

A contextual type theory with judgemental modalities for reasoning from open assumptions

Giuseppe Primiero*
FWO – Centre for Logic and Philosophy of Science
Ghent University (Belgium)

June 6, 2011

Abstract

Contextual type theories are largely explored in their applications to programming languages, but less investigated for knowledge representation purposes. The combination of a constructive language with a modal extension of contexts appears crucial to explore the attractive idea of a type-theoretical calculus of provability from refutable assumptions for non-monotonic reasoning. This paper introduces such a language: the modal operators are meant to internalize two different modes of correctness, respectively with necessity as the standard notion of constructive verification and possibility as provability up to refutation of contextual conditions.

1 Introduction

In the landscape of non-classical logics, constructive formal systems use proofs as first-class citizens to define the notion of truth, generalized to truth valid under assumptions. The idea of contextual truth, that originated with sequent calculi, is well interpreted for provability in the type-theoretical languages based on intuitionistic logic, such as in Martin-Löf Type Theory.¹

In such a system, expressions have judgemental form A *true* with propositional content A , the latter being justified by an appropriate proof term $a : A$. The corresponding notion of contextual truth allows formulas of the form $\Gamma \vdash a : A$, where Γ is of the standard form $[x_1 : A_1, \dots, x_n : A_n]$, a being a proof of A whenever appropriate substitutions are performed on the variables in Γ , so that $[x_1/a_1 : A_1, \dots, x_n/a_n : A_n] \vdash a : A$ holds. This means that, constructively, hypothetical truth is reduced to dependent closed constructions and that hypotheses are grounded on the primitive notion of premise. In Martin-Löf's

*Fellow of the FWO - Research Foundation Flanders.

¹See [22], [23], [25].

Type Theory, this induces the conceptual distinction between a known judgement and a judgement-candidate.² The notion of an assumption is obtained by abstraction: from a construction of a proposition, one abstracts to obtain an assumption used in an implicational relation, whose computational content can be reconstructed when needed (as its content has been known). This construction is reflected in the usual introduction rule for implication in Natural Deduction calculi. Different research fields can be traced back to similar principles grounded around the notion of context. One of the first was the modeling of contexts from AI, which eventually led to the propositional logic and first-order logic of context.³ Further research with applications in linguistics and hardware verification has the very same starting point.⁴

The ability to speak of contexts via a modal extension represents the next obvious step. Along with the standard accessibility relations for modal operators in the intuitionistic translation of K and the constructive version of $S4$,⁵ a weaker format to accommodate the notion of context is given by the calculus CK in [27]. The latter provides a possible-world semantics sound and complete with respect to the natural deduction interpretation given in [14]. The same kind of issues led recently to the formulation of contextual modal type theories in [30] and [29]. In particular, the formal language presented in [30] exploits constructions for both modalities from the same principle of contextual derivability: the possibility judgement ($\diamond A \text{ true}$) (proposition ‘possible A ’ is true) is obtained from a contextually valid proposition; the necessity judgement ($\square A \text{ true}$) (proposition ‘necessarily A ’ is true) internalizes validity by satisfying assumptions, mimicking the Necessitation Rule from the semantics for a normal modal logic.

In the present paper, we deviate from the propositional approach: our formulas will be respectively of the form $\square(A \text{ true})$ (necessarily, proposition A is true) and $\diamond(A \text{ true})$ (possibly, proposition A is true). Our focus is on an interpretation of the modalities as meta-operators to express contextual validity. We shall understand the necessity judgement as saying that the assertion conditions for the related proposition are satisfied. The possibility judgement refers to a proposition whose assertion conditions are admissible, but whose construction is not guaranteed. We shall call these judgements ‘open assumptions’.

Our starting point is the constructive reading of the notion of truth as existence of a verification, i.e. a notion of *truth by verification*; this is extended by a semantic format for the epistemic notion of verification under open assumptions. Technically, this amounts to a version of the type-theoretical language that does not satisfy explicit substitutions on variables for assumptions,⁶ in this way expressing a notion of *truth up to refutability*. This distinction recalls a sensible topic for constructive logics. A standard explanation of constructive

²See e.g. [23], [40]. The term ‘judgement-candidate’ is originally due to Göran Sundholm.

³See [26], [6], [5].

⁴See e.g. the bibliography in [1].

⁵For this see [31], [41], [3], [2], [1]; see [37] for an overview of the early studies on intuitionistic modal logics.

⁶See e.g. [38].

refutation is based on the meaning of intuitionistic negation as the condition that there is no construction for an absurdity. On this basis, indirect proofs in the form of a *reductio ad absurdum* are standardly not admitted, whereas the usual intuitionistic absurdity rule interprets the classical *ex falso quodlibet*.⁷ The foundational work [20] represents the basic result of translation of classical mathematics into intuitionistic mathematics: Kolmogorov reduces classical formulas to intuitionistic ones as long as they are double-negated, what he calls ‘pseudo-truths’, the implication from $\neg\neg A$ to A being valid in the domain of the finitary. The present paper dwells on this foundational idea that truth is admissible for a content which cannot be refuted, provided this is valid over a finitary domain. We provide distinct constructors for ‘pseudo-truths’ (to keep on using Kolmogorov’s terminology) and for constructive truths. Formally, this requires constraining a part of the language to a finitary set of formulas with direct verification processes: in this fragment of the language, the meaning of a valid judgement ‘ A true’ justifies the further conclusion that no construction for ‘ $\neg A$ true’ is possible. We then extend the language by introducing the weaker notion of ‘truth admissible up to refutation’: this is defined by a constructor obtained as a double negation introduction from the previously given set of constructors, representing an appropriate formulation for a constructive notion of admissible or not-yet-refuted truth-candidate. The related constructions do not need to satisfy any corresponding negation conditions, as it is the case for the notion of constructive refutation introduced in [21].⁸ We only require that the notion of admissible truth-candidate literally satisfies the logical concept of an assumption, a term which might not have an appropriate β -reduction (as from the corresponding λ -calculus).⁹ By interpreting hypotheses as open constructions, a judgement $\Gamma \vdash a : A$ expresses the truth of the proposition A on the basis of the information provided in Γ , unless some of the judgements formulated in Γ is refuted. The modal extension solves the problem of expressing such an epistemic relation on contextual constructions in the language, giving a set of rules by which this notion is preserved under logical inference. In this way we are allowed to survey our epistemic stand towards a finite amount of logical information and claim our actual stand towards the possible extensions.

The structure of the paper is as follows. Section 2 provides a variant interpretation of the basic system of constructive type-theory, where the link between hypotheses and refutable contents becomes admissible; in section 2.2 this language is extended by introducing epistemic modal operators defined by

⁷See e.g. [39, p. 40].

