A Combination of Combinatory and Converse PDL with Substitutions

Jon Haël Brenas[1] Rachid Echahed[1] Martin Strecker[2]

[1] CNRS and University of Grenoble
[2] Université de Toulouse / IRIT

Abstract. We introduce a logic called $\mathcal{C}2\mathcal{PDLS}$, motivated by some reasoning about graph rewriting systems. $\mathcal{C}2\mathcal{PDLS}$ is an extension of both combinatory propositional dynamic logic, usually written $\mathcal{CPDL}$, and converse propositional dynamic logic, usually written $\mathcal{CPDL}$ too. In addition to the existing features of both $\mathcal{CPDL}$s, the introduced logic offers the possibility to use the notion of substitutions à la Hoare within its formulae. Such substitutions reflect the effect of some actions on graph structures such as addition or deletion of edges or nodes. These last features led us to introduce restricted universal roles over subsets of the universe. We propose a sound and complete deductive system for $\mathcal{C}2\mathcal{PDLS}$ and show that its validity problem is decidable.

1 Introduction

Graph structures play an important role when modeling complex systems. They are used in different areas going from computer programs to modeling tools in natural science. These graphs could be either static or dynamic. In this paper we are rather concerned with the latter case where graphs can evolve thanks to some actions aimed at performing some graph transformations.

There are several approaches to write programs which handle graph transformations going from classical imperative languages to dedicated rule-based approaches [20]. Reasoning on these transformations is still not mature enough. Some logics tailored to deal with graph transformations have already been proposed (see, e.g. [4,6]) but suffer from the lack of decision procedures.

Recently, an approach to reasoning on graph transformation based on Hoare like calculi [16] has been proposed by several authors (see, e.g., [19,12,1]). Roughly speaking, within such an approach, one may prove that the resulting graph G', obtained after performing an action α over G, fulfills a property P, written $(G' \models P)$, whenever a given graph G satisfies the precondition $P[\alpha]$, $(G \models P[\alpha])$. The notation $[\alpha]$ is known as a *substitution* induced from action α. Different logics could be used to specify formulas such as P or $P[\alpha]$ depending on the properties one may wish to prove. The main features of such logics include facilities to handle substitutions on one hand and the fact to be endowed by some decision procedures on the other hand.

In this paper, we investigate a dynamic logic where substitutions reflecting several elementary actions over graphs are part of its syntax. This logic, called

$\mathcal{C}2\mathcal{PDLS}$, is an extension of both converse [14] and combinatory [17] propositional dynamic logics, both classically designated by $\mathcal{CPDL}$. This allows us to give quite expressive characterizations of graphs: for instance, the formula $[\nu](\langle(\alpha^- \cup \beta^-)^*\rangle C \Rightarrow D)$ describes the graphs such that all nodes that can reach a node labeled C using only edges labeled α or β are labeled D. One of the main contributions of combinatory $\mathcal{PDL}$ is the introduction of the nominals of Hybrid Logics. These can be combined with several modals for instance in temporal logics [2].

The use of dynamic logics [15], an extension of modal logics [8,9], is among the most widespread ways to reason about complex and evolving data. In particular, modal logics are particularly well-suited to describe the relations between diverse states of the data using the programs that allow to go from one state to another. They are also a very efficient way to represent graph structured data. For instance, the formula $\langle\alpha\rangle\phi$ can be used to signify either that from the current state of execution, it is possible to reach a state where ϕ is true by performing α or that the current node is linked, via an edge labeled with α, to a node labeled with ϕ.

On the other hand, Description Logics [5] form one of the most important families of logics used to describe graphs and some of them have thus been extended so that they are closed under substitutions [1,11]. The main issue with DLs is that, even though the more expressive logics allow to define a role as transitive, they lack the reachability assertions that can be expressed using $\mathcal{CPDL}$. That is why we decided to propose and investigate $\mathcal{C}2\mathcal{PDLS}$.

The paper is organized as follows. The logic $\mathcal{C}2\mathcal{PDLS}$ is defined in Sect. 2. Then, in Sect. 3, we prove that the presence of substitutions, while convenient to express the correctness of graph transformations, does not increase the expressive power of $\mathcal{C}2\mathcal{PDLS}$. In Sect. 4, the proof that $\mathcal{C}2\mathcal{PDLS}$ is decidable is sketched. Related work and concluding remarks are given respectively in Sect. 5 and Sect. 6. The missing proofs can be found in the appendix.

2 Syntax and models of $\mathcal{C}2\mathcal{PDLS}$

In this section, we introduce $\mathcal{C}2\mathcal{PDLS}$, a combination of both Combinatory and Converse Propositional Dynamic Logic , both named $\mathcal{CPDL}$ [14,17] augmented with a notion of substitutions. These kind of substitutions are very common in Hoare-like program verification procedures [16]. In order to take into account the interpretations of formulae with substitutions we assume the universe (of names), Σ, split into two subsets Σ_1 and Σ_2. Intuitively, elements whose names are in Σ_1 are the building blocks of the formulae before substitutions are taken into account. Elements whose names are in Σ_2, on the other hand, are introduced when a substitution creates a new node. It also stores the names of elements that have been deleted. The need to access nodes outside of those that form the initial model leads us to the modification of the universal program ν. It is now used with an index indicating on which set of names it operates.

