# Two formalizations of context: a comparison

P. Bouquet[1] and L. Serafini[2]

[1]Department of Computer and Management Sciences – University of Trento
Via Inama, 5 – 38100 Trento (Italy)

[2]ITC-IRST – Istituto per la Ricerca Scientifica e Tecnologica
Via Sommarive – 38050 Trento (Italy)

bouquet@cs.unitn.it serafini@itc.it

**Abstract.** We investigate the relationship between two well known formalizations of context: *Propositional Logic of Context* (PLC) [4], and *Local Models Semantics* (LMS) [11]. We start with a summary of the desiderata for a logic of context, mainly inspired by McCarthy's paper on generality in AI [15] and his notes on formalizing context [16]. We briefly present LMS, and its axiomatization using MultiContext Systems (MCS) [14]. Then we present a revised (and simplified) version of PLC, and we show that local vocabularies – as they defined in [4] – are inessential in the semantics of PLC. The central part of the paper is the definition of a class of LMS (and its axiomatization in MCS, called MMCC), which is provably equivalent to the axiomatization of PLC as described in [4]. Finally, we go back to the general desiderata and discuss in detail how the two formalisms fulfill (or do not fulfill) each of them.

## 1 Introduction

This paper is an investigation on the relationship between two well-known formalizations of context, namely the *Propositional Logic of Context* (PLC) [4] and *Local Models Semantics* (LMS) [11], axiomatized via Multi Context Systems [14, 13] (MCS)[1].

Both PLC and LMS/MCS address issues that were raised by McCarthy in his papers on generality in AI [15] and in his notes on formalizing context [16]. These issues can be summarized as a list of general desiderata for an adequate logic of context: (i) context should allow a simpler formalization of common sense axioms; (ii) context should allow us to restrict the vocabulary and the facts that are used to solve a problem on a given occasion; (iii) the truth of a common sense fact should be dealt with as dependent on a (possibly infinite) collection of assumptions (which implicitly define its *context*); (iv) there are no absolute, context independent facts, namely each fact must be stated into an appropriate context; (v) reasoning across different contexts should be modeled.

The main results of this paper are two: first, we technically prove that LMS is strictly more general than PLC; second, we argue that PLC only partially fulfills

[1] Hereafter, we will refer to the general framework of LMS together with its axiomatization via MCS as LMS/MCS.

the general desiderata for a logic of context. Technically, the first conclusion is justified by proving a theorem of equivalence between PLC and a special class of LMS; the second, by arguing that the technical features of PLC in fact prevent it from fulfilling some of the desiderata, as it fails to formalize a strong form of locality. The main representational drawback of PLC is that it does not fulfill the locality of vocabularies. Indeed, we prove that satisfiability and validity in PLC do not essentially depend on the vocabularies of each single context, and that any PLC-structure with partial vocabulary is equivalent to a PLC-structure where the universal vocabulary is associated to each context.

## 2 Logics for Contexts

In this section we summarize the two formalisms for contexts we will compare. LMS has been presented in [11], while [14, 13] propose a class of proof systems for LMS called Multi Context Systems (MCS). PLC is presented in [4].

### 2.1 Local Model Semantics and Multi Context Systems

Let $\{L_i\}_{i \in I}$ be a family of languages defined over a set of indexes $I$ (in the following we drop the index $i \in I$). Intuitively, each $L_i$ is the (formal) language used to describe the facts in the context $i$. We assume that $I$ is (at most) countable. Let $M_i$ be the class of all the models (interpretations) of $L_i$. We call $m \in M_i$ a *local model* (of $L_i$).

To distinguish the formula $\phi$ occurring in the context $i$ from the occurrences of the "same" formula $\phi$ in the other contexts, we write $\langle \phi, i \rangle$. We say that $\langle \phi, i \rangle$ is a labelled wff, and that $\phi$ is an $L_i$-wff. For any set of labeled formulas $\Gamma$, $\Gamma_i = \{\phi \mid \langle \phi, i \rangle \in \Gamma\}$[2]

**Definition 2.1 (Compatibility sequence).** A *compatibility sequence* $\mathbf{c} = \{c_i \subseteq M_i\}_{i \in I}$ is a family of sets of models of $L_i$. We call $c_i$ the $i$-th element of $\mathbf{c}$. A compatibility sequence is *nonempty* if at least one of its components is nonempty. A *compatibility sequence* $\mathbf{c}$ is a *compatibility chain* if all its elements contain at most one model.

A compatibility sequence represents a set of "instantaneous snapshots of the world" each of which is taken from the point of view of the associated context. Such a snapshot may be incomplete, that is the truth value of some of the propositions in a context cannot be inferred from the information contained in the snapshot. This is formalized by associating *sets of models* to each context rather than a single model.

[2] As contexts have distinct languages, it is not the case that the same formula can belong to different contexts. However, it is possible that two formulas with the same syntax occur in different contexts. In this is the case, it should be clear that the "same" formula in two distinct contexts is interpreted in different sets of local models, and therefore they have different semantics.

**Definition 2.2 (Compatibility relation and LMS model).** A *compatibility relation* is a set of compatibility sequences. A *LMS model* is a compatibility relation that contains a nonempty compatibility sequence.

A compatibility relation formalizes all possible sets of "instantaneous snapshots of the world". Indeed, as contexts are not independent viewpoints, some combinations can never happen. This is expressed via a collection of *compatibility constraints*, namely conditions that say when local models of different contexts can belong to the same compatibility sequence.

**Definition 2.3 (Satisfiability and Entailment).** Let $\models$ be the propositional classical satisfiability relation. We extend the definition of $\models$ as follows:

1. for any $\phi \in L_i$, $c_i \models \phi$ if, for all $m \in c_i$, $m \models \phi$;
2. $\mathbf{c} \models \langle \phi, i \rangle$ if $c_i \models \phi$;
3. $\mathbf{C} \models \langle \phi, i \rangle$ if for all $\mathbf{c} \in \mathbf{C}$, $\mathbf{c} \models \langle \phi, i \rangle$;
4. $\Gamma_i \models_{c_i} \phi$ if, for all $m \in c_i$, $m \models \Gamma_i$ implies $m \models \phi$;
5. $\Gamma \models_{\mathbf{c}} \langle \phi, i \rangle$ if, either there is a $j \neq i$, such that $c_j \not\models \Gamma_j$, or $\Gamma_i \models_{c_i} \phi$;
6. $\Gamma \models_{\mathbf{C}} \langle \phi, i \rangle$, if for all $\mathbf{c} \in \mathbf{C}$, $\Gamma \models_{\mathbf{c}} \langle \phi, i \rangle$;
7. For any class of models $\mathfrak{C}$, $\Gamma \models_{\mathfrak{C}} \langle \phi, i \rangle$, if, for all LMS-models $\mathbf{C} \in \mathfrak{C}$, $\Gamma \models_{\mathbf{C}} \langle \phi, i \rangle$.