⁸The conditions required in [21] are: (I) a construction c proves $\neg A$ if and only if c refutes A ; (II) it is decidable whether or not c proves A , and whether or not c refutes A , whereas it is not explicitly excluded that a formula may be proved by one construction and refuted by another; (III) a construction c refutes $\neg A$ if and only if c proves A . The combination of points (I) and (III) represents a direct translation of constructive proof into refutation.

⁹Notoriously, the Curry-Howard isomorphism which matches formulas in a Hilbert-style system and types in calculi of combinatorial logic, has its further step in the analogy between natural deduction derivations and the terms of a λ -calculus. This latter analogy was established by Martin-Löf in a paper titled *Infinite terms and a system of natural deduction* in 1969. I owe this information to Göran Sundholm.

their judgemental scope; finally in section 3 the set of rules for such modal type-theory is formulated in order to preserve refutability under a consequence relation. In the conclusion, the next steps of this research are mentioned.

2 Interpreting proved and refutable contents

In the description of realistic knowledge processes, it seems appropriate to explain hypotheses as contents whose truth is declared, but whose refutation is not ruled out. Whereas a standard constructive reading of a hypothetical judgement is of the kind

(I know that) S is P , provided (I know that) A_1 to A_n hold,

we refer to the representation of knowledge states related to the following schema of sentential contents:

(I know that) S is P , provided that A_1 to A_n are not refuted

where S, P are terms and A_i are all propositions. Contextual reasoning allows us to implement this meaning of open assumptions as possible truth. To this aim, formally distinct notions of global and local validity can be used to interpret truth in context via the underlying reading of assumptions along with constructive truth. This is required because in the basic constructive definition of truth, refutable contents are not discussed. Hence, if the proof of a construction $[x_1 : A_1, \dots, x_n : A_n] \vdash a : A$ is a process admitting the proof of A under not yet verified assumptions, the truth of propositions A_1, \dots, A_n needs to be introduced in a non-constructive way.

In order to formalize such a reading, one needs to keep introduction rules for *proven* and *assumed* propositional contents separate, i.e. where the latter are not ultimately justified on the basis of the former. Judgemental modalities will be used to express the resulting different *modes of correctness* of a propositional content derivable in the context of either proven or assumed propositions:

- a judgement $\Box(A \text{ true})$ expresses that a content A is true in any epistemic state, as A is independent from any refutable condition (either there are none, or all of them have been secured);
- a judgement $\Diamond(A \text{ true})$ expresses that a content A is true in some epistemic states, namely where certain conditions are not refuted.

To obtain a modal language where the meaning of the operators is explained in this way, we start from a polymorphic language containing one basic sort *type* for categorical (non assumptions-based) constructive judgements with corresponding term constructors a, b ; and one sort *type_{inf}* (*information type*) for judgements in a context of refutable conditions, with corresponding variable

constructors x_1, x_2 .¹⁰ Judgements of the first sort (*type*) induce a constructive notion of truth (*true*), the second ones (*type_{inf}*) a weaker predicate of truth up to verification (*true**): type objects are meaning objects, each related to a corresponding semantic predicate.

Our syntax, justified in the following two subsections and extended to the modal formulas only in the next section, is the following:

Types := *type*; *type_{inf}*;
 Propositions := A ; $A \wedge B$; $A \vee B$; $A \rightarrow B$; $(\exists a_i : A_i)B$; $(\forall a_i : A_i)B$; $A \supset B$;
 $A \rightarrow \perp$
 Proof terms := a ; (a, b) ; $a(b)$; $\lambda(a(b))$; $\langle a, b \rangle$;
 Proof variables := x ; $(x(b))$; $(x(b))(a)$;
 Contexts := $\Gamma, x : A$; $\Gamma, a : A$; $\Box\Gamma$; $\Diamond\Gamma$;
 Judgments := $a : A$; $\neg(A \rightarrow \perp)$; $x : A$; $A \text{ true}$; $A \text{ true}^*$; $\Gamma \vdash A \text{ true}$; $\Diamond(A \text{ true})$;
 $\Box(A \text{ true})$.

The basic novelty of this syntax is represented by the introduction of the non-standard kind *type_{inf}* and the corresponding semantic notion *true**: they will provide us with the required syntactic-semantic weakening appropriate for the notion of refutable assumption. It is on the basis of such extension that the modal fragment will be introduced in the next section.

2.1 The non-modal fragment

The two objects *type* and *type_{inf}* are the linguistic/ontological categories of our system. Type Formation is no longer a unique judgement (as it is the case with Martin-Löf's Type Theory), because it now includes two distinct cases. Absolute judgements in our type theory are of the form $a : A$ and $\neg(A \rightarrow \perp)$, the latter generating an assumption judgement $x : A$. From these two judgements we define both the types and the semantic terms of our language.

Type formation and the verificationist principle of truth for categorically justified propositions are our basic inferences:

$$\frac{a : A}{A \text{ type}} \quad \text{Type formation} \quad \frac{a : A}{A \text{ true}} \quad \text{Truth Definition}$$

The first rule says: given a categorical construction a for proposition A , A is of the sort *type*. The second rule says: given a categorical construction a for proposition A , the sort A is categorically true. Provided they are justified in the same way, in the following – and especially in the elimination rules for

¹⁰The reason to call refutable types *Information Type* is dictated by the underlying epistemic difference between functional information and knowledge, where the former is defined as meaningful data, whereas the latter by meaningful justified (and hence true) data. Given both this conceptual distinction and the formal rules to follow, expressions in *type* can be seen as a proper subset of the *information type*.

connectives – we will take the liberty of using a judgement of the form $A \text{ true}$ in rules that usually require the sort A to be equipped with term a , as $A \text{ true}$ always presupposes $a : A$. For *type* the standard identity rules that define Reflexivity, Symmetry and Transitivity hold as usual. The set of judgements in *type* are the (*categorically*) *verifiable formulas* of the language. Constructors for these judgements are composed by way of listing, application, abstraction and pairing to define connectives and quantifiers: $\wedge, \vee, \rightarrow, \forall, \exists$.