Definition 1 (Syntax of $\mathcal{C}2\mathcal{PDLS}$). *Given three countably infinite and pairwise disjoint alphabets Σ, the set of* names*, Φ_0, the set of* atomic propositions*, and Π_0, the set of* atomic programs*, the language of $\mathcal{C}2\mathcal{PDLS}$ is composed of* formulae*,* programs *and* substitutions*. We partition the set of names Σ into two countably infinite sets Σ_1 and Σ_2 such that $\Sigma_1 \cup \Sigma_2 = \Sigma$ and $\Sigma_1 \cap \Sigma_2 = \emptyset$. Formulae ϕ, programs α and substitutions θ are defined as follows:*

$\phi := i \mid \phi_0 \mid \neg\phi \mid \phi \vee \phi \mid \langle\alpha\rangle\phi \mid \phi[\theta]$

$\alpha := \alpha_0 \mid \nu_S \mid \alpha;\alpha \mid \alpha \cup \alpha \mid \alpha^* \mid \alpha^- \mid \phi?$

$\theta := add(i,\phi_0) \mid del(i,\phi_0) \mid add(i,j,\alpha_0) \mid del(i,j,\alpha_0) \mid$
$\quad i \gg j \mid add(i) \mid del(i)$

where $i, j \in \Sigma$, $\phi_0 \in \Phi_0$, $\alpha_0 \in \Pi_0$ and $S \subseteq \Sigma$.

We denote by Π the set of programs, Φ the set of formulae and Θ the set of substitutions. As usual, $\phi \wedge \psi$ stands for $\neg(\neg\phi \vee \neg\psi)$, $\phi \Rightarrow \psi$ stands for $\neg\phi \vee \psi$, $\phi \Leftrightarrow \psi$ stands for $(\phi \Rightarrow \psi) \wedge (\psi \Rightarrow \phi)$ and $[\alpha]\phi$ stands for $\neg(\langle\alpha\rangle\neg\phi)$.

It is noteworthy that programs define paths between nodes and not actual modifications of the graphs that are handled by the substitutions. One can also see that not all formulae use names in a way that makes sense. Let $\phi \equiv (\langle\nu_{\Sigma_1}\rangle\phi_0)[add(i,\phi_0)]$. Intuitively, ϕ means that there exists a node labeled ϕ_0 ($\langle\nu_{\Sigma_1}\rangle\phi_0$) after labeling i with ϕ_0. On the other hand, let $\phi' \equiv (\langle\nu_{\Sigma_1}\rangle\phi_0)[add(i)][add(i,\phi_0)]$. As substitutions alter graphs from right to left, ϕ' says that there is a node labeled with ϕ_0 after labeling i with ϕ_0 and then creating i. It should not be possible to modify a node that has yet to be created and thus ϕ' should not be possible. Henceforth, we only consider *well-formed* formulae.

Definition 2 (Well-formed formula). *A formula is said to be well-formed if it is possible to find a set $E \subseteq \Sigma$ such that the following inference rules are respected:*

$$\frac{\phi^{E\cup\{i\}} \quad i \notin E}{\phi[add(i)]^E} \qquad \frac{\phi^{E-\{i\}} \quad i \in E}{\phi[del(i)]^E} \qquad \frac{\phi^E \quad \theta \neq add(i), \theta \neq del(i) \quad i,j \in E}{\phi[\theta]^E}$$

$$\frac{}{i^{\Sigma_1}} \qquad \frac{}{\phi_0^{\Sigma_1}} \qquad \frac{\phi^E}{(\neg\phi)^E} \qquad \frac{\phi_1^E \quad \phi_2^E}{(\phi_1 \vee \phi_2)^E}$$

One now has to define how (well-formed) formulae are interpreted. This is done using *models*.

Definition 3 (Model). *A* model *is a tuple $\mathcal{M} = (M, R, \chi, V, \mathcal{E})$ where M is a set called the* universe*, $\chi : \Sigma \to M$ is a surjective mapping such that $\chi(\Sigma_1) \cap \chi(\Sigma_2) = \emptyset$, $R : \Pi \to \mathcal{P}(M^2)$ is a mapping, $V : \Phi \to \mathcal{P}(M)$ is a mapping and $\mathcal{E} : \Phi \Rightarrow \mathcal{P}^2(\Sigma)$ keeps track of the annotation. They are defined such that:*

- *For each $i \in \Sigma$, $V(i) = \{\chi(i)\}$ and $\mathcal{E}(i) = \{\Sigma_1\}$*