We adopt the usual terminology of satisfiability and entailment for the statements about the relation $\models$. Thus we say that $\mathbf{c}$ satisfies $\phi$ at $i$, or equivalently, that $\phi$ is true in $c_i$, to refer to the fact that $c_i \models \phi$. We say that $\Gamma$ entails $\langle \phi, i \rangle$ in $\mathbf{c}$ to refer to the fact that $\Gamma \models_{\mathbf{c}} \langle \phi, i \rangle$. Similar terminology is adopted for $\Gamma \models_{\mathbf{C}} \langle \phi, i \rangle$ and $\Gamma \models \langle \phi, i \rangle$.

MultiContext Systems (MCS) [14] are a class of proof systems for LMS[3]. The key notion of an MCS is that of bridge rule.

**Definition 2.4 (Bridge Rule).** A *bridge rule* on a set of indices $I$ is a schema of the form:

$$\frac{\langle A_1, i_1 \rangle \quad \dots \quad \langle A_n, i_n \rangle}{\langle A, i \rangle} \; br$$

where $i_1, \dots, i_n, i \in I$ and $A_1, \dots, A_n, A$ are schematic formulas. A bridge rule can be associated with a *restriction*, namely a criterion which states the conditions of its applicability.

**Definition 2.5 (MultiContext System (MCS)).** A *Multicontext System (MCS)* for a family of languages $\{L_i\}$, is a pair $\mathrm{MS} = \langle \{C_i = \langle L_i, \Omega_i, \Delta_i \rangle\}, \Delta_{br} \rangle$, where each $C_i = \langle L_i, \Omega_i, \Delta_i \rangle$ is a theory (on the language $L_i$, with axioms $\Omega_i$ and natural deduction inference rules $\Delta_i$), and $\Delta_{br}$ is a set of bridge rules on $I$.

MCSs are a generalization of Natural Deduction (ND) systems [18]. The generalization amounts to using formulae tagged with the language they belong to.

[3] In this paper, we present a definition of MC system which is suitable for our purposes. For a fully general presentation, see [14].

This allows for the effective use of the multiple languages. The deduction machinery of an MCS is the composition of two kinds of inference rules: *local rules*, namely the inference rules in each $\Delta_i$, and *bridge rules*. Local rules formalize reasoning within a context (i.e. are only applied to formulae with the same index), while bridge rules formalize reasoning across different contexts.

Deductions in a MCS are trees of formulae which are built starting from a finite set of assumptions and axioms, possibly belonging to distinct languages, and by a finite number of application of local rules and bridge rules. A formula $\langle \phi,\ i \rangle$ is *derivable* from a set of formulae $\Gamma$ in a MC system MS, in symbols, $\Gamma \vdash_{\mathrm{MS}} \langle \phi,\ i \rangle$, if there is a deduction with bottom formula $\langle \phi,\ i \rangle$ whose undischarged assumptions are in $\Gamma$. A formula $\langle \phi, i \rangle$ is a *theorem* in MS, in symbols $\vdash_{MS} \langle \phi,\ i \rangle$, if it is derivable from the empty set. The standard notation for deductions can be obtained by drawing a tree of labelled formulas. An example is presented in Appendix 2.

### 2.2 Propositional Logics of Contexts Revisited

In this section we present a slightly simplified (but provably equivalent) version of the logic PLC as presented in [4]. The simplification concerns both the semantics and the axiomatization. Given a set $\mathbb{K}$ of labels, intuitively denoting contexts, the language of PLC is a multi modal language on a set of atomic propositions $\mathbb{P}$ with the modality $ist(\kappa, \phi)$ for each context (label) $\kappa \in \mathbb{K}$. More formally, the set of well formed formulas $\mathbb{W}$ of PLC, based on $\mathbb{P}$, are

$$\mathbb{W} := \mathbb{P} \cup (\neg \mathbb{P}) \cup (\mathbb{P} \supset \mathbb{P}) \cup ist(\mathbb{K}, \mathbb{P})$$

The other propositional connectives are defined as usual. The formula $ist(\kappa, \phi)$ can be read as: $\phi$ holds (is true) in the context $\kappa$. PLC allows to describe how a context is views from another context. For this PLC introduces sequences of contexts (labels). Let $\mathbb{K}^*$ denote the set of finite contexts sequences and let $\overline{\kappa} = \kappa_1 \ldots \kappa_n$ denote any (possible empty) element of $\mathbb{K}^*$. Intuitively the sequence of contexts $\kappa_1\kappa_2$ represents how context $\kappa_2$ is viewed from context $\kappa_1$. Therefore, the intuitive meaning of the formula $ist(\kappa_2, \phi)$ in the context $\kappa_1$ is that $\phi$ holds in the context $\kappa_2$, from the point of view of $\kappa_1$. A similar interpretation can be given to formulas in sequences of contexts longer than 2. As a consequence, satisfiability is defined with respect to a context sequence. Indeed a model for PLC associates a set of partial truth assignments to each context sequence. In the following definition we use $A \to_p B$ to denote the set of *partial* functions from $A$ to $B$ and $\mathbf{P}(A)$ to denote the powerset of $A$.

**Definition 2.6.** A *PLC-model* $\mathfrak{M}$ is a partial function that maps each context sequence $\overline{\kappa} \in \mathbb{K}^*$ into a set of partial truth assignments for $\mathbb{P}$.

$$\mathfrak{M} \in (\mathbb{K}^* \to_p \mathbf{P}(\mathbb{P} \to_p \{\text{true}, \text{false}\}))$$

Satisfiability and validity are defined with respect to a vocabulary. A *vocabulary* is a relation $\mathsf{Vocab} \subseteq \mathbb{K}^* \times \mathbb{P}$ that associates a subset of primitive propositions with each context sequence $\overline{\kappa}$. Each formula $\phi$ in a context sequence $\overline{\kappa}$ implicitly defines a vocabulary, denoted by $\mathsf{Vocab}(\overline{\kappa},\phi)$, which intuitively consists of the minimal vocabulary necessary to build the formula $\phi$ in the context sequence $\overline{\kappa}$. $\mathsf{Vocab}(\overline{\kappa},\phi)$ is recursively defined as follows:

$$\begin{aligned}\mathsf{Vocab}(\overline{\kappa},p) &= \{\langle\overline{\kappa},p\rangle\}\\ \mathsf{Vocab}(\overline{\kappa},\neg\phi) &= \mathsf{Vocab}(\overline{\kappa},\phi)\\ \mathsf{Vocab}(\overline{\kappa},\phi\supset\psi) &= \mathsf{Vocab}(\overline{\kappa},\phi)\cup\mathsf{Vocab}(\overline{\kappa},\psi)\\ \mathsf{Vocab}(\overline{\kappa},ist(\kappa,\phi)) &= \mathsf{Vocab}(\overline{\kappa}\kappa,\phi)\end{aligned}$$

Analogously, a model $\mathfrak{M}$ defines a vocabulary denoted by $\mathsf{Vocab}(\mathfrak{M})$. $\langle\overline{\kappa},p\rangle\in\mathsf{Vocab}(\mathfrak{M})$ if and only if $\mathfrak{M}(\overline{\kappa})$ is defined and, for all $\nu\in\mathfrak{M}(\overline{\kappa})$, $\nu(p)$ is defined (where $\nu$ is a truth assignment to atomic propositions). Satisfaction in a model $\mathfrak{M}$ of a formula $\phi$ in a context sequence $\overline{\kappa}$ is defined only when $\mathsf{Vocab}(\overline{\kappa},\phi)\subseteq\mathsf{Vocab}(\mathfrak{M})$. Similarly, validity of a formula is defined only on the class of models that contain the vocabulary of the formula.