$$\begin{array}{c}
\frac{a:A \quad b:B}{(a,b):A \wedge B \text{ true}} I\wedge \quad \frac{A \wedge B \text{ true}}{l(a):A} E\wedge \quad \frac{A \wedge B \text{ true}}{r(b):B} E\wedge \\
\\
\frac{a:A}{l(a):A \vee B \text{ true}} \text{Left IV} \quad \frac{b:B}{r(b):A \vee B \text{ true}} \text{Right IV} \\
\\
\frac{A \vee B \text{ true} \quad A \rightarrow C \text{ true} \quad B \rightarrow C \text{ true}}{C \text{ true}} E\vee \\
\\
\frac{a:A \quad A \text{ true} \vdash b:B}{a(b):A \rightarrow B \text{ true}} I\rightarrow \quad \frac{A \rightarrow B \text{ true} \quad a:A}{(a)b:B} E\rightarrow \\
\\
\frac{a_1:A_1, \dots, a_n:A_n \quad A_i \text{ true} \rightarrow b:B \quad \lambda((a_i(b))A, B)}{(\forall a_i:A_i)B \text{ type}} I\forall \\
\\
\frac{(\forall a_i:A_i)B \text{ type} \quad (a_i(b))A, B}{A_i \text{ true} \rightarrow b:B} E\forall \\
\\
\frac{A_i \text{ true} \rightarrow b:B \quad (\langle a_i, b \rangle, A, B)}{(\exists a_i:A_i)B \text{ type}} I\exists \\
\\
\frac{(\exists a_i:A_i)B \text{ type} \quad (\langle a_i, b \rangle, A, B)}{A_i \text{ true} \rightarrow b:B} E\exists \\
\\
\frac{a:A}{\neg A \rightarrow \perp} I\perp
\end{array}$$

We omit identity rules on constructors, we use the λ -operator as a \forall -constructor on terms combined by application, angled parentheses \langle, \rangle for ordered pairs as an \exists -constructor. \rightarrow is obtained by application $a(b)$ of the construction a of the antecedent to the construction b of the consequent, rather than by abstraction: it can be seen as a λ -term presented *together with* one of its α -terms.¹¹ This construction reduces all implicational relations to categorical terms and it validates no implication from the false.

¹¹It interprets strict implication á la Heyting, recalling ideas mentioned by Martin-Löf and the calculus of types with explicit substitutions presented in [38].

Quantifiers are formulated accordingly. Universal predication abstracts from enumerable sets of equivalent constructions of A_i all implying the same proposition B . Its elimination picks one application out of those constructions. Existential quantification is justified by paired constructions, i.e. from a constructor of B which can be obtained from any of the equivalent constructors of A_i . Its elimination picks out the constructor of A_i to reconstruct the implication.

The negation introduction rule is obviously derivable in the intuitionistic setting and it is crucial for the following extension to functional expressions: \neg occurs in a (negated) implication from a valid type to the absurdum, saying that if A *true* is a known judgement, then one infers that no construction for $\neg A$ holds. The corresponding elimination rule would validate double-negation elimination, but we formulate instead a non-standard extension to functional expressions by a connective \supset . Formally, a functional relation among expressions is explained as follows: if A *type* holds, then a construction of a new type B is possible by considering the latter as a family of sets over some $x : A$ such that B *type* $[x : A]$ whenever the substitution $[x/a]$ is performed. The type checking will require first A to be well-formed, secondly evaluation to a current environment (i.e. the variables' appropriate typing) for extraction of variable terms, thirdly construction for the variable in that environment, and finally evaluation of the variable and the formulation of the binding expression to a value for that environment. The generalization to multiple dependence being allowed, terms for B *type* whenever $[x_1 : A_1, \dots, x_n : A_n]$ are evaluated by being put into normal forms (eventually: weak head normal forms, explicit substitutions, closures) in order for the predication B *type* to be valid.

A new task is to admit no explicit substitution on such formulas, requiring that variables be well-typed without requiring that they be brought to standard normal form. This leads to the introduction of *assumed truth* in the sense of (*still*) *refutable contents*.

We start by allowing the new type format $type_{inf}$. A judgement A $type_{inf}$ is not given by direct construction, rather it is given by checking that no construction for $\neg A$ *type* is already given. In this way, an admissible (but not-yet-grounded) predication is performed; the resulting semantic judgement is the one for hypothetical truth ($true^*$):¹²

$$\frac{\neg(A \rightarrow \perp)}{A \text{ type}_{inf}} \quad \text{Informational Type formation}$$

$$\frac{A \text{ type}_{inf} \quad x : A}{A \text{ true}^*} \quad \text{Hypothetical Truth Definition}$$

¹²It is intuitive to understand the rule formation for A $type_{inf}$ as neutralized by a construction for $\neg A$, i.e. as soon as a refutation of A is given. The epistemic dynamics naturally involved by this rule can be described as the free act of the knowing agent to invoke A 'as long as it is not refuted'. As mentioned in a previous footnote, the reason to call this an Information Type is due to the distinction we draw between judgements grounding knowledge, and judgements providing information used to build (hypothetical) knowledge.

The judgement $\neg(A \rightarrow \perp)$ in the previous fragment of the language says that there exists no pair of constructions $\langle a, b \rangle$ such that $a(b) : A \text{ true} \rightarrow \perp$. It does not imply that $a : A$ follows: the latter justification is kept entirely constructive and therefore cannot be given by indirect proof. The type_{inf} formation rule reminds us of a double negation elimination and it recalls an introduction rule, but it is not the appropriate counterpart of its classical version. This can be only seen in terms of the implicational relation \supset that holds for type_{inf} and that shall be introduced below.

The second rule says that provided A can be admitted as a type_{inf} , a weak truth-predicate true^* (true up to refutation) is inferred by assuming a construction for A exists: it can be seen as a place-holder for admissible but strictly yet-ungrounded truth.¹³ As open terms of this form lack direct computational content (i.e. their constructor is only implicit but not evaluated), intensional identity cannot be defined over type_{inf} and only extensional identity as equality of the negated originating type is defined. For this reason, only conversions are admitted, so that type-checking and well-formedness of types becomes undecidable in view of the proposed extension with type_{inf} ; this also means that the introduction rule for the new sort type_{inf} substitutes the usual η -expansion.¹⁴

Under this interpretation, we introduce within the type_{inf} fragment:

$$\frac{A \text{ type}_{inf} \quad x : A \vdash B \text{ type}_{inf}}{x : A \vdash B \text{ true}^*} \text{ Function Construction}$$

¹³This interpretation represents a more epistemically oriented reading of similar uses of open terms, notoriously relevant in, for instance, partial evaluation, see [19], [18].

