisation system. *Theoretical computer science*, 409(1):94–109, 2008.

[54] R. Garner. On the strength of dependent products in the type theory of martin-löf. *Annals of Pure and Applied Logic*, 160(1):1–12, 2009.

[55] J.-Y. Girard. *Interprétation fonctionnelle et élimination des coupures de l'arithmétique d'ordre supérieur*. PhD thesis, Paris VII, 1972.

[56] V. K. Gödel. Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. *dialectica*, 12(3-4):280–287, 1958.

[57] A. Grzegorczyk. Elementarily definable analysis. *Fundamenta Mathematicae*, 41(2):311–338, 1955.

[58] R. Harper, F. Honsell, and G. D. Plotkin. A framework for defining logics. *J. ACM*, 40(1):143–184, 1993.

[59] A. Heyting. Die Formalen Regeln der intuitionistischen Logik I. *Sitzungsberichte der Preussischen Akademie der Wissenschaften*, pages 42–56, 1930.

[60] A. Heyting. Die Formalen Regeln der intuitionistischen Logik II. *Sitzungsberichte der Preussischen Akademie der Wissenschaften*, pages 57–71, 1930.

[61] A. Heyting. Die Formalen Regeln der intuitionistischen Logik III. *Sitzungsberichte der Preussischen Akademie der Wissenschaften*, pages 158–169, 1930.

[62] A. Heyting. Sur la logique intuitionniste. *Academie Royale de Belgique, Bulletin de la Classe des Sciences*, 16:957–963, 1930.

[63] A. Heyting. Die intuitionistische grundlegung der mathematik. *Erkenntnis*, 2:106–115, 1931.

[64] M. Hofmann. On the Interpretation of Type Theory in Locally Cartesian Closed Categories. In *CSL*, volume 933 of *Lecture Notes in Computer Science*, pages 427–441. Springer, 1994.

[65] M. Hofmann. Conservativity of Equality Reflection over Intensional Type Theory. In *TYPES*, volume 1158 of *Lecture Notes in Computer Science*, pages 153–164. Springer, 1995.

[66] M. Hofmann. *Extensional concepts in intensional type theory.* PhD thesis, University of Edinburgh., 1995.

[67] M. Hofmann and T. Streicher. The groupoid interpretation of type theory. *Twenty-five years of constructive type theory (Venice, 1995)*, 36:83–111, 1998.

[68] W. A. Howard. The Formulae-as-Types Notion of Construction. In H. Curry, H. B., S. J. Roger, and P. Jonathan, editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism.* Academic Press, 1980.

[69] J. Hugunin. Why Not W? In U. de'Liguoro, S. Berardi, and T. Altenkirch, editors, *26th International Conference on Types for Proofs and Programs, TYPES 2020, March 2-5, 2020, University of Turin, Italy*, volume 188 of *LIPIcs*, pages 8:1–8:9. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

[70] H. Ishihara, M. E. Maietti, S. Maschio, and T. Streicher. Consistency of the intensional level of the Minimalist Foundation with Church's thesis and axiom of choice. *Arch. Math. Log.*, 57(7-8):873–888, 2018.

[71] K. Kapulkin and P. L. Lumsdaine. The homotopy theory of type theories. *Advances in Mathematics*, 337:1–38, 2018.

[72] K. Kapulkin and P. L. Lumsdaine. Homotopical inverse diagrams in categories with attributes. *Journal of Pure and Applied Algebra*, 225(4):106563, 2021.

[73] K. Kapulkin and P. L. Lumsdaine. The simplicial model of Univalent Foundations (after Voevodsky). *Journal of the European Mathematical Society*, 23(6):2071–2126, 2021.

[74] S. C. Kleene. Hierarchies of number-theoretic predicates. *Bulletin of the American Mathematical Society*, 61(3):193–213, 1955.

[75] A. Klev. Eta-Rules in Martin-Löf Type Theory. *The Bulletin of Symbolic Logic*, 25(3):333–359, 2019.