- *For each* $\phi_0 \in \Phi_0$, $V(\phi_0) \in \mathcal{P}(\chi(\Sigma_1))$ *and* $\mathcal{E}(\phi_0) = \{\Sigma_1\}$
- $V(\neg\phi) = \overline{V(\phi)}$ *and* $\mathcal{E}(\neg\phi) = \mathcal{E}(\phi)$
- $V(\phi \vee \psi) = V(\phi) \cup V(\psi)$ *and* $\mathcal{E}(\phi \vee \psi) = \mathcal{E}(\phi) \cup \mathcal{E}(\psi)$
- $V(\langle\alpha\rangle\phi) = \{s \in M | \exists t \in M.((s,t) \in R(\alpha) \wedge t \in V(\phi))\}$ *and* $\mathcal{E}(\langle\alpha\rangle\phi) = \mathcal{E}(\phi)$
- *For each* $\alpha_0 \in \Pi_0$, $R(\alpha_0) \in \mathcal{P}(\chi(\Sigma_1)^2)$
- $R(\nu_S) = \chi(S)^2$ *for* $S \subseteq \Sigma$
- $R(\alpha \cup \beta) = R(\alpha) \cup R(\beta)$
- $R(\alpha;\beta) = \{(s,t) \in M^2 | \exists v.((s,v) \in R(\alpha) \wedge (v,t) \in R(\beta))\}$
- $R(\alpha^-) = \{(s,t) \in M^2 | (t,s) \in R(\alpha)\}$
- $R(\alpha^*) = \bigcup_{k<\omega} R(\alpha^k)$ *where* α^k *stands for the sequence* $\alpha;\ldots;\alpha$ *of length* k
- $R(A?) = \{(s,s) \in M^2 | s \in V(A)\}$
- $V(\phi[add(i,\phi_0)]) = V'(\phi)$ *where* $\mathcal{M}' = (M', R', \chi', V', \mathcal{E}')$ *is a model such that* $M' = M$, $R' = R$, $\chi' = \chi$, $\forall\psi_0 \in \Phi_0, \psi_0 \neq \phi_0.V'(\psi_0) = V(\psi_0)$, $V'(\phi_0) = V(\phi_0) \cup \chi(i)$ *and* $\mathcal{E}' = \mathcal{E}$
- $V(\phi[del(i,\phi_0)]) = V'(\phi)$ *where* $\mathcal{M}' = (M', R', \chi', V', \mathcal{E}')$ *is a model such that* $M' = M$, $R = R$, $\chi' = \chi$, $\forall\psi_0 \in \Phi_0, \psi_0 \neq \phi_0.V'(\psi_0) = V(\psi_0)$, $V'(\phi_0) = V(\phi_0) \cap \overline{\chi(i)}$ *and* $\mathcal{E}' = \mathcal{E}$
- $V(\phi[add(i,j,\pi_0)]) = V(\phi)$ *where* $\mathcal{M}' = (M', R', \chi', V', \mathcal{E}')$ *is a model such that* $M' = M$, $\forall\alpha_0 \in \Pi_0, \pi_0 \neq \alpha_0.R'(\alpha_0) = R(\alpha_0)$, $R'(\pi_0) = R(\pi_0) \cup (\chi(i), \chi(j))$, $\chi' = \chi$, $V' = V$ *and* $\mathcal{E}' = \mathcal{E}$
- $V(\phi[del(i,j,\pi_0)]) = V'(\phi)$ *where* $\mathcal{M}' = (M, R', \chi', V')$ *is a model such that* $M' = M$, $\forall\alpha_0 \in \Pi_0, \pi_0 \neq \alpha_0.R'(\alpha_0) = R(\alpha_0)$, $R'(\pi_0) = R(\pi_0) \cap \overline{(\chi(i), \chi(j))}$, $\chi' = \chi$, $V' = V$ *and* $\mathcal{E}' = \mathcal{E}$
- $V(\phi[i \gg j]) = V'(\phi)$ *where* $\mathcal{M}' = (M, R', \chi', V')$ *is a model such that* $M' = M$, $\forall\pi_0 \in \Pi_0.R'(\pi_0) = R(\pi_0) \cap \overline{\{(a,i) \in R(\pi_0)\}} \cup \{(a,j) | (a,i) \in R(\pi_0)\}$, $\chi' = \chi$, $V' = V$ *and* $\mathcal{E}' = \mathcal{E}$.
- $V(\phi[add(i)]) = V'(\phi)$ *where* $\mathcal{M}' = (M', R', \chi', V', \mathcal{E}')$ *is a model such that* $M' = M$, $R' = R$, $\chi' = \chi$, $V' = V$ *and* $\mathcal{E}' = \{S \cup \{i\} | S \in \mathcal{E}\}$
- $V(\phi[del(i)]) = V'(\phi)$ *where* $\mathcal{M}' = (M', R', \chi', V', \mathcal{E}')$ *is a model such that* $M' = M$, $\forall\pi_0, R'(\pi_0) = R(\pi_0) \cap \overline{\{(k,l) | k = i \vee l = i\}}$, $\chi' = \chi$, $\forall\phi_0.V'(\phi_0) = V(\phi_0) \cap \overline{\{i\}}$ *and* $\mathcal{E}' = \{S \cap \overline{\{i\}} | S \in \mathcal{E}\}$