¹⁴This is clearly a different notion of extensional type than what is usually intended for Martin-Löf's Type Theory, which nonetheless leads to an equivalent result of general undecidability. Our language seems inappropriate to define typically extensional concepts such as pointwise equal functions and quotient types. Our main aim is instead to preserve as much as possible a constructive model and then adapt it to forms of natural reasoning. Given the nature of the project, what is more worrying for us is the impossibility of defining, in principle, equality of proofs and identifying equivalent propositions that are not reduced to the type fragment. This suggests that, under this interpretation, forms of reasoning may remain incomparable when starting with distinct – even though equivalent – refutable assumptions. Provided the general incomputability of the type_{inf} extension, peculiar consequences of the extensional version of Type Theory, such as the refutation of Church's Thesis, become less surprising. Nonetheless, it is maybe useful to notice that for every term in type_{inf} that can be correctly instantiated, there will be a corresponding term in type for which standard intensional and extensional identity can be defined; similarly, for any such term that cannot be instantiated, there will be the corresponding neutralizing refutation in type , which again satisfies equality and identity. The philosophical literature questioning the axiomatic view on mathematical proofs and their interpretation as mechanically checkable derivations is growing. In this direction, there is a large convergence on understanding the real process of proving mathematical statements as a problem-solving task that makes a crucial use of hypotheses, supported by a mixture of deductive moves and induction. See e.g. [36], [7], [17]. The present work offers some formal means of representing assumptions-based reasoning, by enriching the formal structure of constructive proofs with a weaker format of admissible truth. Nonetheless, from a purely formal viewpoint, the ultimate understanding of validity for our system relies on the appropriate reduction to the standard fragment of Constructive Type Theory with the type sort only, hence to a standard identity of proofs and executable (mechanically checkable) programs.

$$\frac{A \text{ type}_{inf} \quad x:A \vdash B \text{ true}^*}{((x)b) : A \supset B \text{ true}} \text{I}\supset \frac{(A \supset B) \text{ true} \quad A \text{ type}[x/a]}{B \text{ true}} \text{E}\supset$$

$$\frac{A \text{ type}_{inf} \quad x:A \vdash B \text{ type}_{inf} \quad a:A}{(x(b))(a) = b[a/x] : B \text{ type}[a/x]} \beta\text{-conversion}$$

Function construction says that B is true up to A being refuted (i.e. the formulation of a construction of $\neg A$ being provided). The new implicational connective \supset induces the standard dependent functional construction by abstraction. Its classical-like behavior is preserved by the introduction rule, where the antecedent is formulated in the first premise by an instance of the Information Type Formation rule and thus in principle by a double negation introduction. But the corresponding elimination rule explicitly requires a substitution of the double negated type $x : A$ with a term a , so preventing that it collapses into classical implication (namely by avoiding its holding with a false antecedent). β -conversion provides the appropriate translation to standard dependent type formation by application, expressing reducibility of type_{inf} to type .

The restriction imposed by $\text{E}\supset$ and explicitly formulated by the β -conversion rule, prevents the system from collapsing into classical logic by ensuring that $A \supset B \text{ true}$ whenever $A \rightarrow B \text{ true}$, and allowing that $A \supset B \text{ true}$ without $A \rightarrow B \text{ true}$, so invalidating the equivalence between the two implicational relations. In [8] it is shown how the equivalence of a classical-like and an intuitionistic-like implications can be produced by unrestrictedly accepting one of the schemas $A \text{ true} \rightarrow (B \rightarrow A \text{ true})$ or $A \text{ true} \supset (B \rightarrow A \text{ true})$.¹⁵ The collapse is then justified as follows:

1. $(A \rightarrow B \text{ true}) \supset (A \supset B \text{ true})$
2. $(A \supset B \text{ true}) \supset (A \rightarrow B \text{ true})$
3. $(A \supset B \text{ true}) \supset (A \text{ true} \rightarrow (A \supset B \text{ true}))$
4. $A \rightarrow B$
5. $(A \rightarrow B) \text{ iff } (A \supset B)$.¹⁶

In particular step 2. is obtained from step 1. by using an unrestricted version of $A \text{ true} \supset (B \rightarrow A \text{ true})$. As A occurs in this schema as the antecedent of \supset and the consequent of \rightarrow , obviously by $\text{E}\supset$ the restriction holds that $A \text{ type}$.¹⁷

In the following subsection we will present the extension to epistemic modalities derived from the introduced type constructors expressing the validity of truth over contextual extensions.

¹⁵Here and below, the notation of [8] is abandoned in view of an adaptation to ours.

¹⁶As this is supposed to express the collapse of the two implication relations, the equivalence is metatheoretical and can be expressed in either language.

¹⁷In [8] the corresponding restriction on the axiom schema is that for A every occurrence of classical implication, equivalence or negation be in the scope of an intuitionistic negation or implication, i.e. that $A \text{ true} \supset (B \rightarrow A \text{ true})$ holds if A is so called *persistent*.

2.2 Modalities for provability up to refutation

The distinction between the notions of ‘truth by verification’ and ‘verifiability-up-to-refutation’ can be internalized by the use of epistemic modalities as operators that apply to judgements of the form $A \text{ true}$.

In the preceding section, strictly constructive truth is limited to the epistemic protocol by which a verification can be formulated analytically,¹⁸ i.e. *without* assumptions. This explanation of the truth of a proposition supports the identity between the judgements “ $A \text{ true}$ ” and “ A has a verification in an empty context of assumptions”:

$$A \text{ true} \Leftrightarrow \emptyset \vdash a : A$$

where (\emptyset) describes the epistemic situation in which no condition for a is needed, as such external conditions either do not exist, or they have all been satisfied. Provided categorical justifications have epistemic priority over dependent ones,¹⁹ if $A \text{ true}$ holds, it also holds under refutable data being added, because by definition no declaration $\neg A \text{ type}_{inf}$ will be allowed if $A \text{ type}$ holds and thus $a : A$ is formulated. This will make A verified in *any* extension of the empty context:²⁰

$$A \text{ true} \Leftrightarrow \emptyset \vdash A \text{ true} \Leftrightarrow \Box(A \text{ true}).$$

The following obvious step is to relate dependent truth with a possibility form of judgement. The judgement “it is possible that proposition A is true”, should mean that only in *some* context the truth of A can be stated: the context Γ in which $A \text{ true}$ holds will contain all the conditions that satisfy $a : A$, and will be preserved by any other context in which these conditions are not refuted:

$$A \text{ true}^* \Leftrightarrow \Gamma \vdash A \text{ true} \Leftrightarrow \Diamond(A \text{ true})$$

where Γ contains propositions of the form $A_i \text{ true}^*$.²¹ Where β -conversion applies, there is an immediate reduction to the previous case of $A \text{ true}$ and the necessity judgement.

The use of modal judgements is meant to internalize the distinction among categorical and hypothetical (refutable) truth. The generalization to hypothetical reasoning allows us to clarify the notion of assumption. We shall rely on the different introduction rules for the semantic judgements $A \text{ true}$ and $A \text{ true}^*$.

¹⁸This notion of analytic judgement is introduced by Martin-Löf in [24].

¹⁹Truth by verification has epistemic priority on provability up to refutation, whereas dependent constructions with satisfied conditions (i.e. with valid β -conversions) are conceptually prior (because more general) to categorical constructions.