**Definition 2.7 (Satisfiability and Validity).** A formula $\phi$ such that $\mathsf{Vocab}(\overline{\kappa},\phi)\subseteq\mathsf{Vocab}(\mathfrak{M})$ is satisfied by an assignment $\nu\in\mathfrak{M}(\overline{\kappa})$, in symbols $\mathfrak{M},\nu\models_{\overline{\kappa}}\phi$, according to the following clauses:

1. $\mathfrak{M},\nu\models_{\overline{\kappa}} p$ iff $\nu(p)=$ true;
2. $\mathfrak{M},\nu\models_{\overline{\kappa}}\neg\phi$ iff not $\mathfrak{M},\nu\models_{\overline{\kappa}}\phi$;
3. $\mathfrak{M},\nu\models_{\overline{\kappa}}\phi\supset\psi$ iff not $\mathfrak{M},\nu\models_{\overline{\kappa}}\phi$ or $\mathfrak{M},\nu\models_{\overline{\kappa}}\psi$;
4. $\mathfrak{M},\nu\models_{\overline{\kappa}} ist(\kappa,\phi)$ iff for all $\nu'\in\mathfrak{M}(\overline{\kappa}\kappa)$, $\mathfrak{M},\nu'\models_{\overline{\kappa}\kappa}\phi$;
5. $\mathfrak{M}\models_{\overline{\kappa}}\phi$ iff for all $\nu\in\mathfrak{M}(\overline{\kappa})$; $\mathfrak{M},\nu\models_{\overline{\kappa}}\phi$;
6. $\models_{\overline{\kappa}}\phi$ iff for all PLC-model $\mathfrak{M}$, such that $\mathsf{Vocab}(\overline{\kappa},\phi)\subseteq\mathsf{Vocab}(\mathfrak{M})$, $\mathfrak{M}\models_{\overline{\kappa}}\phi$.

$\phi$ is *valid* in a context sequence $\overline{\kappa}$ if $\models_{\overline{\kappa}}\phi$; $\phi$ is *satisfiable* in a context sequence $\overline{\kappa}$ if there is a PLC-model $\mathfrak{M}$ such that $\mathfrak{M}\models_{\overline{\kappa}}\phi$. A set of formulas $T$ is satisfiable at a context sequence $\overline{\kappa}$ if there is a model $\mathfrak{M}$ such that $\mathfrak{M}\models_{\overline{\kappa}}\phi$ for all $\phi\in T$.

The axiomatization of PLC is a Hilbert style calculus with axioms and rules reported in Figure 1. A formula $\phi$ is derivable from a set of formulas $\Gamma$ in the context sequence $\overline{\kappa}$, in symbols $\Gamma\vdash_{\overline{\kappa}}\phi$, if and only if there are a finite set $\phi_1,\ldots,\phi_n$ of formulas in $\Gamma$, such that the formula $\vdash_{\overline{\kappa}}\phi_1\supset(\phi_2\ldots\supset(\phi_n\supset\phi)\ldots)$ is derivable from (PL), (K), and ($\Delta$) by applying the inference rules (MP) and (CS). A set of formulas $\Gamma$ is consistent in a context sequences $\overline{\kappa}$ if, for every $\phi$, it is not the case that $\Gamma\vdash_{\overline{\kappa}}\phi$ and $\Gamma\vdash_{\overline{\kappa}}\neg\phi$.

The proposed axiomatization is simpler than that contained in [4], as we do not introduce the following axiom:

$$(\Delta_-)\quad\vdash_{\overline{\kappa}} ist(\kappa_1,\neg ist(\kappa_2,\phi)\vee\psi)\supset ist(\kappa_1,\neg ist(\kappa_2,\phi))\vee ist(\kappa_1,\psi)$$

The reason is that it can be derived in PLC as follows:

| | | |
|---|---|---|
| (PL) | $\vdash_{\overline{\kappa}} \phi$ | If $\phi$ is an instance of a classical tautology |
| (K) | $\vdash_{\overline{\kappa}} ist(\kappa, \phi \supset \psi) \supset ist(\kappa, \phi) \supset ist(\kappa, \psi)$ | |
| ($\Delta$) | $\vdash_{\overline{\kappa}} ist(\kappa_1, ist(\kappa_2, \phi) \vee \psi) \supset ist(\kappa_1, ist(\kappa_2, \phi)) \vee ist(\kappa_1, \psi)$ | |
| (MP) | $\frac{\vdash_{\overline{\kappa}} \phi \quad \vdash_{\overline{\kappa}} \phi \supset \psi}{\vdash_{\overline{\kappa}} \psi}$ | |
| (CS) | $\frac{\vdash_{\overline{\kappa}\kappa} \phi}{\vdash_{\overline{\kappa}} ist(\kappa, \phi)}$ | |

**Fig. 1.** Axioms and inference rules for PLC

$$\vdash_{\overline{\kappa}\kappa_1} ist(\kappa_2, \phi) \vee \neg ist(\kappa_2, \phi) \quad \text{By (PL)} \tag{1}$$

$$\vdash_{\overline{\kappa}} ist(\kappa_1, ist(\kappa_2, \phi) \vee \neg ist(\kappa_2, \phi)) \quad \text{By (CS)} \tag{2}$$

$$\vdash_{\overline{\kappa}} ist(\kappa_1, ist(\kappa_2, \phi)) \vee ist(\overline{\kappa}_1, \neg ist(\kappa_2, \phi)) \quad \text{By } (\Delta) \text{ and (MP)} \tag{3}$$

$$\vdash_{\overline{\kappa}\kappa_1} ist(\kappa_2, \phi) \supset (\neg ist(\kappa_2, \phi) \vee \psi) \supset \psi \quad \text{By (PL)} \tag{4}$$

$$\vdash_{\overline{\kappa}} ist(\kappa_1, ist(\kappa_2, \phi) \supset (\neg ist(\kappa_2, \phi) \vee \psi) \supset \psi) \quad \text{By (CS)} \tag{5}$$

$$\vdash_{\overline{\kappa}} ist(\kappa_1, ist(\kappa_2, \phi)) \supset (ist(\kappa_1, \neg ist(\kappa_2, \phi) \vee \psi) \supset ist(\kappa_1, \psi)) \quad \text{By (K)} \tag{6}$$

$$\vdash_{\overline{\kappa}} ist(\kappa_1, \neg ist(\kappa_2, \phi) \vee \psi) \supset ist(\kappa_1, \neg ist(\kappa_2, \phi)) \vee ist(\kappa_1, \psi) \quad \text{From (2) and (6) by PL and MP} \tag{7}$$

**Theorem 2.1 (Completeness [4]).** *Any set of formulas $\Gamma$ is consistent in $\overline{\kappa}$ if and only $\Gamma$ is satisfiable in $\overline{\kappa}$.*

## 3 Reconstructing PLC in LMS

The reconstruction of PLC in LMS is based on the observation that PLC is multi-modal K, restricted to a vocabulary, and extended with the axiom ($\Delta$).