[76] A. Kolmogorov. On the interpretation of intuitionistic logic. *Mathematische Zeitschrift*, 35:58–65, 1932.

[77] G. Kreisel. La prédicativité. *Bulletin de la Société mathématique de France*, 88:371–391, 1960.

[78] G. Kreisel. Ordinal Logics and the characterization of informal concepts of proof. In *Proc. of the International Congress of Mathematicians, Edinburgh, 14-21 August 1958*, pages 289–299. AMS, 1960.

[79] G. Kreisel. Foundations of intuitionistic logic. In *Studies in Logic and the Foundations of Mathematics*, volume 44, pages 198–210. Elsevier, 1966.

[80] G. Kreisel. Church's Thesis: A Kind of Reducibility Axiom for Constructive Mathematics. In A. Kino, J. Myhill, and R. Vesley, editors, *Intuitionism and Proof Theory: Proceedings of the Summer Conference at Buffalo N.Y. 1968*, volume 60 of *Studies in Logic and the Foundations of Mathematics*, pages 121–150. Elsevier, 1970.

[81] G. Kreisel. Perspectives in the philosophy of pure mathematics. In *Studies in Logic and the Foundations of Mathematics*, volume 74, pages 255–277. Elsevier, 1973.

[82] J. Lambek and P. J. Scott. *Introduction to Higher Order Categorical Logic*. Cambridge University Press, USA, 1986.

[83] G. Longo. Review of [48]. *Journal of Symbolic Logic*, 58(3):1085–1086, 1993.

[84] G. Longo. Reflections on Concrete Incompleteness. *Philosophia Mathematica*, 19(3):255–280, 2011.

[85] P. Lorenzen and J. Myhill. Constructive definition of certain analytic sets of numbers. *The Journal of Symbolic Logic*, 24(1):37–49, 1959.

[86] M. Maietti. On Choice Rules in Dependent Type Theory. In *TAMC*, volume 10185 of *Lecture Notes in Computer Science*, pages 12–23, 2017.

[87] M. E. Maietti. About effective quotients in constructive type theory. In *International Workshop on Types for Proofs and Programs*, pages 166–178. Springer, 1998.

[88] M. E. Maietti. Modular correspondence between dependent type theories and categories including pretopoi and topoi. *Mathematical Structures in Computer Science*, 15(6):1089–1149, 2005.

[89] M. E. Maietti. A minimalist two-level foundation for constructive mathematics. *Annals of Pure and Applied Logic*, 160(3):319 – 354, 2009. Computation and Logic in the Real World: CiE 2007.

[90] M. E. Maietti, S. Maschio, and M. Rathjen. Inductive and Coinductive Topological Generation with Church's thesis and the Axiom of Choice. *Log. Methods Comput. Sci.*, 18(4), 2022.

[91] M. E. Maietti and G. Rosolini. Quotient completion for the foundation of constructive mathematics. *Logica Universalis*, 7(3):371–402, 2013.

[92] M. E. Maietti and G. Sambin. Toward a minimalist foundation for constructive mathematics. In *From Sets and Types to Topology and Analysis: Practicable Foundations for Constructive Mathematics*, volume 48, pages 91–114. Oxford Logic Guides, 2005.

[93] M. Makkai. Towards a Categorical Foundation of Mathematics. In *Logic Colloquium*, volume 11 of *Lecture Notes in Logic*, pages 153–190. Springer, 1995.

[94] P. Martin-Löf. *Notes on Constructive Mathematics*. Stockholm, Almqvist & Wiksell, 1970.

[95] P. Martin-Löf. Hauptsatz for the intuitionistic theory of iterated inductive definitions. In *Studies in Logic and the Foundations of Mathematics*, volume 63, pages 179–216. Elsevier, 1971.

[96] P. Martin-Löf. A theory of types, 1971.

[97] P. Martin-Löf. An intuitionistic theory of types: Predicative part. In *Studies in Logic and the Foundations of Mathematics*, volume 80, pages 73–118. Elsevier, 1975.