where $\mathcal{P}(E)$ *(resp.* $\mathcal{P}^2(E)$*) stands for the powerset of* E *(resp. the powerset of the powerset of* E*),* $\overline{E}$ *is the complement of* E *w.r.t. the superset used in the context. In the following, we write* $m \models A$ *whenever* $m \in V(A)$ *for* m *being an element of the universe* M *and* A *is a formula. As usual, a formula* A *is said to be satisfiable if there exist a model* $\mathcal{M} = (M, R, \chi, V)$ *and an element* m *of* M *such that* $m \models A$*. When it is not the case,* A *is said to be unsatisfiable.* A *is said to be valid, written* $\models A$*, if* $\neg A$ *is unsatisfiable, that is for every model* $\mathcal{M}$*, for all elements* m *of* $\mathcal{M}$*,* $m \models A$*, and invalid otherwise. We say that a model* $\mathcal{M}$ *satisfies a formula* A *and write* $\mathcal{M} \models A$ *if there exists an element* m *in* M *such that* $m \models A$*.*

Intuitively, the function χ, which is surjective and such that $\chi(\Sigma_1) \cap \chi(\Sigma_2) = \emptyset$, splits the universe into elements whose names are in Σ_1 and those whose names are in Σ_2. This splitting of the universe M is motivated by the evolution

of models (graphs) that we consider in the forthcoming sections. M is split into nodes that are initially part of the graph $\chi(\Sigma_1)$ and nodes that may be used in the future or may have been deleted in the past $\chi(\Sigma_2)$. This is a way to provide the logic with evolving sets that handle nodes that are either used or potentially usable. So, for all $\phi_0 \in \Phi_0$, $V(\phi_0) \in \mathcal{P}(\chi(\Sigma_1))$, thus for all $n_2 \in \chi(\Sigma_2)$, $n_2 \notin V(\phi_0)$ which means that no atomic proposition is satisfied by any unused node. For almost the same reason, for all $\pi \in \Pi_0$, there is no $m \in M$ such that $(n_2, m) \in R(\pi)$ or $(m, n_2) \in R(\pi)$. Thus, all nodes of $\chi(\Sigma_2)$ are such that they satisfy no atomic proposition and they have no incoming or outgoing edge. Moreover, as χ is surjective, ν_Σ is the usual universal role, that is $R(\nu_\Sigma) = M \times M$.

Example 1. Let's say we want to speak about cities and roads. We define Σ, Φ_0 and Π_0 such that $\{\mathbb{R}, \mathbb{P}\} \subset \Sigma_1$, $\mathbb{A} \in \Sigma_2$, $C \in \Phi_0$ and $r \in \Pi_0$ where $\mathbb{R}$ is the name associated with the city Rome, $\mathbb{P}$ is the name associated with the city of Paris, $\mathbb{A}$ is the name associated with the city of Atlantis, C is the atomic proposition associated with cities and r is the atomic program associated with roads.

The common saying that all roads lead to Rome then becomes $cs \equiv [r]\langle r^* \rangle \mathbb{R}$, that is for all roads ($[r]$) it eventually leads($\langle r^* \rangle$) to Rome ($\mathbb{R}$). Fig. 1 shows a model and a counter-model of cs.

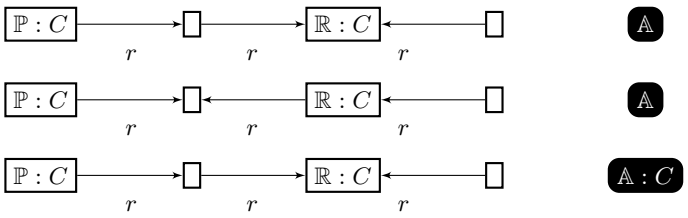

Fig. 1. Models and counter-model. White nodes correspond to names in Σ_1, black ones correspond to those in Σ_2. The first example is a model and every node $\models cs$. The second one is a model but $\mathbb{P} \not\models cs$ hence cs is not valid. The third example is not a model since $\chi(\mathbb{A}) \in V(C)$ which violates Definition3.

Consider now the formula $ac \equiv \langle \nu_{\Sigma_1 \cup \{\chi(\mathbb{A})\}} \rangle (C \vee \mathbb{A})$, it says that there exists an initial node or Atlantis ($\langle \nu_{\Sigma_1 \cup \{\chi(\mathbb{A})\}} \rangle$) such that it is a city or Atlantis ($C \vee \mathbb{A}$). It is an obvious tautology as Atlantis is Atlantis.

As a last example of a formula, we consider $ac' \equiv (\langle \nu_{\Sigma_1} \rangle C)[add(\mathbb{A}, C)][add(\mathbb{A})]$ which says that there exists a city ($\langle \nu_{\Sigma_1} \rangle C$) after adding Atlantis ($[add(\mathbb{A})]$) and making it a city ($[add(\mathbb{A}, C)]$). It is noteworthy that $\mathbb{A} \in \Sigma_2$ initially but as $[add(\mathbb{A})]$ occurs first (that is on the right), it is in Σ_1 when $[add(\mathbb{A}, C)]$ occurs and thus ac' is a (well-formed) formula of $\mathcal{C}2\mathcal{PDLS}$.