In [14], a family of MCS, called MBK, is used to represent multi-modal K; moreover, [11] presents the definition of a LMS for MBK (and the corresponding completeness result). To prove that PLC can be represented in LMS/MCS, we first show that vocabularies in PLC play no logical role. Then we extend MBK for multi-modal K, and we define the MC system MMCC, in which ($\Delta$) is a theorem.

As far as vocabularies are concerned, it is worth remarking that PLC does not make any essential use of them as far as satisfiability and validity are concerned. To argue this, we prove the following theorem of reduction. Proofs of theorems are presented in the appendix.

**Definition 3.1.** A *complete vocabulary* is the vocabulary that associates to each sequence of contexts the whole set of propositions.

**Theorem 3.1 (Complete vocabularies).** *A formulas $\phi$ is valid in a context sequence $\overline{\kappa}$ if and only if it is satisfied at $\overline{\kappa}$ by all the PLC-models with complete vocabulary. Similarly, a formula $\phi$ is satisfiable in a context sequence $\overline{\kappa}$ if and only if there is a PLC-model with complete vocabulary that satisfies $\phi$ at $\overline{\kappa}$.*

Let us now reconstruct PLC in LMS and MCS. For each (possibly empty) sequence $\overline{\kappa} \in \mathbb{K}^*$, the language $L_{\overline{\kappa}}$ is the smallest propositional language that contains $\mathbb{P}$ and the *atomic formula* $ist(\kappa, \phi)$ for any $\kappa \in \mathbb{K}$ and any formula $\phi \in L_{\overline{\kappa}\kappa}$. Notice that, unlike PLC, the formula $ist(\kappa, \phi)$ is an atomic formula of $L_{\overline{\kappa}}$, and not the application of a modal operator to the formula $\phi$.

**Definition 3.2.** An MBK($\mathbb{K}^*$)-model is a model for the family of languages $\{L_{\overline{\kappa}}\}_{\overline{\kappa}\in\mathbb{K}^*}$, such that, for any $\mathbf{c} \in \mathbf{C}$ and $\overline{\kappa}\kappa \in \mathbb{K}^*$:

1. $\mathbf{c}$ is a compatibility chain;
2. if $\mathbf{c} \models \langle ist(\kappa, \phi), \overline{\kappa}\rangle$, then $\mathbf{c} \models \langle \phi, \overline{\kappa}\kappa\rangle$;
3. if $\mathbf{c}' \models \langle \phi, \overline{\kappa}\kappa\rangle$ for all $\mathbf{c}' \in \mathbf{C}$ with $c_{\overline{\kappa}} = c'_{\overline{\kappa}}$, then $\mathbf{c} \models \langle ist(\kappa, \phi), \overline{\kappa}\rangle$.

**Definition 3.3.** MBK($\mathbb{K}^*$) is an MCS on the family of languages $\{L_{\overline{\kappa}}\}_{\overline{\kappa}\in\mathbb{K}}$, where, for any $\overline{\kappa}$, $\Omega_{\overline{\kappa}}$ is empty and $\Delta_{\overline{\kappa}}$ is the set of propositional natural deduction inference rules, and $\Delta_{br}$ is the following set of bridge rule schemas:

$$\frac{\langle ist(\kappa, \phi), \overline{\kappa}\rangle}{\langle \phi, \overline{\kappa}\kappa\rangle}\,\mathcal{R}_{dn\overline{\kappa}\kappa} \qquad \frac{\langle \phi, \overline{\kappa}\kappa\rangle}{\langle ist(\kappa, \phi), \overline{\kappa}\rangle}\,\mathcal{R}_{up\overline{\kappa}\kappa}$$

RESTRICTION $\mathcal{R}_{up\overline{\kappa}\kappa}$ is applicable only if $\langle \phi, \overline{\kappa}\kappa\rangle$ does not depend upon any assumptions in $\overline{\kappa}\kappa$. $\mathcal{R}_{up\overline{\kappa}\kappa}$ and $\mathcal{R}_{dn\overline{\kappa}\kappa}$ are called reflection up and reflection down, respectively.

The soundness and completeness theorems for MBK($\mathbb{K}^*$), with respect to the class of MBK($\mathbb{K}^*$)-models, is given in [11].

**Theorem 3.2 (Soundness and Completeness).** $\Gamma \models_{\mathrm{MBK}(\mathbb{K}^*)} \langle \phi, \overline{\kappa}\rangle$ *if and only if* $\Gamma \vdash_{\mathrm{MBK}(\mathbb{K}^*)} \langle \phi, \overline{\kappa}\rangle$.

**Definition 3.4 (MMCC-model).** An *MMCC model* is defined as a MBK($\mathbb{K}^*$)-model, with the further condition that:

4 if $\mathbf{c} \models \langle ist(\kappa', \phi), \overline{\kappa}\kappa\rangle$, then $\mathbf{c} \models \langle ist(\kappa, ist(\kappa', \phi)), \overline{\kappa}\rangle$.

Condition 4 of the previous definition characterizes the axiom ($\Delta$).

**Theorem 3.3.** *Any* MBK($\mathbb{K}^*$)*-model* $\mathbf{C}$ *is an MMCC-model if and only if* $\mathbf{C} \models \langle (\Delta), \overline{\kappa}\rangle$*, for any* $\overline{\kappa}$*.*

We now extend MBK($\mathbb{K}^*$) in order to prove the axiom ($\Delta$).

**Definition 3.5 (MMCC).** MMCC is an MCS defined as MBK($\mathbb{K}^*$) where the restriction of $\mathcal{R}_{up\overline{\kappa}\kappa}$ is applied only if the premise of $\mathcal{R}_{up\overline{\kappa}\kappa}$, is not of the form $\langle ist(\kappa', \psi), \overline{\kappa}\kappa\rangle$.

Now, we need to prove that the extension of MBK($\mathbb{K}^*$) is the right one, namely that MMCC is sound and complete w.r.t. the class of MMCC-models.

**Theorem 3.4 (Soundness and Completeness of MMCC).** *MMCC is sound and complete w.r.t. the set of MMCC-models. In symbols*

$$\Gamma \vdash \langle \phi, \overline{\kappa} \rangle \text{ if and only if } \Gamma \models \langle \phi, \overline{\kappa} \rangle$$

The last step is to state the equivalence between MMCC and PLC, as far as provability is concerned.

**Theorem 3.5 (MMCC is equivalent to PLC).** $\vdash_{\overline{\kappa}} \phi$ *if and only if* $\vdash \langle \phi, \overline{\kappa} \rangle$.

## 4 Discussion

In the introduction, we recalled some of the desiderata for a logic of context that LMS and PLC share. In this section, we discuss if these desiderata are fulfilled by PLC and LMS, and how.

