Available online at www.sciencedirect.com

ScienceDirect

Electronic Notes in Theoretical Computer Science 305 (2014) 85–102

Electronic Notes in Theoretical Computer Science

www.elsevier.com/locate/entcs

Towards Resolution-based Reasoning for Connected Logics

Cláudia Nalon[1,2]

Department of Computer Science
University of Brasília, Brazil

Oliver Kutz[3]

Institute of Knowledge and Language Engineering
Otto-von-Guericke University Magdeburg, Germany

Abstract

The method of connecting logics has gained a lot of attention in the knowledge representation and ontology communities because of its intuitive semantics and natural support for modular KR, its generality, and its robustness concerning decidability preservation. However, so far no dedicated automated reasoning solutions have been developed, and the only reasoning available was via translation into sufficiently expressive logics. In this paper, we present a simple modalised version of basic $\mathcal{E}$-connections, and develop a sound, complete, and terminating resolution-based reasoning procedure. The approach is modular and can be extended to more expressive versions of $\mathcal{E}$-connections.

Keywords: $\mathcal{E}$-connections, normal modal logics, theorem-proving, resolution method, bridge principles.

1 Introduction

Modal and other non-classical logics have been developed in a great variety to address various modelling requirements, be it spatial, temporal, deontic, etc. However, special purpose formalisms are often difficult to extend, and methodologies for *combining* logics into many dimensional formalisms have therefore been studied

[1] Cláudia Nalon was supported by CAPES Foundation BEX 8712/11-5. The work of Oliver Kutz was supported by the DFG funded Research Center on Spatial Cognition SFB/TR 8, and the COINVENT project funded through the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open Grant number: 611553.

[2] Email: `nalon@unb.br`

[3] Email: `okutz@iws.cs.ovgu.de`

http://dx.doi.org/10.1016/j.entcs.2014.06.007

extensively, in particular techniques such as products [8], fusions [12], or fibrings [5], as well as structuring techniques for heterogeneous logical theories [16].

The method of *connecting*, or $\mathcal{E}$*-connecting* logics, in particular, has gained a lot of attention in the knowledge representation and ontology communities because of its intuitive semantics (being closely related to counterpart theory [11]) and natural support for modular KR [7], its generality, and its robustness concerning decidability preservation [1,14]. However, so far no dedicated automated reasoning solutions have been developed, and the only reasoning available was via translation into sufficiently expressive logics [6,16]. The $\mathcal{E}$-connection method is closely related to Distributed Description Logics (DDLs) [4], for which a method of distributed tableaux has been developed [23]. Nevertheless, the standard DDL language is strictly less expressive than $\mathcal{E}$-connections, as shown in [14]. The main difference, in a nutshell, is that whilst DDLs only provide atomic formulae for linking two logics, $\mathcal{E}$-connections allow to build new 'complex concepts' in the components, using DL terminology. In modal logic terms, it means they allow to construct new formulae using modalities from foreign logics along bridge modalities. Here, bridge modalities belong to neither of the component logics, but are interpreted with the help of new accessibility relations that are two-sorted in nature, i.e. go across respective models interpreting the component languages. This also explains the generality of the approach compared to DDLs, as the logic of these bridge modalities can be freely varied.

Compared to other combination methodologies, an interesting aspect of $\mathcal{E}$-connections is that, unlike products, no so-called bridge principles are introduced by the combination method itself. An example would be $\Box$-commutativity or Church-Rosser properties automatically being validated in products, see [5]. This is avoided in $\mathcal{E}$-connections because the languages are kept disjoint, and are being connected only via the bridge modalities—bridge principles therefore only arise explicitly when added as new axioms.

In this paper, we present a simple modalised version of basic $\mathcal{E}$-connections and a sound, complete, and terminating resolution-based reasoning procedure for dealing with this kind of combination. We note that $\mathcal{E}$-connections have been widely applied to combining Description Logics [13,7] and that experimental evaluation shows that resolution performs well for such logics [10,20]. The reasoning procedure we introduce here is very simple in its structure, keeping the calculi for the component logics disjoint, and introducing a set of resolution-based inference rules that extend the method in [19] to solve the satisfiability problem in logics connecting **K**-components via **K**-bridge modalities. The approach is modular and can be extended to more expressive versions of $\mathcal{E}$-connections.

The paper is structured as follows. In Section 2, we present the syntax and semantics of the multi-modal logic $\mathsf{K}_{(n)}$. In Section 3, a basic modalised version of $\mathcal{E}$-connections is defined. Section 4 introduces the resolution method for connected logics, together with examples, and Section 5 sketches the correctness proofs. Results and related work are discussed in Section 6.

2 The Normal Modal Logic $\mathsf{K}_{(n)}$

The weakest of the normal modal systems, known as $\mathsf{K}_{(n)}$, is an extension of the classical propositional logic with the operators $\boxed{a}$, $1 \leq a \leq n$, where the axioms $\mathbf{K}_a$, that is, $\boxed{a}(\varphi \to \psi) \to (\boxed{a}\varphi \to \boxed{a}\psi)$, hold. There is no restriction on the accessibility relation over worlds. As the subscript in $\mathsf{K}_{(n)}$ indicates, we consider the multi-agent version, that is, the fusion of several copies of $\mathsf{K}_{(1)}$.

Formulae are constructed from a denumerable set of **propositional symbols** (or **variables**), $\mathcal{P} = \{p, q, p', q', p_1, q_1, \ldots\}$. The finite **set of agents** is defined as $\mathcal{A} = \{1, \ldots, n\}$, $n \in \mathbb{N}$. Besides the propositional connectives ($\mathbf{true}, \neg, \wedge$), we introduce a set of unary modal operators $\mathcal{M} = \{\boxed{1}, \ldots, \boxed{n}\}$, where $\boxed{a}\varphi$ is read as "agent a considers φ necessary". When $n = 1$, we may omit the index, that is, $\Box\varphi = \boxed{1}\varphi$. The fact that agent a considers φ possible is denoted by $\langle a \rangle\varphi$. The **language** $\mathcal{L}$ of $\mathsf{K}_{(n)}$ is given by $\mathcal{L} = \mathcal{P} \cup \mathcal{M} \cup \{\mathbf{true}, \neg, \wedge\}$. Next, we define the set of well-formed formulae of $\mathsf{K}_{(n)}$:

Definition 2.1 The **set of well-formed formulae**, $\mathcal{F}(\mathsf{K}_{(n)})$, is given by:

- the propositional symbols are in $\mathcal{F}(\mathsf{K}_{(n)})$;
- **true** is in $\mathcal{F}(\mathsf{K}_{(n)})$;
- if φ and ψ are in $\mathcal{F}(\mathsf{K}_{(n)})$, then so are $\neg\varphi$, $(\varphi \wedge \psi)$, and $\boxed{a}\varphi$ (for all $a \in \mathcal{A}$).

A **literal** is either a proposition or its negation. Let $\mathcal{L}it$ be the set of all literals. A **modal literal** is either $\boxed{a}l$ or $\neg\boxed{a}l$, where $l \in \mathcal{L}it$ and $a \in \mathcal{A}$.

Semantics of $\mathsf{K}_{(n)}$ is given, as usual, in terms of a Kripke structure.

Definition 2.2 A **Kripke structure** $\mathbb{M}$ **for** n **agents over** $\mathcal{P}$ is a tuple $\mathbb{M} = \langle \mathcal{W}, w_0, \pi, \mathcal{R}_1, \ldots, \mathcal{R}_n \rangle$, where $\mathcal{W}$ is a set of possible *worlds* (or *states*) with a distinguished world w_0 ; the function $\pi(w) : \mathcal{P} \to \{\mathit{true}, \mathit{false}\}$, $w \in \mathcal{W}$, is an interpretation that associates with each state in $\mathcal{W}$ a truth assignment to propositions; and each $\mathcal{R}_a \subseteq \mathcal{W} \times \mathcal{W}$ is a binary relation on $\mathcal{W}$.

The binary relation $\mathcal{R}_a$ captures the possibility relation according to agent a: a pair (w, w') is in $\mathcal{R}_a$ if agent a considers world w' possible, given her information in world w. We write $(\mathbb{M}, w) \models \varphi$ to say that φ is true at world w in the Kripke structure $\mathbb{M}$.