[98] P. Martin-Löf. Constructive mathematics and computer programming. In *Studies in Logic and the Foundations of Mathematics*, volume 104, pages 153–175. Elsevier, 1982.

[99] P. Martin-Löf. *Intuitionistic type theory*, volume 9. Bibliopolis, 1984.

[100] P. Martin-Löf. On the meanings of the logical constants and the justifications of the logical laws. In *Proceedings of the conference on mathematical logic (Siena, 1983/1984)*, volume 2, pages 203–281, 1985. reprinted in: Nordic J. Philosophical Logic 1 (1996), no. 1, pages 11–60.

[101] P. Martin-Löf. The logic of judgements. In *Workshop on General Logic, Laboratory for Foundations of Computer Science*, pages 35–36, 1987.

[102] P. Martin-Löf. An intuitionistic theory of types. *Twenty-five years of constructive type theory*, 36:127–172, 1998.

[103] P. Martin-Löf. 100 years of Zermelo's Axiom of Choice: what was the problem with it? *Comput. J.*, 49(3):345–350, 2006.

[104] P. Martin-Löf. The Hilbert-Brouwer controversy resolved? In *One hundred years of intuitionism (1907–2007)*, pages 243–256. Springer, 2008.

[105] P. Martin-Löf. About Models for Intuitionistic Type Theories and the Notion of Definitional Equality. In *Proceedings of the Third Scandinavian Logic Symposium*, volume 82 of *Studies in Logic and the Foundations of Mathematics*, pages 81 – 109. Elsevier, 1975.

[106] S. Maschio and P. Sabelli. On the Compatibility Between the Minimalist Foundation and Constructive Set Theory. In *CiE*, volume 13359 of *Lecture Notes in Computer Science*, pages 172–185. Springer, 2022.

[107] I. Moerdijk and E. Palmgren. Wellfounded trees in categories. *Ann. Pure Appl. Log.*, 104(1-3):189–218, 2000.

[108] A. Mörtberg. Cubical methods in homotopy type theory and univalent foundations. *Mathematical Structures in Computer Science*, pages 1–38, 2021.

[109] A. Mostowski. On various degrees of constructivism. In A. Mostowski, editor, *Foundational Studies*, volume 93 of *Studies in Logic and the Foundations of Mathematics*, pages 359–375. Elsevier, 1979.

[110] J. Myhill. Constructive set theory. *The Journal of Symbolic Logic*, 40(3):347–382, 1975.

[111] J. Myhill and N. Goodman. Choice implies excluded middle. *Zeit Math Log*, 23:461, 1978.

[112] E. Nelson. *Predicative Arithmetic*, volume 32. Princeton University Press, 2014.

[113] B. Nordström, K. Petersson, and J. M. Smith. *Programming in Martin-Löf's type theory*, volume 200. Oxford University Press Oxford, 1990.

[114] N. Oury. Extensionality in the Calculus of Constructions. In *TPHOLs*, volume 3603 of *Lecture Notes in Computer Science*, pages 278–293. Springer, 2005.

[115] E. Palmgren. On universes in type theory. *Twenty five years of constructive type theory (Venice 1995*, pages 191–204, 1998.

[116] E. Palmgren. On Equality of Objects in Categories in Constructive Type Theory. In A. Abel, F. N. Forsberg, and A. Kaposi, editors, *23rd*

International Conference on Types for Proofs and Programs, TYPES 2017, May 29-June 1, 2017, Budapest, Hungary, volume 104 of *LIPIcs*, pages 7:1–7:7. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[117] E. Palmgren. A Constructive Examination of a Russell-Style Ramified Type Theory. *Bull. Symb. Log.*, 24(1):90–106, 2018.

[118] C. Parsons. The Impredicativity of Induction. In M. Detlefsen, editor, *Proof, Logic, and Formalization*, pages 139–161. Routledge, 1992.

[119] C. Paulin-Mohring. Inductive Definitions in the system Coq - Rules and Properties. In *TLCA*, volume 664 of *Lecture Notes in Computer Science*, pages 328–345. Springer, 1993.