²⁰Judgmental necessity satisfies the correlation between validity and justification under no condition, as for the system presented in [30]; it is nonetheless justified in a completely different way, as the judgement “it is necessary that proposition A is true” is based on the analysis of its assertion conditions, by explaining necessity as validity against any possible state that contains refutable data for the construction of A .

²¹For more on the philosophical justification of this notion of judgemental modalities, see [33].

In the following section the full system for a modal language of proven and refutable contents is formulated.

3 Contextual Modal Type Theory for verification and refutation

The system for a modal type theory that includes a validity relation up to refutation is formulated by using the two distinct truth predicates, with propositional variables closed under logical connectives, proof terms and proof variables with distinct operations of application and abstraction depending on the required semantic specification. Terms a, b, \dots and variables x, y, \dots are respectively proof terms and place-holders for admissible proofs; types A, B, \dots are propositions; for dependent judgements we use the given explanation of a judgement A *true* holding under open assumptions $x_1 : A_1, \dots, x_n : A_n$ collected in context on the left-hand side of a \vdash derivability sign, so that from now on our standard judgement is of the form $\Gamma \vdash (A \text{ true})$, with its modal variants; we suppose that all variables in a context are distinct. The modules of verified and refutable contents are introduced in terms, respectively, of a premise and a hypothesis rule:

$$\frac{}{\Gamma, a : A, \Delta \vdash A \text{ true}} \text{Premise Rule} \quad \frac{}{\Gamma, x : A, \Delta \vdash A \text{ true}^*} \text{Hypothesis Rule}$$

The premise rule introduces explicitly verified contents; the hypothesis rule reflects the introduction of contents that are only assumed to be true; both rules can have $\Gamma, \Delta = \{\emptyset\}$. Correspondingly, the *true* predicate can be understood as validity (that is truth in every situation) and it corresponds to truth by verification; the predicate *true*^{*} corresponds to validity in a context of assumptions, or local validity. Standard logical connectives apply on construction-assigned formula, with the extension to hypothetical reasoning (\supset). The identity between categorical judgement and judgement valid under no context allows the internalization of the modal operator of necessity at judgemental level, whereas the hypothesis judgement works in the same way for the internalization of the possibility operator:²²

$$\frac{a : A}{\Box(A \text{ true})} \Box\text{-Formation} \quad \frac{x : A}{\Diamond(A \text{ true})} \Diamond\text{-Formation}$$

We can extend categorical verifications to contextual ones by allowing the use of formulas $a : A$ in context, so that $a : A \vdash a : A$ is a valid construction, but this

²²In line with the general philosophical characterization of this work, the formation rules for modalities are intended as applying to any sort of proposition that can be derived as a theorem within the language; they thus have a logical characterization, but not necessarily a strict mathematical one.

remains in the scope of our categorical module of the language, corresponding to an identity function. Truth formation allows the construction $a : A \vdash A \text{ true}$, but we shall not allow the derivation from $x : A$ to $A \text{ true}$, rather restrict the inference to truth only where verified (valid) assumptions are used. To this aim, modalities are extended to contextual judgement. We shall refer to $\Box\Gamma$ as the necessitation of the context Γ , that is a context containing only valid assumptions; we call these *premises*:

Definition 1 (Necessitation Context). *For any context Γ , $\Box\Gamma$ is given by $\bigcup\{\Box A \text{ true} \mid \text{for all } A \in \Gamma\}$.*

A judgement valid under assumptions becomes a necessary judgement under necessitation of its context of assumptions. Expressions in contexts that are not explicitly verified preserve the notion of refutable truths. We refer now to a context containing assumptions of the form $x : A$ as follows:

Definition 2 (Normal Context). *For any context Γ , $\Diamond\Gamma$ is given by $\bigcup\{\circ A \text{ true} \mid \circ = \{\Box, \Diamond\} \text{ and } \Diamond A \text{ true for at least one } A \in \Gamma\}$.*

A judgement valid under assumptions becomes a possibility judgement if its context remains normal, that is at least one of its propositional contents is *true**.²³

Because of the distinction between justified and refutable contents, the introduction of judgemental \Box is allowed under the verification of judgements in the related context; its elimination rule induces a valid proposition:

$$\frac{\Gamma \vdash A \text{ true}}{\Box\Gamma \vdash \Box(A \text{ true})} \text{I}\Box \quad \frac{\Box\Gamma \vdash \Box(A \text{ true}) \quad \Delta, a : A \vdash B \text{ true}}{\Gamma, \Delta \vdash B \text{ true}} \text{E}\Box$$

where $\Box\Gamma$ iff $[x_i/a_i] : A_i, \forall A_i \in \Gamma$, as by Definition 1.²⁴

To express the relation of truth instantiated by an hypothesis $x : A$, we refer to validity in a context by introduction and elimination rules for the \Diamond -operator:

$$\frac{\Gamma, x : A \vdash B \text{ true}^*}{\Box\Gamma, \Diamond(A \text{ true}) \vdash \Diamond(B \text{ true})} \text{I}\Diamond$$

²³In various literature in modal logic, *Necessitation* and *Normal Context* are usually called *Global* and *Local Context*. This distinction, however it is called, is crucial for preserving the problem of derivability under assumption in modal languages. I have strengthened here the reasoning, by obtaining modal judgements (rather than formulas) from the preservation/verification of assumptions. Cf. [15].

²⁴This rule for the necessity operator is similar the one introduced in [30], where $\Box A$ is derived by a valid A , hence this validates $\Delta; \vdash A \text{ true} \Rightarrow \Delta; \Gamma \vdash \Box A \text{ true}$ and requires an additional assumption $A \text{ valid}$ in the corresponding elimination. We express the validity by the necessitation context (which implies its extension to any other context by any $B \text{ type}_{inf}$ judgement) and analytically formulate it in the elimination rule, by adding the additional premise $a : A$. In the comparison with the system presented in [14], the obvious similarity is that the therein contained modality \Box_k satisfies the same principle of our $\text{I}\Box$, namely it builds-in the substitutions needed for formulas in contexts. On the other hand, the propositional format does not require any \Diamond operator, its role being syntactically satisfied by standard contexts.

$$\frac{\Box\Gamma, \Diamond\Delta \vdash \Diamond(A \text{ true}) \quad \Delta, x:A \vdash B \text{ true}^*}{\Gamma, \Delta \vdash B \text{ true}^*} E\Diamond$$

where the introduction rule shows the dependency of possible contents from refutable contents, whereas the corresponding elimination expresses the use of this information to infer further possible knowledge under the condition expressed by Definition 2.²⁵

Local soundness and completeness of our modal rules can be proved in the usual way by local reductions and expansions. Soundness is obtained by local reduction on $\Box(A \text{ true})$:

$$\frac{\frac{D_1}{\frac{\Gamma \vdash A \text{ true}}{\Box\Gamma \vdash \Box(A \text{ true})} \Box I} \quad \frac{E}{\Delta, a:A \vdash b:B} \Box E}{\Gamma, \Delta \vdash B \text{ true}} \Box E \quad \Rightarrow_{Redex} \quad \frac{D_2}{\Gamma, \Delta \vdash B \text{ true}}$$

D_2 is obtained from D_1 and E by substitution on terms (see Theorem 1 below): a proof term for A is induced from $\Box\Gamma$ in D_1 , i.e. by reducing all open variables in Γ by β -conversion, in turn providing a proof term for B in E , hence allowing the truth judgement. In computational terms, this rule formalizes β -reduction of B (value) with respect to all occurrences of its procedures (codes) in A .