Let us consider the formula $ac_1 \equiv (\langle \nu_{\Sigma_1} \rangle C)[add(\mathbb{A}, C)]$. ac_1 is a $\mathcal{C}2\mathcal{PDLS}$ formula only if we assume that $\mathbb{A}$ is in Σ_1.

Finally, let us consider $ac_2 \equiv (\langle \nu_{\Sigma_1} \rangle C)[add(\mathbb{A})][add(\mathbb{A}, C)]$. ac_2 is not a well-formed $\mathcal{C}2\mathcal{PDLS}$ formula since it impossible to find a E such that $\mathbb{A} \in E$, required for $add(\mathbb{A}, C)$, and $\mathbb{A} \notin E$, required for $add(\mathbb{A})$.

3 $\mathcal{C}2\mathcal{PDLS}$ vs $\mathcal{C}2\mathcal{PDL}$

In this section we investigate a relation between the logic $\mathcal{C}2\mathcal{PDLS}$ and its substitution free counterpart named $\mathcal{C}2\mathcal{PDL}$.

Definition 4. *We define the logic $\mathcal{C}2\mathcal{PDL}$ as the restriction of $\mathcal{C}2\mathcal{PDLS}$ to formulae without substitutions.*

Definition 5. *Two formulae A and A' are said to be equivalent, written $A \equiv A'$ if, given any model $\mathcal{M} = (M, R, \chi, V, \mathcal{E})$, $V(A) = V(A')$. Similarly, two programs α and α' are said to be equivalent, written $\alpha \equiv \alpha'$, if $R(\alpha) = R(\alpha')$.*

In the following we state that the logic $\mathcal{C}2\mathcal{PDL}$ is as expressive as $\mathcal{C}2\mathcal{PDLS}$. That is to say, that for every formula Φ of $\mathcal{C}2\mathcal{PDLS}$ there exist a corresponding one Φ' in $\mathcal{C}2\mathcal{PDL}$ such that ϕ and ϕ' are equivalent.

Theorem 1. *$\mathcal{C}2\mathcal{PDLS}$ is as expressive as $\mathcal{C}2\mathcal{PDL}$.*

To prove the theorem, we introduce a rewriting system $\mathcal{RS}$. Its goal is to transform any formula where substitutions occur into a substitution-free formula. It is not always possible to do that in one step. The rewriting system thus contains rules that remove substitutions completely and other rules that moves the substitution inward. Each rule is such that the left-hand side and the right-hand side are equivalent.

Let $\sigma, \sigma' \in \Theta$, $\sigma \in \{[add(i, j, \alpha_0)],\ [del(i, j, \alpha_0)], [add(i)], [i \gg j]\}$, ϕ_0 and $\phi_1 \in \Phi_0$, $\phi_0 \neq \phi_1$, ϕ and $\psi \in \Phi$, $i \in \Sigma$ and $\alpha \in \Pi$ then $\mathcal{RS}_\phi$ is:

Rule $\phi_1 : \top\sigma \rightsquigarrow \top$ *Rule* $\phi_2 : i\sigma \rightsquigarrow i$

Rule $\phi_3 : \phi_0\sigma' \rightsquigarrow \phi_0$ *Rule* $\phi_4 : \phi_0[add(i, \phi_1)] \rightsquigarrow \phi_0$

Rule $\phi_5 : \phi_0[add(i, \phi_0)] \rightsquigarrow \phi_0 \vee i$ *Rule* $\phi_6 : \phi_0[del(i, \phi_1)] \rightsquigarrow \phi_0$

Rule $\phi_7 : \phi_0[del(i, \phi_0)] \rightsquigarrow \phi_0 \wedge \neg i$ *Rule* $\phi_8 : \phi_0[del(i)] \rightsquigarrow \phi_0 \wedge \neg i$

Rule $\phi_9 : (\neg\phi)\sigma \rightsquigarrow \neg(\phi\sigma)$ *Rule* $\phi_{10} : (\phi \vee \psi)\sigma \rightsquigarrow (\phi\sigma) \vee (\psi\sigma)$

Rule $\phi_{11} : (\langle\alpha\rangle\phi)\sigma \rightsquigarrow \langle\alpha\sigma\rangle(\phi\sigma)$

We now introduce rewriting rules allowing to get rid of the substitutions occurring in programs. Let $\sigma, \sigma', \sigma'' \in \Theta$, $\sigma' \in \{[add(i_1, \phi)], [del(i_1, \phi)], [add(i_2)]\}$, $\sigma'' \notin \{[add(i)], [del(i)]\}$, $S \subseteq \Sigma$, ϕ_0 and $\phi_1 \in \Phi_0$ such that $\phi_0 \neq \phi_1$, ϕ and $\psi \in \Phi$, $\alpha_0, \alpha_0' \in \Pi_0$, $\alpha_0 \neq \alpha_0'$ and α, and $\beta \in \Pi$ and $i, j \in \Sigma$, then $\mathcal{RS}_\alpha$ is:

Rule $\alpha_1 : \alpha_0\sigma' \rightsquigarrow \alpha_0$ *Rule* $\alpha_2 : \alpha_0[add(i, j, \alpha_0')] \rightsquigarrow \alpha_0$