[120] F. Pfenning. Logical Frameworks. In *Handbook of Automated Reasoning*, pages 1063–1147. Elsevier and MIT Press, 2001.

[121] H. Poincaré. Les mathématiques et la logique. *Revue de métaphysique et de morale*, 14(3):294–317, 1906.

[122] D. Prawitz. Hauptsatz for Higher Order Logic. *J. Symb. Log.*, 33(3):452–457, 1968.

[123] D. Prawitz. Ideas and results in proof theory. In *Studies in Logic and the Foundations of Mathematics*, volume 63, pages 235–307. Elsevier, 1971.

[124] D. Prawitz. Towards a foundation of a general proof theory. In P. Suppes, L. Henkin, A. Joja, and G. C. Moisil, editors, *Proceedings of the Fourth International Congress for Logic, Methodology and Philosophy of Science, Bucharest, 1971*, volume 74 of *Studies in Logic and the Foundations of Mathematics*, pages 225–250. Elsevier, 1973.

[125] D. Prawitz. On the Idea of a General Proof Theory. *Synthese*, 27(1-2):63–77, 1974.

[126] D. Prawitz. *Natural deduction: A proof-theoretical study*. Courier Dover Publications, 2006.

[127] W. V. O. Quine. Logic and the Reification of Universals. *From a logical point of view*, pages 102–129, 1953.

[128] W. V. O. Quine. *Set Theory and Its Logic*. Belknap Press of Harvard University Press, 1969.

[129] F. Ramsey. The Foundations of Mathematics. *Proceedings of the London Mathematical Society*, 2(1):338–384, 1926.

[130] M. Rathjen. The Constructive Hilbert Program and the Limits of Martin-Löf Type Theory. *Synthese*, 147(1):81–120, 2005.

[131] J. C. Reynolds. Types, Abstraction and Parametric Polymorphism. In R. E. A. Mason, editor, *Information Processing 83, Proceedings of the IFIP 9th World Computer Congress, Paris, France, September 19-23, 1983*, pages 513–523. North-Holland/IFIP, 1983.

[132] E. Rijke, M. Shulman, and B. Spitters. Modalities in homotopy type theory. *Log. Methods Comput. Sci.*, 16(1), 2020.

[133] B. Russell. Les paradoxes de la logique. *Revue de métaphysique et de morale*, 14(5):627–650, 1906.

[134] B. Russell. Mathematical logic as based on the theory of types. *American Journal of Mathematics*, 30(3):222–262, 1908.

[135] B. Russell and A. N. Whitehead. *Principia Mathematica Vol. I.* Cambridge University Press, 1910.

[136] G. Sambin. Intuitionistic formal spaces - a first communication. In *Mathematical logic and its Applications*, pages 187–204. Springer, 1987.

[137] P. Schroeder-Heister. A Natural Extension of Natural Deduction. *Journal of Symbolic Logic*, 49(4):1284–1300, 1984.

[138] P. Schroeder-Heister. The categorical and the hypothetical: a critique of some fundamental assumptions of standard semantics. *Synthese*, 187(3):925–942, 2012.

[139] K. Schütte. Predicative well-orderings. In *Studies in Logic and the Foundations of Mathematics*, volume 40, pages 280–303. Elsevier, 1965.

[140] D. Scott. Constructive validity. In *Symposium on automatic demonstration*, pages 237–275. Springer, 1970.

[141] R. A. Seely. Locally cartesian closed categories and type theory. In *Mathematical proceedings of the Cambridge philosophical society*, volume 95, pages 33–48. Cambridge University Press, 1984.

[142] A. Setzer. Extending Martin-löf type theory by one Mahlo-universe. *Archive for Mathematical Logic*, 39(3):155–181, 2000.

[143] M. Shulman. Univalence for inverse diagrams and homotopy canonicity. *Mathematical Structures in Computer Science*, 25(5):1203–1277, 2015.

[144] S. G. Simpson. *Subsystems of Second Order Arithmetic*, volume 1. Cambridge University Press, 2009.