Completeness is given by local expansion on $\Box(A \text{ true})$:

$$\frac{\frac{D_1}{\Box\Gamma, \Delta \vdash \Box(A \text{ true})} \Rightarrow_{Exp} \quad \frac{D_2}{\Box\Gamma, \Delta \vdash \Box(A \text{ true})} \quad \frac{\overline{\Gamma, a:A, \Delta \vdash A \text{ true}} \text{ Prem Rule}}{\Box\Gamma, a:A, \Box\Delta \vdash \Box(A \text{ true})} \Box I}{\Box\Gamma, \Delta \vdash \Box(A \text{ true})} \Box E$$

By this expansion one shows how $\Box E$ provides all the information needed to reconstruct $\Box(A \text{ true})$. Computationally, it reconstructs the value on code A . Notice that by the non-reducibility of $x:A$ to simple truth, one reaches completeness for the \Box -rules, which do not violate the meaning of hypotheses, as it is the case with the rules for necessity in [32]. On the other hand, given Definition 1, a side condition on multiple simultaneous substitutions is unavoidable, see [3].

Soundness is given by local reduction on $\Diamond(A \text{ true})$:

$$\frac{\frac{D_1}{\frac{\Gamma, x:A \vdash B \text{ true}^*}{\Box\Gamma, \Diamond(A \text{ true}) \vdash \Diamond(B \text{ true})} \Diamond I} \quad \frac{E}{\Gamma, \Delta \vdash A \text{ true}^*} \Diamond E}{\Gamma, \Delta \vdash B \text{ true}^*} \Diamond E \Rightarrow_{Redex} \quad \frac{D_2}{\Gamma, \Delta \vdash B \text{ true}^*}$$

²⁵Also in this case we have an analogy with the corresponding rules from [30]: we require the possibility judgement to be bounded explicitly to a $x:A$ in context, whereas their approach infers it directly from contextual truth; the corresponding elimination rule uses the semantic judgement involving the true^* predicate, whereas in their case a $C \text{ poss}$ judgement is inferred from contextually bounded truth.

D_2 is justified from D_1 and E by the Hypothesis Rule and $I\Diamond$: by E , Γ, Δ in reduced form will contain at least one formula of $type_{inf}$, which justifies $true^*$ in D_2 .²⁶

Finally, completeness by local expansion on $\Diamond(A \text{ true})$:

$$\begin{array}{c} \begin{array}{c} D_1 \\ \Rightarrow_{Exp} \\ \Diamond\Gamma, \Delta \vdash \Diamond(A \text{ true}) \end{array} \quad \frac{\begin{array}{c} D_2 \\ \Diamond\Gamma, \Delta \vdash \Diamond(A \text{ true}) \end{array} \quad \frac{\Gamma, x:A, \Delta \vdash A \text{ true}^*}{\Gamma, \Delta \vdash A \text{ true}^*} \text{Hypo Rule}}{\Diamond\Gamma, \Delta \vdash \Diamond(A \text{ true})} \Diamond E \quad \Diamond I \end{array}$$

This expansion shows how to reconstruct all the information needed to formulate $\Diamond(A \text{ true})$.

The local reductions and expansions are usually completed in terms of standard β -reduction and η -expansion, where by the former one shows how formulas terminate presenting their computational content, and by the latter how to abstract variables, implementing the extensionality principle. As our variables are not abstracted from corresponding terms with computational content, we cannot implement expansion. Its role is actually played by the Informational Type Formation rule.²⁷ The standard substitution operation of variables by constants is as usual indicated by $[x/a]B$ as the substitution of occurrences of x in B by a ; in our system this gives the relation between verification and truth. The corresponding modal version shows that term substitution satisfies the inclusion of \Diamond in \Box .

Theorem 1 (Substitution on terms). *The following substitutions hold:*

1. If $\Gamma, x:A, \Delta \vdash B \text{ true}^*$ and $\Gamma, \Delta \vdash a:A$, then $\Gamma, \Delta \vdash [x/a]B \text{ true}$.
2. If $\Box\Gamma, \Diamond(A \text{ true}), \Box\Delta \vdash \Diamond(B \text{ true})$ and $\Box\Gamma, \Box\Delta \vdash \Box(A \text{ true})$, then $\Box\Gamma, \Box\Delta \vdash \Box(B \text{ true})$.

Proof.

1. by induction on the first given derivation, using the Hypothesis Rule and the inclusion of $B \text{ true}^*$ in $B \text{ true}$; from the second premise all occurrences of A are declared being *type*, in particular those in $\Gamma, \Delta \vdash B \text{ true}^*$ by β -conversion, then $B \text{ true}$ follows as valid in any extension of Γ, Δ .
2. again by induction on the first given derivation: by $\Diamond E$ on the first premise one obtains an occurrence of $x:A$, using β -conversion on $A \text{ true}^*$ one obtains $B \text{ true}$ in the second premise; by $I\Box$ one finally obtains $\Box(B \text{ true})$.

□

²⁶Computationally, this reduction formalizes the naming of codes that are presented partially evaluated to program B .

²⁷It is worth remember that this limitation is avoided for any term in $type_{inf}$ that is actually β -reduced, as it then induces a corresponding term in *type*.

Substitution on the different truth predicates and modal judgements defines structural rules for the system:

Theorem 2 (Weakening). *The inference systems satisfies Weakening:*

1. If $\Gamma \vdash B \text{ true}$, then $\Gamma, a : A \vdash B \text{ true}$.
2. If $\Gamma \vdash B \text{ true}^*$, then $\Gamma, x : A \vdash B \text{ true}^*$.
3. If $\Box \Gamma \vdash \Box(B \text{ true})$, then $\Box \Gamma, \Box(A \text{ true}) \vdash \Box(B \text{ true})$.
4. If $\Diamond \Gamma \vdash \Diamond(B \text{ true})$, then $\Diamond \Gamma, \Diamond(A \text{ true}) \vdash \Diamond(B \text{ true})$.