Definition 2.3 Truth of a formula is given as follows:

- $(\mathbb{M}, w) \models \mathbf{true}$
- $(\mathbb{M}, w) \models p$ if, and only if, $\pi(w)(p) = \mathit{true}$, where $p \in \mathcal{P}$
- $(\mathbb{M}, w) \models \neg\varphi$ if, and only if, $(\mathbb{M}, w) \not\models \varphi$
- $(\mathbb{M}, w) \models (\varphi \wedge \psi)$ if, and only if, $(\mathbb{M}, w) \models \varphi$ and $(\mathbb{M}, w) \models \psi$
- $(\mathbb{M}, w) \models \boxed{a}\varphi$ if, and only if, for all w', such that $(w, w') \in \mathcal{R}_a$, $(\mathbb{M}, w') \models \varphi$.

The formulae **false**, $(\varphi \vee \psi)$, $(\varphi \to \psi)$, and $\langle a \rangle\varphi$ are introduced as the usual abbreviations for $\neg\mathbf{true}$, $\neg(\neg\varphi \wedge \neg\psi)$, $(\neg\varphi \vee \psi)$, and $\neg\boxed{a}\neg\varphi$, respectively. Formulae are interpreted with respect to the distinguished world w_0. Intuitively, w_0 is the world from which we start reasoning. Let $\mathbb{M} = \langle \mathcal{W}, w_0, \pi, \mathcal{R}_1, \ldots, \mathcal{R}_n \rangle$ be a Kripke structure with a distinguished world w_0. Thus, a formula φ is said to be **satisfiable in** $\mathbb{M}$ if $(\mathbb{M}, w_0) \models \varphi$; φ is said to be **satisfiable** if there is a model $\mathbb{M}$ such that $(\mathbb{M}, w_0) \models \varphi$; and φ is said to be **valid** if for all models $\mathbb{M}$ then $(\mathbb{M}, w_0) \models \varphi$. Satisfiability of sets is defined as usual. A finite set $\Gamma \subset \mathcal{F}(\mathsf{K}_{(n)})$ is **satisfiable at the state** w **in** $\mathbb{M}$, denoted by $(\mathbb{M}, w) \models \Gamma$, if $(\mathbb{M}, w) \models \gamma_0 \wedge \ldots \wedge \gamma_n$, for all $\gamma_i \in \Gamma$, $0 \leq i \leq n$; Γ is **satisfiable in a model** $\mathbb{M}$, denoted by $\mathbb{M} \models \Gamma$, if $(\mathbb{M}, w_0) \models \Gamma$; and Γ is **satisfiable**, if there is a model $\mathbb{M}$ such that $\mathbb{M} \models \Gamma$.

3 Modalising Connections

In this section, we present a basic modalised version of $\mathcal{E}$-connections. For the purpose of illustrating our resolution-based calculus, it will suffice to introduce connections of normal modal operators with **K**-like bridge operators.

$\mathcal{E}$-connections were originally conceived as a versatile and well-behaved technique for combining logics [14], but have been quickly adopted as a framework for the integration of ontologies and modular reasoning in the Semantic Web [7]. The general idea behind this combination method is that the interpretation domains of the connected logics are interpreted by disjoint vocabulary and interconnected by means of *link relations*. The language of the $\mathcal{E}$-connection is then the union of the original languages enriched with operators capable of talking about the link relations.

The most important feature of $\mathcal{E}$-connections is that they offer an appealing compromise between expressive power and computational complexity: although powerful enough to express many interesting concepts, the coupling between the combined logics is sufficiently loose for proving general results about the transfer of decidability. Such transfer results state that if the connected logics are decidable, then their connection (under certain restrictions) will also be decidable.

The generality of the transfer results for $\mathcal{E}$-connections obtained in [14] is due to the fact that $\mathcal{E}$-connections are defined and investigated using the framework of so-called *abstract description systems* (ADSs), a common generalisation of description logics, modal logics, logics of time and space, and many other logical formalisms [2]. Thus, we can connect not only DLs with DLs, but also, say, description logics with spatial logics (and in the general case n ADSs for any $n \in \mathbb{N}$). A natural interpretation of link relations in this context would then be, for instance, to describe the spatial extension of abstract (DL) objects. Indeed, the idea of $\mathcal{E}$-connections was first described for this scenario in [17], and the '$\mathcal{E}$' in $\mathcal{E}$-connections stems from the concept of 'spatial **E**xtension'. Several extensions to the basic $\mathcal{E}$-connection language have been studied in [14], including Booleans on links, number restrictions on links, link operators on object names, and first-order link constraints.

Let L_1 and L_2 be two normal multi-modal logics that are to be connected. We assume that the **languages** $\mathcal{L}_1$ and $\mathcal{L}_2$, i.e., the propositional variables and modal operators of L_1 and L_2, are pairwise disjoint; however, for simplicity of presentation we here identify the (classical) Boolean operators of $\mathcal{L}_1$ and $\mathcal{L}_2$. Note that separating propositional connectives only becomes relevant when connecting logics with a non-classical propositional base logic, e.g. when combining intuitionistic logic with classical or relevant logic, a topic which we intend to follow up on in future work.

To define the language of a modal connection, we need to fix additionally the sets of **bridge modalities** given by

$$\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2, \text{ with } \mathcal{M}_1 = \{ \Diamond_{j}^{1} \mid j \in \mathcal{I}_1\}, \quad \mathcal{M}_2 = \{ \Diamond_{k}^{2} \mid k \in \mathcal{I}_2\}$$

where both $\mathcal{I}_1$, $\mathcal{I}_2$ are countable, non-empty index sets.

A similar language was sketched in [6], called **one-way** $\mathcal{E}$-connections, presenting a formulation of $\mathcal{E}$-connections in DL syntax and removing the build-in inverse relations of [14] in order to allow for a better comparison to DDLs [4].

The set of **formulae** of the **basic modal connection language** $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$ is a two-sorted language partitioned into a set of 1-formulae and a set of 2-formulae. In the following, we set $\overline{1} = 2$ and $\overline{2} = 1$ and denote by $|\mathcal{S}|$ the cardinality of a set $\mathcal{S}$. Intuitively, i-formulae are the formulae of $\mathcal{L}_i$ enriched with new modalities for talking about $\overline{i}$-formulae accessed via linking accessibility relations. We often refer to a connection $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$ simply as $\mathcal{C}^{\mathcal{M}}$ once the L_i have been fixed.

Definition 3.1 [Modal Connection Language] The sets of 1-*formulae* and *2-formulae* of $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$ are defined by simultaneous induction, for $i \in \{1, 2\}$:

(1) every propositional variable of $\mathcal{L}_i$ is an i-formula;

(2) i-formulae are closed under Boolean operators and the modalities of $\mathcal{L}_i$;

(3.1) if φ is a 1-formula and $k \in \mathcal{I}_2$, then $\Diamond_{k}^{2}\varphi$ is a 2-formula.

(3.2) if ψ is a 2-formula and $j \in \mathcal{I}_1$, then $\Diamond_{j}^{1}\psi$ is a 1-formula.

The set of i-**formulae** of $\mathcal{C}^{\mathcal{M}}$ is denoted by $\mathcal{F}(\mathcal{C}^{\mathcal{M}})^i$, $i = 1, 2$. The set of **formulae** of $\mathcal{C}^{\mathcal{M}}$ is $\mathcal{F}(\mathcal{C}^{\mathcal{M}})^1 \cup \mathcal{F}(\mathcal{C}^{\mathcal{M}})^2$. A **theory** in $\mathcal{C}^{\mathcal{M}}$ is a pair $\Gamma = \langle \Gamma_1, \Gamma_2 \rangle$, where Γ_i, $i = 1, 2$, are finite sets of i-formulae. $\mathcal{C}^{\mathcal{M}}$ is called **finitely linked** if $|\mathcal{I}_1|, |\mathcal{I}_2| \in \mathbb{N}$, and **unarily linked** if $|\mathcal{I}_1| = |\mathcal{I}_2| = 1$.

Example 3.2 [Well-formed expressions] To illustrate the language just defined, we give a number of well-formed expressions. Let φ_1, φ_2 be formulae of L_1, and ψ_1, ψ_2 formulae of L_2. The following are well-formed

$$\Diamond_{h}^{1} \neg \Diamond_{j}^{2} \varphi_1 \to \neg \Diamond_{k}^{1} \psi_1 \qquad \varphi_1 \wedge \neg \Diamond_{h}^{1} (\psi_1 \vee \psi_2 \vee \Diamond_{l}^{2} \varphi_2)$$

In contrast, the following expressions are ill-formed:

$$\Diamond_{h}^{1} \neg \Diamond_{j}^{2} \varphi_1 \to \neg \Diamond_{k}^{2} \varphi_1 \qquad \varphi_1 \wedge \neg \Diamond_{h}^{1} \varphi_1$$

because, in the first case, we are forming a Boolean combination of a 1-formula with a 2-formula, which is undefined, and in the second case, we apply a 1-modality to a 1-formula, which is undefined.