**Contexts allow simpler formalization of common sense.** LMS and MCS have been successfully used for formalizing important aspects of common-sense knowledge and reasoning. Among others PLC and its extensions have been used to formalize information integration, planning composition and reuse, and discourse representation [17]. LMS has been used to formalize reasoning about beliefs [1, 7, 8, 6, 12], a solution to the qualification problem [3], multiple viewpoints [11], meta-reasoning [9, 10] multi-agent specification languages [2]. A discussion on the above results is beyond the scope of this paper. We only remark that the two formalisms seem to have fulfilled the goal of allowing new formalizations of various domains.

**Contexts allow formalizations using a restricted vocabulary.** PLC meets this desideratum only in a weak form. First of all, there is no neat way of restricting the language of a context. If, on the one hand, partial truth assignments are introduced as a way of restricting the vocabulary of a context $\kappa$, on the other hand the formula $ist(\kappa, \phi)$ is a well-formed formula for whatever $\phi$. This observation is formally supported by Theorem 3.1, which shows that the notion of vocabulary defined in the semantics of PLC does not impact the definition of satisfiability and validity of formulas. Therefore, PLC with different vocabularies for different contexts is essentially equivalent to PLC with a unique global vocabulary shared by all the contexts.

One of the consequences of this formal property of PLC is that we in a context one can always represent what is true in another context. Technically, $ist(\kappa, \phi)$ is a well formed formula in the context $\overline{\kappa}$, if and only if, $\phi$ is well formed in the context sequence $\overline{\kappa}\kappa$. Furthermore, under the so called "flatness" hypothesis, namely when the context sequence $\overline{\kappa}\kappa$ coincides with the context $\kappa$, $\phi$ is well formed in the context $\kappa$, if and only if $ist(\kappa, \phi)$ is well formed in all the other contexts. This property is very strong (perhaps too strong) for a logic of context, as it has two main consequences. First, in every context we can always represent all the propositions that may be true in any other context. For example, in the context of the Sherlock Holmes story we would be able to represent the fact that the sentence "The Millenium bug produced no serious damages" is true in the context of XXI century technology, which seems a little odd. Second, if in a context we state that a proposition is true (or false) in some other context, we force that fact to be expressible in the language of that context. In other words, if in $\kappa$ we state that $\phi$ is true (or false) in some other context $\kappa'$, then $\phi$ is necessarily a well formed formula of $\kappa\kappa'$. For instance, we cannot say that "Galileo believed that the millenium bug produced no serious damages" is false in the context of XXI century technology without having a formula for the fact "the millenium bug produced no serious damages" in the context of Galileo's beliefs.

Unlike PLC, in LMS we are allowed to associate a distinct language to each context. This means that, if we put the formula $ist(\kappa', \phi)$ in the language $L_\kappa$ of the context $\kappa$, we don't need to have $\phi$ as a well formed formula in the language $L_{\kappa\kappa'}$ of the context $\kappa\kappa'$ or in the context $\kappa'$. Going back to the Galileo example, in the context $T$ of XXIst century technology, we can have the following formula, which states that Galileo believed that the millenium bug produced no serious damages,

$$ist(G, MB) \tag{8}$$

without forcing $MB$ to be a well formed formula in the context $G$ of Galileo's beliefs. The interpretation of (8) is given by the local models of the language $L_T$. The fact that $MB$ is not a formula in the language of the context $G$ simply means that one cannot impose any constraint on the interpretation of (8) in $T$ and the interpretation of $MB$ in the context $G$.

To sum up, it is worth stressing the following difference between PLC and MMCC: in the former, this strong property is part of the logic itself, whereas in the latter it is an additional constraint on the definition of a MMCC model that can be relaxed at any time. This gives MMCC a further degree of flexibility.

**Contexts allow to "localize" reasoning.** Both LMS/MCS and PLC allow facts to be partitioned into different contexts. The set of facts belonging to a context can be defined in two ways: *directly*, by explicitly enumerating the facts that are in a context (using expressions of the form $\langle\phi, \kappa\rangle$); and *compositionally*, by defining the set of facts of a context from sets of facts in other contexts. In PLC this is done via *lifting axioms*, which are formulas of the form:

$$\langle ist(\kappa_1, \phi) \supset ist(\kappa_2, \phi),\ \kappa_{ext}\rangle \tag{9}$$

Intuitively, (9) says that, if $\kappa_1$ contains the fact $\phi$, than this fact is lifted also in the context $\kappa_2$. In PLC, lifting axioms are necessarily stated in an external context, which must be expressive enough to represent the truth of facts in both contexts (using *ist*-formulae). In LMS, compositional definition is formalized via compatibility relation, and represented in MCS via bridge rules. For example, the compatibility relation corresponding to the lifting axiom (9) is:

$$\text{for any } \mathbf{c} \in \mathbf{C}, \text{ if } \mathbf{c} \models \langle \phi,\ \kappa_1 \rangle, \text{ then } \mathbf{c} \models \langle \phi,\ \kappa_2 \rangle \tag{10}$$

or some equivalent formulation of (10). The corresponding bridge rule is:

$$\frac{\langle \kappa_1,\ \phi \rangle}{\langle \kappa_2,\ \phi \rangle}\ br_{(10)} \tag{11}$$

Moreover, LMS/MCS allows us to compositionally define the content of a context via lifting axioms, as in PLC. Indeed, to lift a fact $\phi$ from $\kappa_1$ to $\kappa_2$, it is enough to define an external context connected with $\kappa_1$ and $\kappa_2$ via reflection rules (see Definition 3.3) and explicitly add axiom (9) to this context. This approach was used in the solution to the qualification problem presented in [3].

The main difference between a bridge rule such as (11) and a lifting axiom such as (9) is that the first case does not require to define an external global context, capable of representing the truth of all the other contexts, the second does. Notice that, in some cases, defining an external context might be very expensive – especially when there are many interconnected contexts –, as the external context essentially duplicates all the information presented in each single context. Still, having a context that contains all lifting axioms may have some advantages, e.g., it allows one to reason about them. This is useful, for instance, to discover whether certain lifting axioms are redundant, or lead to inconsistent contexts. Since in LMS/MCS both approaches are possible, the formalism seems to be more flexible.

**The truth of a common sense fact always depends on their context.** In PLC this is only partially true. It holds for the formulas that do not contain the *ist* predicate, since a PLC-model associates to each context sequence a set of evaluations for primitive propositions, which defines the truth of facts for that context. However, for *ist* formulas this is not completely true, as the truth of a formula $ist(\kappa, \phi)$ is independent of the assignments of the contexts in which it occurs. Indeed, the following property holds:

$$\text{for any pair of assignments } \nu, \nu' \in \mathfrak{M}(\overline{\kappa}),\ \mathfrak{M}, \nu \models_{\overline{\kappa}} ist(\kappa, \phi) \text{ if and only if } \mathfrak{M}, \nu' \models_{\overline{\kappa}} ist(\kappa, \phi) \tag{12}$$

The truth of $ist(\kappa, \phi)$ in $\overline{\kappa}$, is actually defined by the assignments of the context $\overline{\kappa}\kappa$ and the assignments of the context $\overline{\kappa}$ cannot affects such a truth in any way. One of the effects of this definition is that the formula ($\Delta$) is valid. However, in our view, it seems quite hard to argue that such a formula describes a genuine principle of contextual reasoning.