Proof. By induction on derivations: in 1. uses the Premise Rule; in 2. uses the Hypothesis Rule; in 3. uses $I\Box$, in 4. uses $I\Diamond$. \square

Theorem 3 (Contraction). *The inference system satisfies Contraction:*

1. If $\Gamma, a_1 : A, a_2 : A \vdash B \text{ true}$, then $\Gamma, a : A \vdash [a_1 \approx a_2/a]B \text{ true}$.
2. If $\Gamma, x_1 : A, x_2 : A \vdash B \text{ true}^*$, then $\Gamma, x : A \vdash [x_1 \approx x_2/x]B \text{ true}^*$.
3. If $\Box \Gamma, a_1 : A, a_2 : A \vdash \Box(B \text{ true})$, then $\Box \Gamma, \Box(A \text{ true}) \vdash \Box(B \text{ true})$.
4. If $\Box \Gamma, x_1 : A, x_2 : A \vdash \Diamond(B \text{ true})$, then $\Box \Gamma, \Diamond(A \text{ true}) \vdash \Diamond(B \text{ true})$.

Proof. By induction on derivations: Reflexivity and Symmetry for proof terms in 1.; uniqueness of proof variables for $type_{inf}$ in 2.; in addition Truth Definition and $I\Box$ for 3.; Hypothetical Truth Definition and $I\Diamond$ for 4.. \square

Theorem 4 (Exchange). *The inference system satisfies Exchange:*

1. If $\Gamma, a_1 : A, a_2 : A \vdash B \text{ true}$, then $\Gamma, a_2 : A, a_1 : A \vdash B \text{ true}$.
2. If $\Gamma, x_1 : A, x_2 : A \vdash B \text{ true}^*$, then $\Gamma, x_2 : A, x_1 : A \vdash B \text{ true}^*$.
3. If $\Box \Gamma, a_1 : A, a_2 : A \vdash \Box(B \text{ true})$, then $\Box \Gamma, a_2 : A, a_1 : A \vdash \Box(B \text{ true})$.
4. If $\Box \Gamma, x_1 : A, x_2 : A \vdash \Diamond(B \text{ true})$, then $\Box \Gamma, x_2 : A, x_1 : A \vdash \Diamond(B \text{ true})$.

Proof. By induction and using the same properties on terms and variables as for Contraction. \square

4 Conclusions and further work

In this paper we have presented a type system that extends a constructive syntactic-semantic method inspired by Martin-Löf's type theory with an admissible treatment of refutable conditions for judgements. We have drawn a distinction between contents that are categorically justified and contents that are accepted as meaningful, but whose truth is debatable. The introduction of

epistemic modalities allows us to internalize contextual truth in the standard constructive approach.

The effect of extending a finitistic type-theoretical language by the $type_{inf}$ sort basically recalls the switch from extensional to intensional models, well-known for Martin-Löf’s Type Theory. Such a step, which essentially establishes the propositions-as-sets identity and the equivalence between truth and inhabitation (Curry-Howard correspondence), notoriously leads to impredicativity by defining a *type of all types*.²⁸ In the categorical models of the simple type theory, all types can be interpreted by countable sets: in the contextual format one obtains the same models as Cartesian closed categories, with objects Γ, Δ and mapping $\Gamma \rightarrow \Delta$ as co-products to introduce dependent types. These are the standard categorical models for intuitionistic propositional logic.

In the case of our modal contextual type-theory, it is not enough to explain an appropriate extension of the standard categorical approach to model the modalities, as it is the case with categorical models of constructive $S4$, see [1]. This is due to the unusual nature of the underlying polymorphism induced by $type_{inf}$ and hence requires non-standard models. Intuitively, the basic extension provided by the $type_{inf}$ sort in our language can be thought of as a combination of a typed terms structure (à la Church), by which terms have unique types, with a typing assignment procedure (à la Curry), where computation does not necessarily terminate. The extension of standard models of Constructive Type Theory by a function that introduces terms in $type_{inf}$ recalls the partial function space constructor which is usually lacking in type theories, a well-known fact from the theories of types such as from Martin-Löf’s, the Calculus of Constructions, the Nuprl implementation and the λ -calculus format.²⁹ $Type_{inf}$ can be seen as a constructor for *partial objects* and it modifies the standard models of type theory with dependent products by allowing terms on which substitutions are not defined. Under the propositions-as-types principle, such objects should be intended as partial proofs, and the sort of informational expressions identified under the $type_{inf}$ sort expresses precisely the notion of a process of proving that is admitted without a proper constructor, only provided none of its validity conditions is refuted. The analysis of these categories is reserved to another occasion. The issue of open proof terms is also receiving attention in systems of higher-order rewriting.

The extension to multi-modalities as distinctly indexed operators is the next obvious step for multi-agents and multi-source contextual modal type-theory. A multi-modal type theory based on the polymorphic setting here introduced is developed in [34], interpreted in terms of trust relations among agents in a network, where information flows in a strictly ordered way. Trust is defined as a second-order property and the formalization interprets communication pro-

²⁸This was first resolved by preserving the notion of universe of small types extended by dependent ones, the basis for an intensional minimal simple type theory of which, for example, the calculus in [16] is a corresponding λ -calculus of proofs, and which can be translated to an extensional classical system by the already mentioned negative translation of [20] (together with other possible extensions).

²⁹See e.g. [4], [11], [13], [12], [10], [9], [28].

cesses between sources (the prioritized structure generated by contexts of the dependently typed language) and a receiver (the indexed contextually derived judgement). The modal operator attached to a set of assumptions is induced from the priority relation among expressions; the modality prefixing the derived judgement is meant to represent the epistemic status of the receiver in the communication protocol, strictly determined by introduction and elimination rules for modalities that are based on canonical verification processes. A different interpretation is provided in [35] for a computational interpretation of programs equipped with locations for data accessibility in the context of distributed processing.

Acknowledgments

The author wishes to acknowledge the following persons for their help with discussion and comments at various stages of this research: Sara Negri, Peter Schröder-Heister, Valeria de Paiva, Frank Pfenning, Michele Friend and two anonymous referees.