Rule $\alpha_3 : \alpha_0[add(i, j, \alpha_0)] \rightsquigarrow \alpha_0 \cup (i?; \nu_{\sigma_1}; j?)$ *Rule* $\alpha_4 : \alpha_0[del(i, j, \alpha_0')] \rightsquigarrow \alpha_0$

Rule $\alpha_5 : \alpha_0[del(i, j, \alpha_0)] \rightsquigarrow (\neg i)?; \alpha_0 \cup \alpha_0; (\neg j)?$

Rule $\alpha_6 : \alpha_0[del(i)] \rightsquigarrow (\neg i)?; \alpha_0; (\neg i)?$

Rule $\alpha_7 : \alpha_0[i \gg j] \rightsquigarrow \alpha_0; ((\neg i)? \cup i?; \nu; j?)$

Rule $\alpha_8 : \nu_S\sigma'' \rightsquigarrow \nu_S$ *Rule* $\alpha_9 : \nu_S[add(i)] \rightsquigarrow \nu_{S[add(i)]}$

Rule $\alpha_{10} : \nu_S[del(i)] \rightsquigarrow \nu_{S[del(i)]}$ *Rule* $\alpha_{11} : (\alpha; \beta)\sigma \rightsquigarrow (\alpha\sigma); (\beta\sigma)$

Rule $\alpha_{12} : (\alpha \cup \beta)\sigma \rightsquigarrow (\alpha\sigma) \cup (\beta\sigma)$ *Rule* $\alpha_{13} : (\alpha^-)\sigma \rightsquigarrow (\alpha\sigma)^-$

Rule $\alpha_{14} : (\alpha^*)\sigma \rightsquigarrow (\alpha\sigma)^*$ *Rule* $\alpha_{15} : (A?)\sigma \rightsquigarrow (A\sigma)?$

The rules α_9 and α_{10} introduce substitutions that affect sets and thus forces the introduction of new rules. Let $i \in \Sigma$, $S_1, S_2 \subseteq \Sigma$:

Rule $S_1 : \Sigma_1[add(i)] \rightsquigarrow \Sigma_1 \cup \{i\}$ *Rule* $S_2 :] \Sigma_1[del(i)] \rightsquigarrow \Sigma_1 \cap \overline{\{i\}}$
Rule $S_3 : \Sigma_2[add(i)] \rightsquigarrow \Sigma_2 \cap \overline{\{i\}}$ *Rule* $S_4 : \Sigma_2[del(i)] \rightsquigarrow \Sigma_2 \cup \{i\}$
Rule $S_5 : (S_1 \cup S_2)\sigma \rightsquigarrow S_1\sigma \cup S_2\sigma$ *Rule* $S_6 : (S_1 \cap S_2)\sigma \rightsquigarrow S_1\sigma \cap S_2\sigma$
Rule $S_7 : \overline{S_1}\sigma \rightsquigarrow \overline{S_1\sigma}$ *Rule* $S_8 : \{i\}\sigma \rightsquigarrow \{i\}$

Proving that these rules are correct, that is that the valuations of the left- and right-hand sides are equal, is not difficult. In order to save space, only few of these proofs are reported here, the others being in the appendix.

Proof.

Rule ϕ_2 : As nodes are never renamed, $V(i\sigma) = V(i)$
Rule α_5 : As $R(\alpha_0[del(i,j,\alpha_0)]) = R'(\alpha_0) = R(\alpha_0) \cap \overline{\{\chi(i), \chi(j)\}}$,
$R(\alpha_0[del(i,j\alpha_0)]) = R((\neg i)?;\alpha_0 \cup \alpha_0;(\neg j)?)$.
Rule S_3 : As i is deleted from Σ_2, $\chi(\Sigma_2[add(i)]) = \chi(\Sigma_2) \cap \overline{\{i\}}$

Example 2. Applying the rules given in the proof allows one to prove that the formula ac' rewrites to ac.

4 Deductive system for $\mathcal{C}2\mathcal{PDL}$

We now introduce a deductive system $\mathcal{DS}$ for $\mathcal{C}2\mathcal{PDL}$. It is composed of 17 axioms (from $(Bool)$ to $(\Sigma_2 2)$) and 5 deductive rules (from (Ax) to (Nec)). It is noteworthy that formulae (resp. programs) of $\mathcal{C}2\mathcal{PDL}$ are also formulae (resp. programs) of $\mathcal{DS}$ and the other way round.