[145] J. Sterling and C. Angiuli. Normalization for cubical type theory. In *LICS*, pages 1–15. IEEE, 2021.

[146] T. Streicher. Independence of the induction principle ad the axiom of choice in the pure calculus of constructions. *Theoretical computer science*, 103(2):395–408, 1992.

[147] T. Streicher. Investigations into intensional type theory. *Habilitiation Thesis, Ludwig Maximilian Universität*, 1993.

[148] T. Streicher. *Semantics of type theory: correctness, completeness and independence results*. Springer Science & Business Media, 2012.

[149] G. Sundholm. Constructions, proofs and the meaning of logical constants. *J. Philos. Log.*, 12(2):151–172, 1983.

[150] G. Sundholm. Proof theory and meaning. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic: Volume III: Alternatives in Classical Logic*, pages 471–506. Springer Netherlands, Dordrecht, 1986.

[151] W. W. Tait. A nonconstructive proof of Gentzen's Hauptsatz for second order predicate logic. *Bulletin of the American Mathematical Society*, 72(6):980–983, 1966.

[152] W. W. Tait. Intensional interpretations of functionals of finite type I. *The journal of symbolic logic*, 32(2):198–212, 1967.

[153] W. W. Tait. Constructive Reasoning. In B. Van Rootselaar and J. Staal, editors, *Logic, Methodology and Philosophy of Science III*, volume 52 of *Studies in Logic and the Foundations of Mathematics*, pages 185–199. Elsevier, 1968.

[154] M. Takahashi. A proof of cut-elimination theorem in simple type-theory. *Journal of the mathematical society of Japan*, 19(4):399–410, 1967.

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

[156] A. S. Troelstra. *Metamathematical investigation of intuitionistic arithmetic and analysis*, volume 344. Springer Science & Business Media, 1973.

[157] A. S. Troelstra. A note on non-extensional operations in connection with continuity and recursiveness. *Indagationes Mathematicae (Proceedings)*, 80(5):455–462, 1977.

[158] A. S. Troelstra and D. van Dalen. *Constructivism in Mathematics : An Introduction*. Studies in logic and the foundations of mathematics 121/123. North Holland, Amsterdam, 1988.

[159] V. Voevodsky. A very short note on the homotopy λ-calculus. *Unpublished note*, pages 10–27, 2006.

[160] P. Wadler. Theorems for free! In *FPCA*, pages 347–359. ACM, 1989.

[161] H. Wang. The formalization of mathematics. *The Journal of Symbolic Logic*, 19(4):241–266, 1954.

[162] H. Wang. Ordinal numbers and predicative set theory. *Mathematical Logic Quarterly*, 5(14-24):216–239, 1959.

[163] H. Weyl. Über die Definitionen der mathematischen Grundbegriffe. *Mathematisch-naturwissenschaftliche Blätter*, 7(1910), 1910.

[164] H. Weyl. *Das Kontinuum: Kritische Untersuchungen über die Grundlagen der Analysis*. Verlag von Veit & Comp., 1918.

[165] H. Weyl. *Raum, Zeit, Materie*. Springer-Verlag, 1918.

[166] H. Weyl. Mathematics and logic: A brief survey serving as preface to a review of The Philosophy of Bertrand Russell. *The American Mathematical Monthly*, 53(1):2–13, 1946.

[167] H. Weyl. Axiomatic versus Constructive Procedures in Mathematics. *The Mathematical Intelligencer*, 7(4):10–17, 1985.

[168] H. Weyl. *The Continuum: a critical examination of the foundation of analysis*. Courier Corporation, 1994.

[169] H. Weyl. *Levels of Infinity: Selected Writings on Mathematics and Philosophy*. Courier Corporation, 2013.

[170] H. Weyl. *Philosophy of mathematics and natural science*. Princeton University Press, 2021.

[171] T. Winterhalter, M. Sozeau, and N. Tabareau. Eliminating reflection from type theory. In *CPP*, pages 91–103. ACM, 2019.