References

- [1] N. Alechina, M. Mendler, V. de Paiva, and E. Ritter. Categorical and Kripke Semantics for Constructive S4 Modal Logic. In *Proceedings of the 15th International Workshop on Computer Science Logic*, volume 2142 of *Lecture Notes In Computer Science*, pages 292 – 307, 2001.
- [2] G. Bellin, V. de Paiva, and E. Ritter. Extended Curry-Howard Correspondence for a Basic Constructive Modal Logic. preprint; presented at M4M-2, ILLC, UvAmsterdam, 2001, 2001.
- [3] G.M. Bierman and V. de Paiva. Intuitionistic necessity revisited. Technical Report CSRP-96-10, School of Computer Science, University of Birmingham, 1996.
- [4] E. Bishop. *Foundations of Constructive Mathematics*, volume 3 of *Ergebnisse der Mathematik und ihrer Grenzgebiete*. Springer Verlag, 1985.
- [5] S. Buvač. Quantificational logic of context. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 600–606, 1996.
- [6] S. Buvač, V. Buvač, and I. Mason. Metamathematics of contexts. *Fundamenta Informaticae*, 23(3):412–419, 1995.
- [7] C. Cellucci. Why Proof? What is Proof?. In G. Corsi and R. Lupacchini (eds.), *Deduction, Computation, Experiment. Exploring the effectiveness of Proof*, pp. 1 – 27. Springer-Verlag, Berlin: 2008.

- [8] L.F.d. Cerro and A. Herzig. Combining classical and intuitionistic logic. In F. Baader and K.U. Schulz, editors, *Frontiers of Combining Systems, vol. 3*, pages 93–102. Kluwer Academic Publisher, 1996.
- [9] R. L. Constable and Scott Fraser Smith. Partial objects in constructive type theory. In *Proceedings of the Second LICS Symposium*, IEEE, 1987.
- [10] Robert L. Constable et al. *Implementing Mathematics in the Nuprl Proof Development System*. Englewood Cliffs, NJ, 1986.
- [11] T. Coquand. *Une Théorie des Constructions*. PhD thesis, Université Paris VII, 1985.
- [12] T. Coquand. Metamathematical investigations of a calculus of constructions. In P. Odifreddi, editor, *Logic and Computer Science*, volume 31 of *APIC Studies in Data Processing*, pages 91–122. London, 1990.
- [13] T. Coquand and G. Huet. The Calculus of Constructions. *Information and Computation*, 76(2/3):95–120, February/March 1988.
- [14] V. de Paiva. *Natural Deduction and Context as (Constructive) Modality*, volume 2680 of *Lecture Notes in Artificial Intelligence*, pages 116–129. Springer Verlag, 2003.
- [15] M. Fitting. *Basic modal logic*, volume 4 of *Handbook of Logic in Artificial Intelligence and Logic Programming*, pages 368–449. Oxford University Press, 1994.
- [16] J.-Y. Girard. Une extension de l’interprétation de Gödel à l’analyse, et son application à l’élimination des coupures dans l’analyse et la théorie des types. In *Proceedings of the Second Scandinavian Logic Symposium*. North-Holland, 1970.
- [17] N. Goethe, M. Friend. Confronting Ideals of Proof with the Ways of Proving of the Research Mathematician *Studia Logica*, vol. 96, pp. 273–288, 2010.
- [18] N.D. Jones, C. Gomard, and P. Sestoft. *Partial Evaluation and Automatic Program Generation*. International Series in Computer Science. Prentice-Hall International, 1993.
- [19] N.D. Jones, P. Sestoft, and H. Søndergaard. Mix: A self-applicable partial evaluator for experiments in compiler generation. *Lisp and Symbolic Computation*, 2:9–50, 1989.
- [20] A. Kolmogorov. On the principle of excluded middle. In J. Van Heijenoort, editor, *From Frege to Gödel: a source book in mathematical logic 1879-1931*, pages 414–437. Harvard University Press, 1967.
- [21] E.G.K. López-Escobar. Refutability and elementary number theory. *Indagationes Mathematicae (Koninklijke Nederlandse Akademie van Wetenschappen)*, 34:362–374, 1972.

- [22] P. Martin-Löf. *Intuitionistic Type Theory*. Bibliopolis, 1984.
- [23] P. Martin-Löf. Truth of a proposition, evidence of a judgement, validity of a proof. *Synthese*, 73(3):407–420, 1987.
- [24] P. Martin-Löf. *Analytic and Synthetic Judgments in Type Theory*, pages 87–99. Kluwer Academic Publisher, 1994.
- [25] P. Martin-Löf. On the meaning of the logical constants and the justifications of the logical laws. *Nordic Journal of Philosophical Logic*, 1(1):11–60, 1996.
- [26] J. McCarthy. Notes on formalizing context. In *Proceedings of the 13th Joint Conference on Artificial Intelligence (IJCAI-93)*, 1993.
- [27] M. Mendler and V. de Paiva. Constructive CK for Contexts. In *Proceedings of the first Workshop on Context Representation and Reasoning - CONTEXT05*, 2005.
- [28] E. Moggi. *The partial λ -calculus*. PhD thesis, School of Computer Sciences, University of Edinburgh, 1988.
- [29] A. Nanevski, F. Pfenning, and B. Pientka. Contextual modal type theory. *ACM Transactions on Computational Logic*, 9(3):1–48, 2008.
- [30] F. Pfenning and R. Davies. A judgemental reconstruction of modal logic. *Mathematical Structures in Computer Science*, 11:511–540, 2001.
- [31] G. Plotkin and C.P. Stirling. A framework for intuitionistic modal logic. In J.Y. Halpern, editor, *Theoretical Aspects of Reasoning about Knowledge*, 1986.
- [32] D. Prawitz. *Natural Deduction*. Almqvist & Wiksell, 1965.
- [33] G. Primiero. Epistemic modalities. In G. Primiero and S. Rahman, editors, *Acts of Knowledge: History, Philosophy and Logic*, volume 9 of *Tributes*, pages 207–232. College Publications, 2009.
- [34] G. Primiero and M. Taddeo. A prioritized multi-modal polymorphic type theory for formalizing trusted communications. Manuscript. Submitted., 2010.
- [35] G. Primiero. A multi-modal type system and its procedural semantics for safe distributed programming. In *Intuitionistic Modal Logic and Applications Workshop (IMLA11)*, Nancy, 2011. Manuscript.
- [36] Y. Rav, A Critique of a Formalist-Mechanist Version of the Justification of Arguments in Mathematicians’ Proof Practices. *Philosophia Mathematica*, (III) 15 (2007), pp. 291–320.
- [37] A.K. Simpson. *The Proof Theory and Semantics of Intuitionistic Modal Logic*. PhD thesis, University of Edinburgh. College of Science and Engineering. School of Informatics, 1994.

- [38] A. Tasistro. Formulation of Martin-Löf's Type Theory with Explicit Substitutions. Lic thesis, Department of Computing Science, Chalmers University of Technology and University of Göteborg, 1993.
- [39] A.S. Troelstra and D. van Dalen. *Constructivism in Mathematics: An Introduction*, volume I,II. North-Holland, Amsterdam, 1988.
- [40] M. van der Schaar. The assertion-candidate and the meaning of mood. *Synthese*, 159(1):61–82, 2007.
- [41] D. Wijesekera. Constructive modal logics I. *Annals of Pure and Applied Logic*, 50:271–301, 1990.