In LMS, the satisfiability of a formula of the type $ist(\kappa, \phi)$ is local to the context in which the formula is asserted (this is one of the distinguished properties of LMS in general), and therefore such a problem can be avoided. Indeed, in order to prove the equivalence between MMCC and PLC, we had to impose a very strong compatibility relation such as condition 4 of Definition 3.5. However, it can be easily relaxed, as it is not part of the underlying logic.

**There are no absolute, context independent facts.** This is true both with PLC and LMS. Indeed we have that each formulas is prefixed by a label that contextualizes it. Neither PLC nor LMS and have absolute external language. In both formalisms we have the two formulas $\langle\phi,\ \kappa_1\rangle$ and $\langle\psi,\ \kappa_2\rangle$ but there is no formula that represents the conjunction or the implication or any other kind of logical relation between the two formulas.

In PLC, however, the context labelled with the empty string $\epsilon$ can represent the facts contained in any other context, and therefore any logical relation between them. For instance the implication between the formulas $\langle\phi,\ \overline{\kappa}_1\rangle$ and $\langle\psi,\ \overline{\kappa}_2\rangle$, is representable in the root context $\epsilon$, by the formula $\langle ist(\overline{\kappa}_1, \phi) \supset ist(\overline{\kappa}_2, \psi),\ \epsilon\rangle$.

In LMS and MCS we have that contexts are not necessarily organized in a hierarchy, so that we can consider an MCS composed of two simple contexts $\kappa_1$ and $\kappa_2$, each of which has its own vocabulary. In such an MCS a logical relation, e.g., conjunction or disjunction of two formulas in $\kappa_1$ and $\kappa_2$ is not representable in any contexts.

**Reasoning "navigates" across contexts.** The reasoning systems associated to PLC and LMS are very different. Despite the fact that they both implement reasoning in different contexts, PLC is about *validity*, while LMS is about *logical consequence* between formulas. A reasoning system about validity allows one to prove that a formula is true in the whole class of models, while a reasoning system about logical consequence allows one to prove that a formula is true in the class of models that satisfy a set of formulas called assumptions.

For a formal system for contexts, it is very important to be able to represent logical consequence across different contexts, in order to adequately formalize reasoning across contexts. Logical consequence across different contexts, formalizes the dependence between the truth of formulas in different contexts, which is the basis of the reasoning steps that allow switching from one context to another. While in many (single language) formal system, logical consequence can be reduced to validity of an implication formula (think for instance to the deduction theorem in propositional logic, where $\phi \models \psi$ is equivalent to $\models \phi \supset \psi$), in a logic of contexts, in general this rewriting is not possible. For instance, to represent the fact that $\psi$ in $\kappa$ is a logical consequence of $\phi$ in $\kappa'$, one needs a third context containing the language of $\kappa$ and $\kappa'$, where it is possible to express the implication between $\phi$ and $\psi$. This of course is not always guaranteed.

## 5 Conclusions

To the best of our knowledge, this paper is the first attempt to make a technical comparison between PLC and LMS/MCS. Even though the two formalisms are perhaps the most significant attempts to provide a logic of context in AI, so far the comparison between them has been limited to a few lines describing related work in papers by authors of the two proponent groups. We hope that the results provided here will help to clarify the the technical and conceptual differences between the two approaches.

A final remark. In the paper, we only considered PLC and did not discuss the quantificational logic of context proposed in [5]. However, [19] shows that such a logic is provably equivalent to a quantified multimodal logic K45 + T on single modality. Using the results on the reduction of modal logic to MCS presented in [14], the equivalence between Buvač's quantificational logic of context and a suitable MCS is almost straightforward.

## References


1. M. Benerecetti, P. Bouquet, and C. Ghidini. Formalizing belief report – the approach and a case study. In F. Giunchiglia, editor, *Artificial Intelligence: Methodology, Systems, and Applications (AIMSA'98)*, volume 1480 of *Lecture Notes in Artificial Intelligence*, pages 62–75. Springer, 1998.
2. M. Benerecetti, F. Giunchiglia, and L. Serafini. Model Checking Multiagent Systems. *Journal of Logic and Computation, Special Issue on Computational & Logical Aspects of Multi-Agent Systems*, 8(3):401–423, 1998.
3. P. Bouquet and F. Giunchiglia. Reasoning about Theory Adequacy: A New Solution to the Qualification Problem. *Fundamenta Informaticae*, 23(2–4):247–262, June,July,August 1995.
4. S. Buvač and Ian A. Mason. Propositional logic of context. In R. Fikes and W. Lehnert, editors, *Proc. of the 11th National Conference on Artificial Intelligence*, pages 412–419, Menlo Park, California, 1993. AAAI Press.
5. Saša Buvač. Quantificational logic of context. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, 1996.
6. A. Cimatti and L. Serafini. Multi-Agent Reasoning with Belief Contexts III: Towards the Mechanization. In P. Brezillon and S. Abu-Hakima, editors, *Proc. of the IJCAI-95 Workshop on "Modelling Context in Knowledge Representation and Reasoning"*, pages 35–45, 1995.
7. A. Cimatti and L. Serafini. Multi-Agent Reasoning with Belief Contexts: the Approach and a Case Study. In M. Wooldridge and N. R. Jennings, editors, *Intelligent Agents: Proceedings of 1994 Workshop on Agent Theories, Architectures, and Languages*, number 890 in Lecture Notes in Computer Science, pages 71–85. Springer Verlag, 1995.
8. A. Cimatti and L. Serafini. Multi-Agent Reasoning with Belief Contexts II: Elaboration Tolerance. In *Proc. 1st Int. Conference on Multi-Agent Systems (ICMAS-95)*, pages 57–64, 1996.
9. G. Criscuolo, F. Giunchiglia, and L. Serafini. A Foundation for Metareasoning, Part I: The proof theory. Technical Report 0003-38, IRST, Trento, Italy, 2001. To appear in the Journal of Logic and Computation.