Given the language of a connection $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$, a **connected Kripke model** for a modal connection $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$ consists of a Kripke model for L_1, a Kripke model for L_2, and an interpretation of a set $\mathcal{E}$ of link relations associated with the bridge modalities. The details of the semantics are as follows:

Definition 3.3 [Connected Kripke Models] A structure

$$\mathbb{M} = \left\langle \mathbb{W}_1, \mathbb{W}_2, (E^1_j)_{j \in \mathcal{I}_1}, (E^2_k)_{k \in \mathcal{I}_2} \right\rangle$$

where $\mathbb{W}_i = \langle \mathcal{W}_i, w^i_0, \pi_i, \mathcal{R}^i_1, \ldots, \mathcal{R}^i_n \rangle$ (as defined in Def. 2.2) is a Kripke model of L_i for $i \in \{1, 2\}$, and $E^1_j \subseteq \mathcal{W}_1 \times \mathcal{W}_2$ for each $j \in \mathcal{I}_1$ and $E^2_k \subseteq \mathcal{W}_2 \times \mathcal{W}_1$ for each $k \in \mathcal{I}_2$, is called a **connected Kripke model** for $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$. The members of the set

$$\mathcal{E} = \mathcal{E}_1 \cup \mathcal{E}_2, \text{ with } \mathcal{E}_1 = \{E^1_j \mid j \in \mathcal{I}_1\}, \quad \mathcal{E}_2 = \{E^2_k \mid k \in \mathcal{I}_2\}$$

are called **link relations**.

Satisfaction of an i-formula χ at a world w of L_i is defined by simultaneous induction. The Booleans and local modalities of logic L_i are defined in the standard way (as given earlier). The remaining cases are as follows. Let φ be a 1-formula, and ψ be a 2-formula, w_1 a world in $\mathcal{W}_1$, w_2 a world in $\mathcal{W}_2$:

$$(\mathbb{M}, w_1) \models \diamondsuit\!\!\!\!\;j^{\,1}\psi \iff \exists w_2 \in \mathcal{W}_2 \text{ such that } w_1 E^1_j w_2 \text{ and } (\mathbb{M}, w_2) \models \psi$$
$$(\mathbb{M}, w_2) \models \diamondsuit\!\!\!\!\;k^{\,2}\varphi \iff \exists w_1 \in \mathcal{W}_1 \text{ such that } w_2 E^2_k w_1 \text{ and } (\mathbb{M}, w_1) \models \varphi$$

Example 3.4 [Normality and Bridge Logic] Define bridge boxes by setting:

$$\boxed{j}^{\,1} := \neg \diamondsuit\!\!\!\!\;j^{\,1} \neg \qquad \boxed{k}^{\,2} := \neg \diamondsuit\!\!\!\!\;k^{\,2} \neg$$

Then, for any connected Kripke model $\mathbb{M}$ and any worlds w_1, w_2, we have

$$(\mathbb{M}, w_1) \models \boxed{j}^{\,1}(\varphi \to \psi) \to (\boxed{j}^{\,1}\varphi \to \boxed{j}^{\,1}\psi)$$

$$(\mathbb{M}, w_2) \models \boxed{k}^{\,2}(\chi \to \tau) \to (\boxed{k}^{\,2}\chi \to \boxed{k}^{\,2}\tau)$$

The proof is given in Example 4.2. Moreover, it is easy to see that bridge modalities satisfy the rule of Necessitation. Indeed, this already gives a complete axiomatisation of the basic bridge modalities introduced in Def. 3.3 (as hinted at already in [1]) and shows that they are **K**-modalities, illustrating that the connection method does not, by itself, introduce additional bridge principles.

Example 3.5 [Inverse Relations] In general connected Kripke structures, the relations in $\mathcal{E}_1$ and $\mathcal{E}_2$ are completely independent. In DLs, inverse (or converse) relations are of great importance in modelling, and they were natively built into the (semantically given) standard definition of $\mathcal{E}$-connections. Here, E^1 is the inverse of E^2 if for all x, y we have: $\langle x, y\rangle \in E^1 \iff \langle y, x\rangle \in E^2$. However, this is unnecessary: it is folklore in temporal logic that inverse modalities can be easily axiomatised (see [9]).

Consider the following theory T in $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$, where the L_i denote two arbitrary propositional modal logics, and p is a 1-variable, $j \in I_1$, and q is a 2-variable, $k \in I_2$.

$$T = \{p \to \boxed{j}^{1} \langle k \rangle^{2} p, \quad q \to \boxed{k}^{2} \langle j \rangle^{1} q\}$$

We claim that T is valid in $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$ if, and only if, E^1_j is the inverse of E^2_k.

A proof is obtained by mimicking the proof for monomodal logic given in [22, Theorem 1]. A sketch is as follows: (i) the validity of T is immediate if we assume that E^1_j is the inverse of E^2_k; (ii) conversely, assume that T is valid in $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$. Assume $\langle w_1, w_2\rangle \in E^1_j$ for a model where $w_1 \models p$ and p is false everywhere else in $\mathcal{W}_1$. Since $w_1 \models p \to \boxed{j}^{1} \langle k \rangle^{2} p$, it follows that $w_2 \models \langle k \rangle^{2} p$, i.e. there is a $w_3 \in \mathcal{W}_1$ such that $\langle w_2, w_3\rangle \in E^2_k$ and $w_3 \models p$. But, by the definition of the model, it follows that $w_1 = w_3$, which means that $E^1_j \subseteq (E^2_k)^{-1}$. The other inclusion is obtained in a similar way using the second axiom.

4 The Bridge Calculus

In this section we present the resolution-based calculus for $\mathcal{C}^{\mathcal{M}}$, $\mathsf{RES}_{\mathcal{E}}$. The approach is clausal: a formula to be tested for (un)satisfiability is firstly translated into a normal form, given in Section 4.1, and then the inference rules given in Section 4.2 are applied until either a contradiction is found or no new clauses can be generated. The calculus incorporates inference rules to deal with each of the component logics, which are syntactical variants of the inference rules given in [19], and also inference rules to deal with the connections between these components.

In the following, let L_1 and L_2 be two normal multi-modal logics, where the set of propositional symbols and modal operators in $\mathcal{L}_1$ and $\mathcal{L}_2$ are pairwise disjoint. Let $\{\boxed{a}^{i} \mid a \in \mathcal{A}_i\}$, with $\mathcal{A}_i = \{1, \ldots, n_i\}$, $i = 1, 2$, $n_i \in \mathbb{N}$, be the set of modalities in the language of L_i. Let $\mathcal{C}^{\mathcal{M}}$ be the language of the connection between L_1 and L_2, where the set of connecting modalities is given by $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2$ with $\mathcal{M}_1 = \{\langle j \rangle^{1} \mid j \in \mathcal{I}_1\}$ and $\mathcal{M}_2 = \{\langle k \rangle^{2} \mid k \in \mathcal{I}_2\}$, where both $\mathcal{I}_1$, $\mathcal{I}_2$ are countable, non-empty index sets. Let $\mathbb{M} = \langle \mathbb{W}_1, \mathbb{W}_2, (E^1_j)_{j \in \mathcal{I}_1}, (E^2_k)_{k \in \mathcal{I}_2}\rangle$ be the connected Kripke model for $\mathcal{C}^{\mathcal{M}}(\mathsf{L}_1, \mathsf{L}_2)$, where $\mathbb{W}_i = \langle W_i, w^i_0, \pi_i, \mathcal{R}^i_1, \ldots, \mathcal{R}^i_{n_i}\rangle$ is the underlying model for L_i and w^i_0 is the distinguished world in W_i.