4.1 Deductive system $\mathcal{DS}$

Let A and $B \in \Phi$, α and $\beta \in \Pi$, c and $d \in \Sigma$, $S \subseteq \Sigma$,

- PDL axioms:
  (Bool) All boolean tautologies
  (□) $[\alpha](A \Rightarrow B) \Rightarrow ([\alpha]A \Rightarrow [\alpha]B)$
  (;) $\langle \alpha;\beta\rangle A \Leftrightarrow \langle\alpha\rangle\langle\beta\rangle A$
  (∪) $\langle \alpha \cup \beta\rangle A \Leftrightarrow \langle\alpha\rangle A \vee \langle\beta\rangle A$
  (?) $\langle A?\rangle B \Leftrightarrow A \wedge B$
  (*) $\langle \alpha^*\rangle A \Leftrightarrow A \vee \langle\alpha\rangle\langle\alpha^*\rangle A$
  (-) $A \Rightarrow [\alpha]\langle\alpha^-\rangle A$
- Names
  (Σ1) $\langle\nu_\Sigma\rangle c$
  (Σ2) $\langle\nu_\Sigma\rangle(c \wedge A) \Rightarrow [\nu_\Sigma](c \Rightarrow A)$
- Universal programs
  (ν_S1) $\forall c', d' \in S.c' \Rightarrow \langle\nu_S\rangle d'$
  (ν_S2) $\forall \{c'', d''\} \not\subseteq S.c'' \Rightarrow [\nu_S]\neg d''$
  (ν_S3) $\langle\nu_S\rangle\langle\nu_S\rangle A \Rightarrow \langle\nu_S\rangle A$

($\nu_S 4$) $A \Rightarrow [\nu_S]\langle\nu_S\rangle A$
($\nu_\Sigma 1$) $A \Rightarrow \langle\nu_\Sigma\rangle A$
($\nu_\Sigma 2$) $\langle\alpha\rangle A \Rightarrow \langle\nu_\Sigma\rangle A$

– Names $\in \Sigma_2$
($\Sigma_2 1$) $\forall c \in \Sigma_2, \forall \phi \in \Phi_0 . c \Rightarrow \neg\phi$
($\Sigma_2 2$) $\forall c \in \Sigma_2, \forall \alpha \in \Pi_0 . c \Rightarrow [\alpha]\bot \wedge [\alpha^-]\bot$

– Rules:
We give 5 deductive rules:
(Ax) If A is an axiom, $\vdash A$.
(MP) If $\vdash A$ and $\vdash A \Rightarrow B$, then $\vdash B$
(Ind) If $\vdash [\gamma][\alpha^k]A$, for all $k < \omega$, then $\vdash [\gamma][\alpha^*]A$
(Cov) If $\vdash [\gamma]\neg c$, for all $c \in \Sigma$, then $\vdash [\gamma]\bot$
(Nec) If $\vdash A$, then $\vdash [\nu_\Sigma]A$

Definition 6. *We write $\vdash A$ if A is an axiom of $\mathcal{DS}$ or A can be inferred from the axioms using the deductive rules of $\mathcal{DS}$. We call $\mathcal{LDS}$ the set of $\mathcal{C2PDL}$-formulae $\{A | \vdash A\}$.*

4.2 Soundness

Theorem 2 (Soundness). *Let A be a $\mathcal{C2PDL}$ formula, if $\vdash A$ then $\vDash A$.*

The proof of the soundness theorem is a quite direct. The complete proof is given in the appendix.

We discuss below the case of three axioms, namely the axioms $(-)$, which is not part of Combinatory PDL, $(\Sigma 2)$, which is not part of Converse PDL, and $(\Sigma_2 1)$, which is introduced due to the splitting of the universe. The idea of the proof consists, for a formula A such that $\vdash A$, to show that one can pick any model $\mathcal{M} = (M, V, R, \chi)$, any element m of M and prove that $m \in V(A)$.

(-) Let m be an element of a model $\mathcal{M}$ then:
– Either $m \in V(A)$ and then $\forall m', ((m, m') \notin R(\alpha)$ or $\exists m'' = m.$ such that $(m', m'') \in V(A))$. Thus $m \in V([\alpha]\langle\alpha^-\rangle A)$,
– or $m \notin V(A)$ and thus $m \in V(\neg A)$.
In all cases, $m \in V([\alpha]\langle\alpha^-\rangle A \vee \neg A)$ thus $m \in V(A \Rightarrow [\alpha]\langle\alpha^-\rangle A)$.

($\Sigma 2$) Let m be an element of a model $\mathcal{M}$ then:
– either $\chi(c) \notin V(A)$ and thus $m \in V(\langle\nu_\Sigma\rangle(c \wedge \neg A))$ but then $\forall m''.\ m'' \notin V(c) = \{\chi(c)\}$ or $m'' \notin V(A)$ thus $m \in V([\nu_\Sigma](\neg c \vee \neg A)$. Thus $m \in V([\nu_\Sigma](\neg c \vee \neg A) \wedge \langle\nu_\Sigma\rangle(c \wedge \neg A))$,
– or $\chi(c) \in V(A)$ and thus $\forall m', m' \notin V(c)$ or $m' \in V(A)$ thus $m \in V([\nu_\Sigma](\neg c \vee A)$. But then $\exists m'' = \chi(c)$ such that $m'' \in V(c \wedge A)$ and thus $m \in V(\langle\nu_\Sigma\rangle(c \wedge A))$ thus $m \in V([\nu_\Sigma](\neg c \vee A) \wedge \langle\nu_\Sigma\rangle(c \wedge A))$
In all possible cases, $m \in V(([\nu_\Sigma](\neg c \vee \neg A) \wedge \langle\nu_\Sigma\rangle(c \wedge \neg A)) \wedge ([\nu_\Sigma](\neg c \vee A) \wedge \langle\nu_\Sigma\rangle(c \wedge A)))$ that is $m \in V(\langle\nu_\Sigma\rangle(c \wedge A) \Leftrightarrow [\nu_\Sigma](c \Rightarrow A))$