10. G. Criscuolo, F. Giunchiglia, and L. Serafini. A Foundation for Metareasoning, Part II: The model theory. Technical Report 0010-07, IRST, Trento, Italy, 2000. To appear in the Journal of Logic and Computation.
11. C. Ghidini and F. Giunchiglia. Local Models Semantics, or Contextual Reasoning = Locality + Compatibility. *Artificial Intelligence*, 127(2):221–259, 2001.
12. E. Giunchiglia and F. Giunchiglia. Ideal and real belief about belief. *Journal of Logic and Computation*, To appear in 2000.
13. F. Giunchiglia. Contextual reasoning. *Epistemologia, special issue on I Linguaggi e le Macchine*, XVI:345–364, 1993. Short version in Proceedings IJCAI'93 Workshop on Using Knowledge in its Context, Chambery, France, 1993, pp. 39–49. Also IRST-Technical Report 9211-20, IRST, Trento, Italy.
14. F. Giunchiglia and L. Serafini. Multilanguage hierarchical logics (or: how we can do without modal logics). *Artificial Intelligence*, 65:29–70, 1994.
15. J. McCarthy. Generality in Artificial Intelligence. *Communications of ACM*, 30(12):1030–1035, 1987. Also in V. Lifschitz (ed.), *Formalizing common sense: papers by John McCarthy*, Ablex Publ., 1990, pp. 226–236.
16. J. McCarthy. Notes on Formalizing Context. In *Proc. of the 13th International Joint Conference on Artificial Intelligence*, pages 555–560, Chambery, France, 1993.
17. J. McCarthy and S. Buvač. Formalizing Context (Expanded Notes). In A. Aliseda, R.J. van Glabbeek, and D. Westerståhl, editors, *Computing Natural Language*, volume 81 of *CSLI Lecture Notes*, pages 13–50. Center for the Study of Language and Information, Stanford University, 1998.
18. D. Prawitz. *Natural Deduction - A proof theoretical study.* Almquist and Wiksell, Stockholm, 1965.
19. L. Serafini. Quantificational logic of contexts revisited. Technical Report 0105-01, ITC-IRST, 2001.

## A Proof of theorems

*Proof. of Theorem 3.1* We prove the theorem by showing that each PLC-model $\mathfrak{M}$ can extended to a PLC-model $\mathfrak{M}_c$ with a complete vocabulary with the following property:

For any formula $\phi$ and context sequence $\overline{\kappa}$, such that $\mathsf{Vocab}(\phi, \overline{\kappa}) \in \mathsf{Vocab}(\mathfrak{M})$, $\mathfrak{M} \models_{\overline{\kappa}} \phi$ iff $\mathfrak{M}_c \models_{\overline{\kappa}} \phi$ (13)

The *completion* of a PLC-model $\mathfrak{M}$, is the PLC-model $\mathfrak{M}_c$ defined as follows. For any $\overline{\kappa} \in \mathbb{K}^*$:

- if $\mathfrak{M}(\overline{\kappa})$ is undefined, then $\mathfrak{M}_c(\overline{\kappa})$ contains all the possible total assignments to $\mathbb{P}$.
- if $\mathfrak{M}(\overline{\kappa})$ is defined, then $\mathfrak{M}_c(\overline{\kappa})$ is the following set of assignments:

$$\left\{ \nu_c : \mathbb{P} \rightarrow \{\text{true}, \text{false}\} \;\middle|\; \begin{array}{l} \nu_c \text{ is a completion of} \\ \text{some assignment } \nu \in \\ \mathfrak{M}(\overline{\kappa}) \end{array} \right\}$$

where $\nu_c$ is a *completion* of $\nu$ if and only if $\nu_c$ agree with $\nu$ on the domain of $\nu$.

Clearly $\mathfrak{M}_c$ is a PLC-model. To prove property (13) we show by induction on the complexity of $\phi$, that for any assignment $\nu \in \mathfrak{M}(\overline{\kappa})$, and for any completion $\nu_c$ of $\nu$ in $\mathfrak{M}_c$:

$$\mathfrak{M}, \nu \models_{\overline{\kappa}} \phi \text{ iff } \mathfrak{M}_c, \nu_c \models_{\overline{\kappa}} \phi$$

*Basis case:* $\mathfrak{M}, \nu \models_{\overline{\kappa}} p$ iff $\nu(p) = \text{true}$, since any extension of $\nu_c$ agree with $\nu$ on its domain, then $\nu_c(p) = \text{true}$.

*Inductive case:* $\mathfrak{M}, \nu \models_{\overline{\kappa}} \neg\phi$ iff not $\mathfrak{M}, \nu \models_{\overline{\kappa}} \phi$, iff, by induction, not $\mathfrak{M}_c, \nu_c \models_{\overline{\kappa}} \phi$, iff $\mathfrak{M}_c, \nu_c \models_{\overline{\kappa}} \neg\phi$. The case of $\phi \supset \psi$ is similar. Let us consider the case of $ist(\kappa, \phi)$. $\mathfrak{M}, \nu \models_{\overline{\kappa}} ist(\kappa, \phi)$ iff for all $\nu' \in \mathfrak{M}(\overline{\kappa}\kappa)$, $\mathfrak{M}, \nu' \models_{\overline{\kappa}\kappa} \phi$, iff, by induction, for all $\nu'_c \in \mathfrak{M}_c(\overline{\kappa}\kappa)$, $\mathfrak{M}_c, \nu'_c \models_{\overline{\kappa}\kappa} \phi$, iff $\mathfrak{M}, \nu_c \models_{\overline{\kappa}} ist(\kappa, \phi)$. □

*Proof. of Theorem 3.3* Suppose that $\mathbf{c} \models \langle ist(\kappa, ist(\kappa', \phi) \vee \psi),\ \overline{\kappa}\rangle$. If for all $\mathbf{c}'$, with $c_{\overline{\kappa}} = c'_{\overline{\kappa}}$, we have that $\mathbf{c}' \models \langle \psi,\ \overline{\kappa}\kappa\rangle$, then by condition 3 of Definition 3.3 of MBK($\mathbb{K}^*$)-model, we have that $\mathbf{c} \models \langle ist(\kappa, \psi),\ \overline{\kappa}\rangle$ and therefore that $\mathbf{c} \models \langle ist(\kappa, ist(\kappa', \phi)) \vee ist(\kappa, \psi),\ \overline{\kappa}\rangle$. If there is such a $\mathbf{c}'$, such that $\mathbf{c}' \not\models \langle \psi,\ \overline{\kappa}\kappa\rangle$, from the fact that, by condition 2 of Definition 3.3 of MBK($\mathbb{K}^*$)-model $\mathbf{c}' \models \langle ist(\kappa', \phi) \vee \psi,\ \overline{\kappa}\kappa\rangle$, we have that $\mathbf{c}' \models \langle ist(\kappa', \phi),\ \overline{\kappa}\kappa\rangle$. By condition 4 of Definition 3.4 of MMCC-model, we have that $\mathbf{c}' \models \langle ist(\kappa, ist(\kappa', \phi)),\ \overline{\kappa}\rangle$. Since $c_{\overline{\kappa}} = c'_{\overline{\kappa}}$, then $\mathbf{c} \models \langle ist(\kappa, ist(\kappa', \phi)),\ \overline{\kappa}\rangle$, and therefore $\mathbf{c} \models \langle ist(\kappa, ist(\kappa', \phi)) \vee ist(\kappa, \psi),\ \overline{\kappa}\rangle$.