4.1 The Normal Form for Connected Logics

Formulae in the language of $\mathcal{C}^{\mathcal{M}}$ can be transformed into a normal form called **Separated Normal Form for Connected Logics**, $\mathsf{SNF}_{\mathcal{E}}$. A formula to be tested for satisfiability is firstly translated into a $\mathcal{C}^{\mathcal{M}}$-problem, given by $\mathbb{C} = \langle \mathbb{C}_1, \mathbb{C}_2\rangle$ where each $\mathbb{C}_i$, $i = 1, 2$, is a tuple $\mathbb{C}_i = \langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i\rangle$, where $\mathbb{M} \models \mathbb{C}_i$ if and only if $(\mathbb{M}, w^i_0) \models \mathcal{S}_i$ and $(\mathbb{M}, w) \models \mathcal{G}_i \cup \mathcal{K}_i$, for all $w \in W_i$. Also, $\mathbb{M} \models \mathbb{C}$ if and only if $\mathbb{M} \models \mathbb{C}_i$, $i = 1, 2$. Each $\mathbb{C}_i$ of a $\mathcal{C}^{\mathcal{M}}$-problem is called a $\mathbb{C}_i$-problem. Intuitively, a $\mathbb{C}_i$-problem contains formulae which are in $\mathcal{F}(\mathcal{C}^{\mathcal{M}})^i$. As before, we set $\overline{1} = 2$ and $\overline{2} = 1$. Thus, in the following, $\mathbb{C}_{\overline{1}}$ (resp. $\mathbb{C}_{\overline{2}}$) stands for $\mathbb{C}_2$ (resp. $\mathbb{C}_1$).

Recall that the semantics in each component of the connected logic is given with respect to a pointed-model, that is, satisfiability is defined in terms of the distinguished world w_0^i within each component logic. Therefore, in order to represent the world from which we start reasoning, we introduce the new constants $\mathbf{start}_i$, $i = 1, 2$, where $(\mathbb{M}, w) \models \mathbf{start}_i$ if, and only if, $w = w_0^i$.

In order to apply the resolution method to a problem, we further require that the formulae in $\mathcal{S}_i$ are initial clauses; the formulae in $\mathcal{G}_i$ are literal clauses; and the formulae in $\mathcal{K}_i$ are modal clauses. That is, they have the following syntactic form, for each component logic L_i:

$$
\begin{array}{ll}
\text{Initial clause} & \mathbf{start}_i \rightarrow \bigvee_{b=1}^{r} l_b \\
\text{Literal clause} & \mathbf{true} \rightarrow \bigvee_{b=1}^{r} l_b \\
\text{Positive } a\text{-clause} & l' \rightarrow \boxed{a}^{\,i} l \\
\text{Negative } a\text{-clause} & l' \rightarrow \neg \boxed{a}^{\,i} l \\
\text{Positive } \mathcal{E}_i^k\text{-clause} & l' \rightarrow \boxed{\boxed{k}}^{\,i} l \\
\text{Negative } \mathcal{E}_i^k\text{-clause} & l' \rightarrow \neg \boxed{\boxed{k}}^{\,i} l
\end{array}
$$

where l, l', and l_b are literals; $a \in \mathcal{A}_i$; and $k \in \mathcal{I}_i$. Positive and negative a-clauses (resp. positive and negative $\mathcal{E}_i^k$-clauses) are together known as **modal a-clauses** (resp. **modal $\mathcal{E}_i^k$-clauses**); the index may be omitted if it is clear from the context. Modal a-clauses and $\mathcal{E}_i^k$-clauses are referred as **modal clauses**.

The translation into the $\mathsf{SNF}_{\mathcal{E}}$ uses the renaming technique [21], where complex subformulae are replaced by new propositional symbols and the truth of these new symbols is linked to the formulae that they replaced in all states within the model corresponding to the component language we are dealing with. Classical operators are removed by rewriting.

Assume that a given formula φ to be tested for (un)satisfiability is an i-formula in Negated Normal Form (NNF), that is, a formula where implications are removed by classical rewriting and negations are applied to propositional symbols only. The transformation into the $\mathsf{SNF}_{\mathcal{E}}$, starts by taking the $\mathcal{C}^{\mathcal{M}}$-problem $\langle \mathbb{C}_1, \mathbb{C}_2 \rangle$, where $\mathbb{C}_i = \langle \{\mathbf{start}_i \rightarrow t\}, \{t \rightarrow \varphi\}, \emptyset \rangle$ and $\mathbb{C}_{\bar{i}} = \langle \emptyset, \emptyset, \emptyset \rangle$ where t is a new propositional symbol in $\mathcal{L}_i$. The transformation proceeds by applying the following rewrite rules together with the usual simplification rules for classical logic (where φ_1 and φ_2 are formulae in $\mathcal{F}(\mathcal{C}^{\mathcal{M}})^i$, t is a literal, t_1 is a new propositional symbol in the language of L_i, and $\mathbb{C}_i = \langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \rangle$):

$$\begin{aligned}
\langle \mathcal{S}_i, \mathcal{G}_i \cup \{t \to \varphi_1 \wedge \varphi_2\}, \mathcal{K}_i\rangle &\longrightarrow \langle \mathcal{S}_i, \mathcal{G}_i \cup \{t \to \varphi_1, t \to \varphi_2\}, \mathcal{K}_i\rangle \\
\langle \mathcal{S}_i, \mathcal{G}_i \cup \{t \to \varphi_1 \vee \varphi_2\}, \mathcal{K}_i\rangle &\longrightarrow \langle \mathcal{S}_i, \mathcal{G}_i \cup \{t \to \varphi_1 \vee t_1, t_1 \to \varphi_2\}, \mathcal{K}_i\rangle \\
&\qquad \text{where } \varphi_2 \text{ is not a disjunction of literals} \\
\langle \mathcal{S}_i, \mathcal{G}_i \cup \{t \to \varphi_1\}, \mathcal{K}_i\rangle &\longrightarrow \langle \mathcal{S}_i, \mathcal{G}_i \cup \{\mathbf{true} \to \neg t \vee \varphi_1\}, \mathcal{K}_i\rangle \\
&\qquad \text{where } \varphi_1 \text{ is a disjunction of literals or a constant} \\
\langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \langle a\rangle^i \varphi_1\}\rangle &\longrightarrow \langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \neg \boxed{a}^i \neg \varphi_1\}\rangle
\end{aligned}$$

The next rule moves modal clauses to the appropriate set, where φ_1 is either of the form $\boxed{a}^i\psi$, $\neg\boxed{a}^i\psi$, $\boxed{\boxed{k}}^i\psi$, $\neg\boxed{\boxed{k}}^i\psi$, with $a \in \mathcal{A}_i$ and $k \in \mathcal{I}_i$:

$$\langle \mathcal{S}_i, \mathcal{G}_i \cup \{t \to \varphi_1\}, \mathcal{K}_i\rangle \longrightarrow \langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \varphi_1\}\rangle$$

We rename complex subformulae enclosed in a modal operator as follows, where t_1 is a new propositional symbol in $\mathcal{L}_i$ and $\varphi_1 \in \mathcal{F}(\mathcal{C}^{\mathcal{M}})^i$ is not a literal.

$$\begin{aligned}
\langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \boxed{a}^i \varphi_1\}\rangle &\longrightarrow \langle \mathcal{S}_i, \mathcal{G}_i \cup \{t_1 \to \varphi_1\}, \mathcal{K}_i \cup \{t \to \boxed{a}^i t_1\}\rangle \\
\langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \neg\boxed{a}^i \varphi_1\}\rangle &\longrightarrow \langle \mathcal{S}_i, \mathcal{G}_i \cup \{t_1 \to \neg\varphi_1\}, \mathcal{K}_i \cup \{t \to \neg\boxed{a}^i \neg t_1\}\rangle
\end{aligned}$$

We also rename complex subformulae enclosed in a connecting operator as follows, where t_1 is a new propositional symbol in the language of $\mathsf{L}_{\bar{i}}$ and $\varphi_1 \in \mathcal{F}(\mathcal{C}^{\mathcal{M}})^{\bar{i}}$.

$$\tau_{\mathbf{1}}: \left\langle \begin{array}{c} \langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \boxed{\boxed{a}}^i \varphi_1\}\rangle \\ \langle \mathcal{S}_{\bar{i}}, \mathcal{G}_{\bar{i}}, \mathcal{K}_{\bar{i}}\rangle \end{array} \right\rangle \longrightarrow \left\langle \begin{array}{c} \langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \boxed{\boxed{a}}^i t_1\}\rangle \\ \langle \mathcal{S}_{\bar{i}}, \mathcal{G}_{\bar{i}} \cup \{t_1 \to \varphi_1\}, \mathcal{K}_{\bar{i}}\rangle \end{array} \right\rangle$$

$$\tau_{\mathbf{2}}: \left\langle \begin{array}{c} \langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \neg\boxed{\boxed{a}}^i \varphi_1\}\rangle \\ \langle \mathcal{S}_{\bar{i}}, \mathcal{G}_{\bar{i}}, \mathcal{K}_{\bar{i}}\rangle \end{array} \right\rangle \longrightarrow \left\langle \begin{array}{c} \langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \to \neg\boxed{\boxed{a}}^i \neg t_1\}\rangle \\ \langle \mathcal{S}_{\bar{i}}, \mathcal{G}_{\bar{i}} \cup \{t_1 \to \neg\varphi_1\}, \mathcal{K}_{\bar{i}}\rangle \end{array} \right\rangle$$

Some care needs to be taken when applying the preceding rewrite rules in order to ensure that the translation is terminating. This can be easily done by keeping track of which clauses have already been rewritten and, as so, preventing the procedure of applying these rules twice to any $\mathcal{E}_i^k$-clauses. The proof that φ, the original formula, is satisfiable if, and only if, the problem $\mathbb{C}$ resulting from applying the rewrite rules above is satisfiable is similar to that in [19] and sketched in Section 5. Note that, at the end of the translation, each a-modal clause and each $\mathcal{E}_i^k$-modal clause contains only one modal literal. As so, the different contexts belonging to different agents and to different connecting modalities are already separated at the end of the translation. Separating such contexts helps in the design and implementation of the resolution calculus given in Section 4.2.

Example 4.1 [Transformation] Consider the following 1-formula:

$$\varphi = \boxed{}^1\boxed{}^2(p \to q) \wedge \boxed{}^1\boxed{}^2 p \wedge \Diamond^1\Diamond^2\neg q$$

where $\{p, q\}$ is in L_1. We start the transformation by taking $\mathbb{C}_1 = \langle\{\mathbf{start}_1 \to t_0\}, \{t_0 \to \varphi\}, \emptyset\rangle$ and $\mathbb{C}_2 = \langle\emptyset, \emptyset, \emptyset\rangle$. As φ is a conjunction, we apply the transformation to formulae in $\mathcal{G}_1$, obtaining:

$$\mathcal{G}_1 = \{t_0 \to \boxed{}^1\underline{\boxed{}^2(p \to q)}, t_0 \to \boxed{}^1\boxed{}^2 p, t_0 \to \Diamond^1\Diamond^2\neg q\}$$

where the underlined formula is a 2-formula. Complex subformulae in the scope of modal operators are not allowed by the normal form. We introduce the propositional symbol t_1 and replace $\boxed{}^2(p \to q)$ in the scope of $\boxed{}^1$. That is, we have the set $\mathcal{G}_1$ rewritten as

$$\mathcal{G}_1 = \{t_0 \to \boxed{}^1 t_1, t_0 \to \boxed{}^1\boxed{}^2 p, t_0 \to \Diamond^1\Diamond^2\neg q\}$$

and the set $\mathcal{G}_2$ is now given by:

$$\mathcal{G}_2 = \{t_1 \to \boxed{}^2(p \to q)\}.$$

As this is a modal formula, $\mathbb{C}_2$ is rewritten as:

$$\mathcal{G}_2 = \emptyset \text{ and } \mathcal{K}_2 = \{t_1 \to \boxed{}^2\underline{(p \to q)}\}.$$

The underlined formula is a 1-formula, therefore a new propositional symbol t_2 is introduced and linked to the formula $p \to q$. That is, the set of modal formulae in $\mathbb{C}_2$ is rewritten as:

$$\mathcal{K}_2 = \{t_1 \to \boxed{}^2 t_2\}$$

and the corresponding 1-formula is added to $\mathbb{C}_1$, that is, $\mathcal{G}_1$ is rewritten as:

$$\mathcal{G}_1 = \{t_0 \to \boxed{}^1 t_1, \underline{t_0 \to \boxed{}^1\boxed{}^2 p}, \underline{t_0 \to \Diamond^1\Diamond^2\neg q}, t_2 \to (p \to q)\}$$

Similar steps are taken to transform the underlined formulae above. After classical rewriting, the resulting problem is given by $\langle\mathbb{C}_1, \mathbb{C}_2\rangle$, where:

$$\begin{aligned}
\mathbb{C}_1 = \langle & \mathcal{S}_1 = \{\mathbf{start}_1 \to t_0\}, \\
& \mathcal{G}_1 = \{\mathbf{true} \to \neg t_2 \vee \neg p \vee q, \mathbf{true} \to \neg t_4 \vee p, \mathbf{true} \to \neg t_6 \vee \neg q\}, \\
& \mathcal{K}_1 = \{t_0 \to \boxed{}^1 t_1, t_0 \to \boxed{}^1 t_3, t_0 \to \neg\boxed{}^1\neg t_5\}\rangle
\end{aligned}$$

and

$$\begin{aligned}\mathbb{C}_2 = \langle & \mathcal{S}_2 = \{\},\\ & \mathcal{G}_2 = \{\},\\ & \mathcal{K}_2 = \{t_1 \to \boxed{}^2 t_2, t_3 \to \boxed{}^2 t_4, t_5 \to \neg \boxed{}^2 \neg t_6\}\rangle\end{aligned}$$

4.2 Inference Rules

The resolution-based calculus for the connected logics $\mathcal{C}^{\mathcal{M}}$, $\mathsf{RES}_{\varepsilon}$, is applied to a $\mathcal{C}^{\mathcal{M}}$-problem until either a contradiction is found or no new clauses can be generated. Given a $\mathcal{C}^{\mathcal{M}}$-problem $\mathbb{C} = \langle \mathbb{C}_1 = \langle \mathcal{S}_1, \mathcal{G}_1, \mathcal{K}_1 \rangle, \mathbb{C}_2 = \langle \mathcal{S}_2, \mathcal{G}_2, \mathcal{K}_2 \rangle\rangle$, a contradiction is given by $\mathbf{start}_i \to \mathbf{false} \in \mathcal{S}_i$ or $\mathbf{true} \to \mathbf{false} \in \mathcal{G}_i$, for any $i = 1, 2$.

The (sets of) inference rules deal with the different contexts within each component logic. Therefore, there is a set of inference rules to deal with the propositional part of each component language; a set of inference rules to deal with the multi-modal part within each language; and a set of inference rules to deal with the connection between the two languages. The first two set of rules, related to literal and modal resolution, are a syntactic variation of the calculus presented in [19].

In the following, l, l', l_b, $l'_b \in \mathcal{L}it$ ($b \in \mathbb{N}$) and D, D' are disjunctions of literals.

The first set of inference rules correspond to *classical resolution*. Literal resolution (**LRES**) is classical resolution applied to the propositional portion of the multi-modal logic within each component logic. Also, an initial clause may be resolved with either a literal clause or an initial clause (**IRES1** and **IRES2**). For those rules, both the parent clauses and the resolvent are in sets of the same $\mathbb{C}_i$-problem. Because clauses are in a specific form, all three rules are needed for completeness.

$$[\mathbf{IRES1}]\ \frac{\begin{array}{l}\mathbf{true} \to D \lor l\\ \mathbf{start} \to D' \lor \neg l\end{array}}{\mathbf{start} \to D \lor D'} \qquad [\mathbf{IRES2}]\ \frac{\begin{array}{l}\mathbf{start} \to D \lor l\\ \mathbf{start} \to D' \lor \neg l\end{array}}{\mathbf{start} \to D \lor D'} \qquad [\mathbf{LRES}]\ \frac{\begin{array}{l}\mathbf{true} \to D \lor l\\ \mathbf{true} \to D' \lor \neg l\end{array}}{\mathbf{true} \to D \lor D'}$$