Vice-versa, suppose that $\mathbf{C} \models \langle(\Delta),\ \overline{\kappa}\rangle$ and let us prove condition 4 of Definition 3.4. Since the formula $\langle ist(\kappa, ist(\kappa', \phi) \vee \neg ist(\kappa', \phi)) \supset ist(\kappa, ist(\kappa', \phi)) \vee ist(\kappa, \neg ist(\kappa', \phi)),\ \overline{\kappa}\rangle$, is an instance of $(\Delta)$, and since $\mathbf{c} \models \langle ist(\kappa, ist(\kappa', \phi) \vee \neg ist(\kappa', \phi)),\ \overline{\kappa}\rangle$,

$$\mathbf{c} \models \langle ist(\kappa, ist(\kappa', \phi)) \vee ist(\kappa, \neg ist(\kappa', \phi)),\ \overline{\kappa}\rangle \tag{14}$$

Suppose that $\mathbf{c} \models \langle ist(\kappa', \phi),\ \overline{\kappa}\kappa\rangle$, then $\mathbf{c} \not\models \langle \neg ist(\kappa', \phi),\ \overline{\kappa}\kappa\rangle$, and by condition 2 of Definition 3.3, $\mathbf{c} \not\models \langle ist(\kappa, \neg ist(\kappa', \phi)),\ \overline{\kappa}\rangle$. By property (14), and by the fact that $|c_{\overline{\kappa}}| \leq 1$, we have that $\mathbf{c} \models \langle ist(\kappa, ist(\kappa', \phi)),\ \overline{\kappa}\rangle$. □

*Proof. of Theorem 3.4* To prove soundness it is enough to prove that the unrestricted version of $\mathcal{R}_{up}$, is sound w.r.t. logical consequence in MMCC-models. Namely that:

$$\langle ist(\kappa', \phi),\ \overline{\kappa}\kappa\rangle \models_{\mathrm{MMCC}} \langle ist(\kappa, ist(\kappa', \phi)),\ \overline{\kappa}\rangle$$

This is a trivial consequence of condition 4 of the definition of MMCC-model. Completeness of MMCC can be proved in an indirect way. We have indeed that MBK($\mathbb{K}^*$) is complete w.r.t. the class of MBK($\mathbb{K}^*$)-models. Furthermore, from Theorem 3.3, we have that, the class of MMCC-models, is the class of MBK($\mathbb{K}^*$)-models that satisfy $\langle(\Delta),\ \overline{\kappa}\rangle$. Completeness can be therefore proved by showing that $(\Delta)$ can be proved in MMCC. Figure 2, we show a deduction of the $(\Delta)$ Notationally, $Prem(\Delta)$ and $Cons(\Delta)$ denote the premise and the consequence of $(\Delta)$ respectively.

□

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\langle Prem(\Delta),\,\overline{\kappa}\rangle}{\langle ist(\kappa',\phi)\vee\psi,\,\overline{\kappa}\kappa\rangle}\,\mathcal{R}_{dn\overline{\kappa}\kappa} \qquad \langle\neg\psi,\,\overline{\kappa}\kappa\rangle}{\langle ist(\kappa',\phi),\,\overline{\kappa}\kappa\rangle}}{\langle ist(\kappa, ist(\kappa',\phi)),\,\overline{\kappa}\rangle}\,\mathcal{R}_{up\overline{\kappa}\kappa}}{\langle Cons(\Delta),\,\overline{\kappa}\rangle}\,\vee\mathrm{I}_{\overline{\kappa}} \qquad \langle\neg Cons(\Delta),\,\overline{\kappa}\rangle}{\langle\bot,\,\overline{\kappa}\rangle}\,\supset\mathrm{E}_{\overline{\kappa}}}{\langle ist(\kappa,\bot),\,\overline{\kappa}\rangle}\,\bot}{\langle\bot,\,\overline{\kappa}\kappa\rangle}\,\mathcal{R}_{dn\overline{\kappa}\kappa}}{\langle\psi,\,\overline{\kappa}\kappa\rangle}\,\bot}{\langle ist(\kappa,\psi),\,\overline{\kappa}\rangle}\,\mathcal{R}_{up\overline{\kappa}\kappa}}{\langle Cons(\Delta),\,\overline{\kappa}\rangle}\,\vee\mathrm{I}_{\overline{\kappa}} \qquad \langle\neg Cons(\Delta),\,\overline{\kappa}\rangle}{\langle\bot,\,\overline{\kappa}\rangle}\,\supset\mathrm{E}_{\overline{\kappa}}}{\langle Cons(\Delta),\,\overline{\kappa}\rangle}\,\bot}{\langle Prem(\Delta)\supset Cons(\Delta),\,\overline{\kappa}\rangle}\,\supset\mathrm{I}
$$

**Fig. 2.** A proof of $\Delta$ in MMCC

*Proof. of Theorem 3.5* Provability in PLC can be defined as provability in multi modal $K$ (denoted by $\vdash_K$) plus the axiom $(\Delta)$. For any subset $\mathbb{H}$ of $\mathbb{K}^*$, the notation $ist(\mathbb{H},\phi)$ denotes the set of formulas:

$$ist(\mathbb{H},\phi) = \{ist(k_1, ist(k_2, \ldots ist(k_n,\phi))) \mid \kappa_1\kappa_2\ldots\kappa_n \in \mathbb{H}\}$$

For any finite set of formulas $\Gamma = \{\gamma_1,\ldots,\gamma_n\}$, $\bigwedge\Gamma$ denotes the formula $\gamma_1 \wedge \ldots \wedge \gamma_n$, If $\vdash_{\overline{\kappa}} \phi$, then there is a *finite* set $\mathbb{H} \subseteq \mathbb{K}^*$, such that

$$\vdash_{\mathrm{K}} \bigwedge ist(\mathbb{H},(\Delta)) \supset \phi$$

From the equivalence between multi modal $K$ and $\mathrm{MBK}(\mathbb{K}^*)$ we have that

$$\vdash_{\mathrm{MBK}(\mathbb{K}^*)} \langle \bigwedge ist(\mathbb{H},(\Delta)) \supset \phi,\, \overline{\kappa}\rangle$$

Since any formula in $\langle ist(\mathbb{H},(\Delta)),\,\overline{\kappa}\rangle$ is provable in MMCC, then we can conclude that

$$\vdash_{\mathrm{MMCC}} \langle\phi,\,\overline{\kappa}\rangle$$

If $\not\vdash_{\overline{\kappa}} \phi$, then we have that $\not\vdash_{\epsilon} \phi$, (where $\epsilon$ is the empty sequence). This implies that there is a model $\mathfrak{M}$, such that $\mathfrak{M} \not\models_{\epsilon} \phi$. We define the MMCC-model $\mathbf{C}_{\mathfrak{M}}$, that contains all the sequences $\mathbf{c}$ such that $c_{\overline{\kappa}} \in \mathfrak{M}(\overline{\kappa})$, and $c_{\overline{\kappa}}$ is empty if $\mathfrak{M}$ is not defined for some $\overline{\kappa}'$, such that $\overline{\kappa} = \overline{\kappa}'\overline{\kappa}''$. It can be easily show that $\mathbf{C}_{\mathfrak{M}}$ is a MMCC-model, and that $\mathbf{C}_{\mathfrak{M}} \not\models \langle\phi,\,\epsilon\rangle$. □