The *modal resolution* rules are applied between clauses which refer to the same context, that is, they must refer to the same agent, within the same component logic. For instance, we can resolve two or more $\boxed{a}^1$-clauses (**MRES** and **GEN2**); or several $\boxed{a}^i$-clauses and a literal clause in $\mathcal{G}_i$ (**GEN1** and **GEN3**). The modal inference rules are:

$$
\textbf{[MRES]} \quad \begin{array}{c}
l_1 \rightarrow \boxed{a}^i l \\
l_2 \rightarrow \neg \boxed{a}^i l \\
\hline
\mathbf{true} \rightarrow \neg l_1 \vee \neg l_2
\end{array}
\qquad
\textbf{[GEN2]} \quad \begin{array}{c}
l'_1 \rightarrow \boxed{a}^i l_1 \\
l'_2 \rightarrow \boxed{a}^i \neg l_1 \\
l'_3 \rightarrow \neg \boxed{a}^i \neg l_2 \\
\hline
\mathbf{true} \rightarrow \neg l'_1 \vee \neg l'_2 \vee \neg l'_3
\end{array}
$$

$$
\textbf{[GEN1]} \quad \begin{array}{c}
l'_1 \rightarrow \boxed{a}^i \neg l_1 \\
\vdots \\
l'_m \rightarrow \boxed{a}^i \neg l_m \\
l' \rightarrow \neg \boxed{a}^i \neg l \\
\mathbf{true} \rightarrow l_1 \vee \ldots \vee l_m \vee \neg l \\
\hline
\mathbf{true} \rightarrow \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'
\end{array}
\qquad
\textbf{[GEN3]} \quad \begin{array}{c}
l'_1 \rightarrow \boxed{a}^i \neg l_1 \\
\vdots \\
l'_m \rightarrow \boxed{a}^i \neg l_m \\
l' \rightarrow \neg \boxed{a}^i \neg l \\
\mathbf{true} \rightarrow l_1 \vee \ldots \vee l_m \\
\hline
\mathbf{true} \rightarrow \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'
\end{array}
$$

The inference rule **MRES** is a syntactic variation of classical resolution, as a formula and its negation cannot be true at the same state. The rule **GEN1** corresponds to generalisation and several applications of classical resolution in a particular state: as clauses in $\mathcal{G}_i$ are true at every state, the literal clause in the premises implies $\mathbf{true} \rightarrow \boxed{a}^i(l_1 \vee \ldots \vee l_m \vee \neg l)$; by propositional reasoning and by the axiom **K**, we have $\mathbf{true} \rightarrow \neg \boxed{a}^i \neg l_1 \vee \ldots \vee \neg \boxed{a} \neg l_m \vee \boxed{a} \neg l$; the modal literals in this formula can then be resolved with their complements in the other parent clauses. **GEN2** is a special case of **GEN1**, as the parent clauses can be resolved with tautologies as $\mathbf{true} \rightarrow l_1 \vee \neg l_1 \vee \neg l_2$. **GEN3** is similar to **GEN1** but the contradiction occurs between the right-hand side of the positive a-clauses and the literal clause. The resolvents in the inference rules **GEN1**, **GEN2**, and **GEN3** impose that the literals on the left-hand side of the modal clauses in the premises are not all satisfied whenever their conjunction leads to a contradiction in a successor state. Given the syntactic forms of clauses, the three rules are needed for completeness [19].

The *bridge resolution* rules that deal with the connecting operators, that is, $\boxed{\boxed{a}}^i$ and $\neg \boxed{\boxed{a}}^i$, $a \in \mathcal{I}_i$, are similar to the modal inference rules given above. The inference rules $\mathcal{E}$-**MRES** and $\mathcal{E}$-**GEN2** are, in fact, just syntactic variants of **MRES** and **GEN2**: reasoning can be applied in the component language L_i even if there is no information about the other component, $\mathsf{L}_{\bar{i}}$. However, the inference rules $\mathcal{E}$-**GEN1** and $\mathcal{E}$-**GEN3** are different, in the sense that they implement the reasoning between the two different languages. Note that the modal clauses in the premises of these inference rules are in $\mathbb{C}_i$, but the literal clauses are in the other component, $\mathbb{C}_{\bar{i}}$, as the literals in the scope of the $\boxed{\boxed{a}}^i$ and $\neg \boxed{\boxed{a}}^i$ operators are in the language of $\mathsf{L}_{\bar{i}}$. Therefore, we use the propositional language of $\mathsf{L}_{\bar{i}}$ to pass enough information from $\mathsf{L}_{\bar{i}}$ to L_i, in order to apply the reasoning mechanism in the context of L_i.

$$[\mathcal{E}\textbf{-MRES}]\quad \begin{array}{c} l_1 \to \boxed{a}^i l \\ l_2 \to \neg \boxed{a}^i l \\ \hline \textbf{true} \to \neg l_1 \vee \neg l_2 \end{array}$$

$$[\mathcal{E}\textbf{-GEN2}]\quad \begin{array}{c} l'_1 \to \boxed{a}^i l_1 \\ l'_2 \to \boxed{a}^i \neg l_1 \\ l'_3 \to \neg \boxed{a}^i \neg l_2 \\ \hline \textbf{true} \to \neg l'_1 \vee \neg l'_2 \vee \neg l'_3 \end{array}$$

$$[\mathcal{E}\textbf{-GEN1}]\quad \begin{array}{c} l'_1 \to \boxed{a}^i \neg l_1 \\ \vdots \\ l'_m \to \boxed{a}^i \neg l_m \\ l' \to \neg \boxed{a}^i \neg l \\ \textbf{true} \to l_1 \vee \ldots \vee l_m \vee \neg l \\ \hline \textbf{true} \to \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l' \end{array}$$

$$[\mathcal{E}\textbf{-GEN3}]\quad \begin{array}{c} l'_1 \to \boxed{a}^i \neg l_1 \\ \vdots \\ l'_m \to \boxed{a}^i \neg l_m \\ l' \to \neg \boxed{a}^i \neg l \\ \textbf{true} \to l_1 \vee \ldots \vee l_m \\ \hline \textbf{true} \to \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l' \end{array}$$

The justification for the bridge inference rules are similar to the justification for the modal inference rules. We sketch the soundness proof for some of the bridge inference rules in Section 5. Completeness is sketched in the same section.

Simplification. We assume standard simplification from classical logic to keep the clauses as simple as possible. For example, $D \vee l \vee l$ on the right-hand side of an initial or literal clause would be rewritten as $D \vee l$.

Example 4.2 The schemata given in Example 3.4 is a valid formula in $\mathcal{C}^{\mathcal{M}}$. The following formula is a negated instance of such schema

$$\varphi = \Box^1 \Box^2 (p \to q) \wedge \Box^1 \Box^2 p \wedge \neg \Box^1 \Box^2 q$$

and we show that φ is, in fact, unsatisfiable. The problem resulting from transforming φ into the $\mathsf{SNF}_{\mathcal{E}}$ was given in Example 4.1. Clauses 1-10 are resulting from the transformation of φ into its normal form.

1.	$\textbf{start}_1 \to t_0$	$[\mathcal{S}_1]$	6.	$t_3 \to \Box^2 t_4$	$[\mathcal{K}_2]$
2.	$t_0 \to \Box^1 t_1$	$[\mathcal{K}_1]$	7.	$\textbf{true} \to \neg t_4 \vee p$	$[\mathcal{G}_1]$
3.	$t_1 \to \Box^2 t_2$	$[\mathcal{K}_2]$	8.	$t_0 \to \neg \Box^1 \neg t_5$	$[\mathcal{K}_1]$
4.	$\textbf{true} \to \neg t_2 \vee \neg p \vee q$	$[\mathcal{G}_1]$	9.	$t_5 \to \neg \Box^2 \neg t_6$	$[\mathcal{K}_2]$
5.	$t_0 \to \Box^1 t_3$	$[\mathcal{K}_1]$	10.	$\textbf{true} \to \neg t_6 \vee \neg q$	$[\mathcal{G}_1]$

The refutation proceeds as follows. Clauses 11 and 12 are obtained by applications of classical resolution. Clause 13 is resulting from an application of $\mathcal{E}$**-GEN1** to 2-clauses in $\mathbb{C}_2$ and a literal clause in $\mathbb{C}_1$. Clause 14 is also resulting from an application of $\mathcal{E}$**-GEN1**, but to 1-clauses in $\mathbb{C}_1$ and a literal clause in $\mathbb{C}_2$. As a contradiction was found, given by clause 15, which is obtained by an application of classical resolution, the $\mathcal{C}^{\mathcal{M}}$-problem is unsatisfiable and so is φ.

11. $\mathbf{true} \rightarrow \neg t_2 \vee \neg p \vee \neg t_6$ $[\mathcal{G}_1, \mathbf{LRES}, 10, 4]$

12. $\mathbf{true} \rightarrow \neg t_2 \vee \neg t_4 \vee \neg t_6$ $[\mathcal{G}_1, \mathbf{LRES}, 11, 7]$

13. $\mathbf{true} \rightarrow \neg t_1 \vee \neg t_3 \vee \neg t_5$ $[\mathcal{G}_2, \mathcal{E}\text{-}\mathbf{GEN1}, 12, 9, 6, 3]$

14. $\mathbf{true} \rightarrow \neg t_0$ $[\mathcal{G}_1, \mathcal{E}\text{-}\mathbf{GEN1}, 13, 8, 5, 2]$

15. $\mathbf{start} \rightarrow \mathbf{false}$ $[\mathcal{S}_1, \mathbf{IRES2}, 14, 1]$

5 Correctness Results

In this section, we sketch the correctness results related to the resolution-based calculus for connected logics, $\mathsf{RES}_{\mathcal{E}}$, that is, soundness, termination, and completeness results for the method. The soundness proof shows that the transformation into $\mathsf{SNF}_{\mathcal{E}}$ as well as the application of the inference rules are satisfiability preserving. Termination is ensured by the fact that a given set of clauses contains only finitely many propositional symbols, from which only finitely many $\mathsf{SNF}_{\mathcal{E}}$ clauses can be constructed and therefore only finitely many new $\mathsf{SNF}_{\mathcal{E}}$ clauses can be derived. Completeness is proved by showing that if a given set of clauses is unsatisfiable, there is a refutation produced by $\mathsf{RES}_{\mathcal{E}}$.

The proof that transformation of a formula $\varphi \in \mathcal{F}(\mathcal{C}^{\mathcal{M}})$ into its normal form is satisfiability preserving can be obtained as in [19]. We have added to the transformation rules presented in [19] two new rewrite rules, which deal with the connecting modalities. For the first introduced rewrite rule, $\boldsymbol{\tau_1}$, assume $\langle\langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \rightarrow \boxed{a}^i\varphi\}\rangle, \langle \mathcal{S}_{\bar{i}}, \mathcal{G}_{\bar{i}}, \mathcal{K}_{\bar{i}}\rangle\rangle$ is satisfiable in a model $\mathbb{M}$. Construct $\mathbb{M}'$ exactly as $\mathbb{M}$ but where $\pi_2(w_2)(t_1) = \mathit{true}$ if, and only if, $(\mathbb{M}, w_2) \models \varphi$. It follows from the semantics of implication, the semantics of the connecting modality, and the semantics of $\mathcal{C}^{\mathcal{M}}$-problems that $\mathbb{M}' \models \langle\langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \rightarrow \boxed{a}^i t_1\}\rangle, \langle \mathcal{S}_{\bar{i}}, \mathcal{G}_{\bar{i}} \cup \{t_1 \rightarrow \varphi\}, \mathcal{K}_{\bar{i}}\rangle\rangle$. For the only if part, it is also easy to check that if $\mathbb{M}' \models \langle\langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \rightarrow \boxed{a}^i t_1\}\rangle, \langle \mathcal{S}_{\bar{i}}, \mathcal{G}_{\bar{i}} \cup \{t_1 \rightarrow \varphi\}, \mathcal{K}_{\bar{i}}\rangle\rangle$, then $\mathbb{M}' \models \langle\langle \mathcal{S}_i, \mathcal{G}_i, \mathcal{K}_i \cup \{t \rightarrow \boxed{a}^i\varphi\}\rangle, \langle \mathcal{S}_{\bar{i}}, \mathcal{G}_{\bar{i}}, \mathcal{K}_{\bar{i}}\rangle\rangle$. The proof that the rewrite rule $\boldsymbol{\tau_2}$ is satisfiability preserving is similar.

Soundness proofs for the new inference rules can also be obtained as in [19]. For $\mathcal{E}$-**MRES**, if both left-hand side of the premises, l_1 and l_2, are satisfied at a world w_i of a model $\mathbb{W}_i$, then both the right-hand sides would also be satisfied. As $\boxed{a}^i l$ and $\neg\boxed{a}^i l$ are contradictory, the resolvent imposes that we cannot satisfy both l_1 and l_2 at any state $w_i \in \mathbb{W}_i$, that is, we have $\mathbf{true} \rightarrow \neg l_1 \vee \neg l_2$ added to $\mathbb{C}_i$. For $\mathcal{E}$-**GEN1**, assume there is a state $w_i \in \mathcal{W}_i$, such that $(\mathbb{M}, w_i) \models l'_1 \wedge \ldots \wedge l'_m \wedge l'$. By the semantics of implication and by the semantics of the connecting operators $\boxed{a}^i$ and $\neg\boxed{a}^i\neg$, we have that there is a state $w_{\bar{i}} \in \mathcal{W}_{\bar{i}}$ that satisfies all right-hand sides of those modal clauses in $\mathbb{C}_i$, i.e. we have that $(\mathbb{M}, w_{\bar{i}}) \models l \wedge \neg l_1 \wedge \ldots \wedge \neg l_m$. As the premise $\mathbf{true} \rightarrow l_1 \vee \ldots \vee l_m \vee \neg l$ holds in every state in $\mathcal{W}_{\bar{i}}$, by applying classical resolution at $w_{\bar{i}}$, we obtain a contradiction. Thus, the resolvent of $\mathcal{E}$-**GEN1** requires that no state in $\mathcal{W}_i$ satisfies all the left-hand sides of the modal premises. Similar reasoning applies to $\mathcal{E}$-**GEN2** and $\mathcal{E}$-**GEN3**.

Completeness can be proven similarly to the completeness of the resolution method given in [19], as all modalities in a $\mathcal{C}^{\mathcal{M}}$-problem, including the connecting modalities, behave like **K**-modalities. The proof, only sketched here, is based on a *behaviour graph*, which is essentially a structure that represents all possible models that can be associated with the combined logics. A behaviour graph $\mathcal{G} = \langle \mathcal{N}, \mathcal{B} \rangle$ contains a set $\mathcal{N}$ of nodes, which are maximally consistent sets of literals and modal literals, and a set $\mathcal{B}$ of edges, which are labelled by the indexes of modalities in a given logic, that is, they represent the accessibility relation of agents within that logic. For the $\mathcal{E}$-connected logics, given a $\mathcal{C}^{\mathcal{M}}$-problem $\mathbb{C} = \langle \mathbb{C}_1, \mathbb{C}_2 \rangle$ the behaviour graph is given by $\mathcal{G} = \langle \mathcal{N}_1, \mathcal{B}_1, \mathcal{N}_2, \mathcal{B}_2, \mathcal{B} \rangle$, that is, we have one (sub)behaviour graph $\mathcal{G}_i = \langle \mathcal{N}_i, \mathcal{B}_i \rangle$ associated with the formulae in $\mathbb{C}_i$, for each i, and a set $\mathcal{B}$ of edges labelled by the connecting relations. The completeness proof consists in showing that the applications of the inference rules of $\mathsf{RES}_{\mathcal{E}}$ correspond to deletions of either edges and nodes in the behaviour graph $\mathcal{G}$ related to a $\mathcal{C}^{\mathcal{M}}$-problem. That is, we show that $\mathcal{G}$ is empty if, and only if, the corresponding problem is unsatisfiable and, in this case, that there is a proof by the set of inference rules in $\mathsf{RES}_{\mathcal{E}}$. As the calculus for L_1 and L_2 are both complete, the correspondence between deletions in $\mathcal{G}_i$ and the set of inference rules for $\mathcal{L}_i$ is ensured by the results in [19]. For bridge resolution, during the construction of the behaviour graph, we try to add as many edges related to the connecting modalities as possible. In order to satisfy the clauses in $\mathbb{C}$, some edges and nodes are immediately deleted. After that, some nodes and edges are deleted because the modal clauses are no longer satisfied; thus they must also be deleted from the graph. We consider only the case when $\neg l'$ in a clause such as $l \rightarrow \neg \boxed{k}^i l'$, the literals in the scope of the connecting modality on the right-hand side of the positive modal $\mathcal{E}$-clauses and the literal clauses all contribute to the contradiction. If there is a node w_i that satisfies l and there is no node $w_{\bar{i}} \in \mathcal{N}_{\bar{i}}$, $\langle w_i, w_{\bar{i}} \rangle \in \mathcal{B}$, and $w_{\bar{i}}$ satisfies $\neg l'$, by applying $\mathcal{E}$-**GEN1** to the relevant clauses deletes w_i as required. We can show that all deletions in the graph correspond to (some) applications of the inference rules for $\mathsf{RES}_{\mathcal{E}}$. The introduction of the resolvents of the inference rules for the connected modalities in the component $\mathbb{C}_i$ deletes nodes in the reduced behaviour graph related to the language of L_i as this corresponds to the fact that a modal literal in the form of $\neg \boxed{k}^i \neg l$ (with $k \in \mathcal{I}_i$), where l is a literal, is not satisfied in the structure. By induction on the number of nodes, we can show that the behaviour graph for a $\mathcal{C}^{\mathcal{M}}$-problem $\mathbb{C}$ is empty if, and only if, $\mathbb{C}$ is unsatisfiable.

Termination is ensured by the fact that no new literals are added to a $\mathcal{C}^{\mathcal{M}}$-problem by any of the inference rules in $\mathsf{RES}_{\mathcal{E}}$. Thus, as there is only a finite number of clauses that can be obtained by the method (modulo simplification), at some point either a contradiction is found or no new clauses can be generated.

6 Outlook

In this paper, we have presented a modalised version of $\mathcal{E}$-connections, which formalises a simple combination of K logics via a K-bridge logic. As shown in Section 3, the method does not introduce new bridge principles and, therefore, the interaction that arises can be completely controlled by inspecting newly introduced bridge axioms connecting the various modalities.

We have also presented a sound, complete, and terminating resolution-based method for dealing with such combinations. Transformation into the normal form separates the different dimensions where reasoning is carried out. Thus, different sets of specialised inference rules are applied to the different portions of the language and the calculi for the component logics remain independent. Information between the different modalities within each component logic is made available through the propositional language that those modalities share. The resolution calculus for connected logics also introduces a set of inference rules to deal with the bridge modalities. Those rules are applied to clauses containing connecting modalities in one logic and literal clauses in the other logic. Therefore, when a set of connecting modalities in one logic cannot be satisfied in the model of the other logic, some restrictions are imposed via the propositional language in the first one.

The simplicity of the resolution method for connected logics is due to the fact that the dimensions for reasoning are kept separated. Implementation can be obtained in a quite straightforward way: the provers for the independent logics can be kept separated and the implementation of the bridge inference rules can be kept local, whenever a suitable communication channel between the provers is implemented. Therefore, the method presented here can be easily parallelised and/or distributed. Moreover, as the normal form is independent of the particular proof method we developed here, the transformed problems can be used to feed other theorem provers, after translation (if needed), providing a general approach for reasoning about connected logics.

We strongly believe that the method presented here can be extended to deal with more powerful varieties of connections between logics. For stronger connecting theories, we should be able to establish completeness whenever the bridge inference rules mimic (complete) resolution procedures for logics with corresponding (complete) frame properties introduced by the connecting modalities. This idea is not only applicable to the combination of standard, classical normal modal logics, but, importantly, also to other non-classical logics such as intuitionistic, relevant, or paraconsistent logic.

In this respect, future work will include studying generalisations of the resolution method introduced here, and a detailed comparison to the more algebraic-driven techniques of [1], which also provide general decidability preservation results.

Dedicated reasoning procedures for $\mathcal{E}$-connections will be very relevant in particular for the Distributed Ontology Language DOL [18], currently under standardisation in the `ontoiop.org` working group. DOL is a metalanguage for combining specifications written in various ontology languages, and includes as linking constructs, besides alignments and theory interpretations, also the method of $\mathcal{E}$-connection. As examples, recent application areas of DOL and $\mathcal{E}$-connections include blending in computational creativity [15], architectural design [3], and the Semantic Web [7].

The work presented in this paper, therefore, is a first step towards establishing the connection method as a viable tool for modular knowledge representation with generic proof support. Given the generality of the method, this could significantly contribute to more usable methods to deal with combined logics in a large variety of applications.

References

[1] Baader, F. and S. Ghilardi, *Connecting Many-Sorted Theories*, The Journal of Symbolic Logic **72** (2007), pp. 535–583.

[2] Baader, F., C. Lutz, H. Sturm and F. Wolter, *Fusions of description logics and abstract description systems*, Journal of Artificial Intelligence Research (JAIR) **16** (2002), pp. 1–58.

[3] Bhatt, M., J. Hois and O. Kutz, *Ontological modelling of form and function in architectural design*, Applied Ontology **7** (2012), pp. 233–267.

[4] Borgida, A. and L. Serafini, *Distributed Description Logics: Assimilating Information from Peer Sources*, Journal of Data Semantics **1** (2003), pp. 153–184.

[5] Carnielli, W., M. Coniglio, D. Gabbay, M. Gouveia and C. Sernadas, "Analysis and Synthesis of Logics - How To Cut And Paste Reasoning Systems," Springer, 2008.

[6] Cuenca-Grau, B. and O. Kutz, *Modular Ontology Languages Revisited*, in: *Proc. of the IJCAI'07 Workshop on Semantic Web for Collaborative Knowledge Acquisition (SWeCKa), Hyderabad, India, January 2007*, 2007, pp. 22–31.

[7] Cuenca-Grau, B., B. Parsia and E. Sirin, *Combining OWL Ontologies Using $\mathcal{E}$-Connections*, Journal of Web Semantics **4** (2006), pp. 40–59.

[8] Gabbay, D., A. Kurucz, F. Wolter and M. Zakharyaschev, "Many-Dimensional Modal Logics: Theory and Applications," Number 148 in Studies in Logic, Elsevier, Amsterdam, 2003.

[9] Goldblatt, R., "Logics of Time and Computation," Number 7 in CSLI Lecture Notes, Stanford, CSLI, 1987.

[10] Hustadt, U. and R. A. Schmidt, *On evaluating decision procedures for modal logic*, in: *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI'97)* (1997), pp. 202–207.

[11] Kracht, M. and O. Kutz, *Logically Possible Worlds and Counterpart Semantics for Modal Logic*, in: D. Jacquette, editor, *Philosophy of Logic*, Handbook of the Philosophy of Science **5**, Elsevier, Amsterdam, 2007 pp. 943–996.

[12] Kracht, M. and F. Wolter, *Properties of independently axiomatizable bimodal logics*, The Journal of Symbolic Logic **56** (Dec. 1991), pp. 1469–1485.

[13] Kutz, O., C. Lutz, F. Wolter and M. Zakharyaschev, *$\mathcal{E}$-connections of Description Logics*, in: *Proc. of DL*, Rome, 2003, pp. 178–187.

[14] Kutz, O., C. Lutz, F. Wolter and M. Zakharyaschev, *$\mathcal{E}$-connections of Abstract Description Systems*, Artificial Intelligence **156** (2004), pp. 1–73.

[15] Kutz, O., T. Mossakowski, J. Hois, M. Bhatt and J. Bateman, *Ontological Blending in DOL*, in: T. Besold, K.-U. Kuehnberger, M. Schorlemmer and A. Smaill, editors, *Computational Creativity, Concept Invention, and General Intelligence, Proc. of the 1st Int. Workshop C3GI@ECAI*, Publications of the Institute of Cognitive Science, Osnabrück **01-2012**, Montpellier, France, 2012, pp. 33–40.

[16] Kutz, O., T. Mossakowski and D. Lücke, *Carnap, Goguen, and the Hyperontologies: Logical Pluralism and Heterogeneous Structuring in Ontology Design*, Logica Universalis **4** (2010), pp. 255–333, Special Issue on 'Is Logic Universal?'.

[17] Kutz, O., F. Wolter and M. Zakharyaschev, *A note on concepts and distances*, in: *Proceedings of the International Workshop on Description Logics (DL-2001)*, Stanford, 2001, pp. 113–121.

[18] Mossakowski, T., C. Lange and O. Kutz, *Three Semantics for the Core of the Distributed Ontology Language*, in: M. Donnelly and G. Guizzardi, editors, *7th International Conference on Formal Ontology in Information Systems (FOIS)*, Frontiers in Artificial Intelligence and Applications **239** (2012), pp. 337–352, FOIS Best Paper Award.

[19] Nalon, C. and C. Dixon, *Clausal resolution for normal modal logics*, J. Algorithms **62** (2007), pp. 117–134.
URL http://dl.acm.org/citation.cfm?id=1316091.1316347

[20] Paramasivam, M. and D. A. Plaisted, *Automated deduction techniques for classification in description logic systems*, Journal of Automated Reasoning **3** (1998), pp. 337–364.

[21] Plaisted, D. A. and S. A. Greenbaum, *A Structure-Preserving Clause Form Translation*, Journal of Logic and Computation **2** (1986), pp. 293–304.

[22] Ryan, M. and P.-Y. Schobbens, *Counterfactuals and Updates as Inverse Modalities*, Journal of Logic, Language and Information **6** (1997), pp. 123–146.
URL http://dx.doi.org/10.1023/A%3A1008218502162

[23] Serafini, L. and A. Tamilin, *DRAGO: Distributed Reasoning Architecture for the Semantic Web*, in: *In Proceedings of the Second European Semantic Web Conference (ESWC'05)* (2005), pp. 361–376.