($\Sigma_2 1$) Let m be a element of a model $\mathcal{M}$, $\phi \in \Phi_0$ then:
– Either $m \in V(c) = \{\chi(c)\}$ and then as $V(\phi) \subseteq \chi(\Sigma_1)$ and $\chi(\Sigma_1) \cap \chi(\Sigma_2) = \emptyset$, $m \notin V(\phi)$ and thus $m \in V(\neg\phi)$,
– or $m \notin V(c)$ and thus $m \in V(\neg c)$.
In all possible cases, $m \in V(\neg c \vee \neg\phi)$ that is $m \in V(c \Rightarrow \neg\phi)$

4.3 Completeness

Theorem 3 (Completeness). *Let A be a $\mathcal{C}2\mathcal{PDL}$ formula, if $\vDash A$ then $\vdash A$.*

The completeness proof is much more involved than the soundness proof. The idea is to prove that if $\nvdash A$ then $\nvDash A$, which is obviously equivalent to Theorem 3.

The main argument of the proof makes use of the notion of extension of the logic $\mathcal{C}2\mathcal{PDL}$:

Definition 7. *A logic (over $\mathcal{DS}$) is any set of $\mathcal{C}2\mathcal{PDL}$ formulae L such that:*

- *L contains all axioms of $\mathcal{DS}$*
- *L is closed under (MP), (Ind), (Cov) and (Nec).*

To establish the completeness, the notion of logics that do not lead to inconsistencies is used.

Definition 8. *A logic L is consistent if $\bot \notin L$.*

We can now state the following theorem.

Theorem 4. *If L is a consistent logic, then L has a model.*

Let $log(\Gamma, A)$ denote the least logic containing the set of formulae Γ and the formula A. Then we can show that Theorem 3 is a consequence of Theorem 4. Indeed, assume $\nvdash A$. Then, $log(\mathcal{LDS}, \neg A)$ is consistent. Thus, from Theorem 4, $log(\mathcal{LDS}, \neg A)$ has a model. That is $\nvDash A$.

In order to prove Theorem 4, one may use the notion of maximal logics and the Lindenbaum lemma.

Definition 9. *A logic L is said to be maximal if for all $\mathcal{C}2\mathcal{PDL}$-formulae A, either $A \notin L$ or $A \in L$.*

Lemma 1 (Lindenbaum lemma). *If L is a consistent logic then there exists a maximal consistent logic L^* such that $L \subseteq L^*$.*

The proof of Lemma 1 is quite straightforward. The set of $\mathcal{C}2\mathcal{PDL}$-formulae being recursively enumerable, it is possible to make a list of them, say $\{\phi_1, \phi_2, \ldots\}$. Then, starting from $L_0 = L$, we try adding the n-th formula ϕ_n to L_{n-1}. If $log(L_{n-1}, \phi_n)$ is consistent, we define it at L_n, if not $log(L_{n-1}, \neg\phi_n)$ is consistent and it is defined as L_n. The final step consists in proving that the union of all the L_ns is a maximal consistent logic.

In order to build the model for the proof of Theorem 4, the main remaining obstacle is that names can occur several times, at different nodes (elements of the universe). Remember that each name can only name one element. Then, to solve this issue, we introduce an equivalence relation over names: $c \sim d = \langle \nu_\Sigma \rangle (c \wedge d) \in L^*$. $[c]_\sim$ is defined as the equivalence class of c. Intuitively, $c \sim d$ if both c and d name the same node. We define $\mathcal{M}_\sim = (M_\sim, R_\sim, \chi_\sim, V_\sim)$, where

$M_\sim = \{[c]_\sim | c \in \Sigma\}$, for all $\alpha \in \Pi$, $R_\sim(\alpha) = \{([c]_\sim, [d]_\sim) | \langle \nu_\Sigma \rangle (c \wedge \langle \alpha \rangle d) \in L^*\}$, for all $c \in \Sigma$, $\chi_\sim(c) = [c]_\sim$ and for all $A \in \Phi$, $V_\sim(A) = \{[c]_\sim | \langle \nu_\Sigma \rangle (c \wedge A) \in L^*\}$. One now has to prove that $\mathcal{M}_\sim$ is a model. It is straightforward thanks to the Lemma 1 that allows us to use a maximal logic. Indeed, let us show that $V_\sim(\neg A) = \overline{V_\sim(A)}$:

- Assume $[c] \in \overline{V_\sim(A)}$ then $\langle \nu_\Sigma \rangle (c \wedge A) \notin L^*$. As L^* is maximal, $[\nu_\Sigma](c \Rightarrow \neg A) \in L^*$. As $(\Sigma 2)$ is in L^* so is $\langle \nu_\Sigma \rangle (c \wedge \neg A) \notin L^*$ and thus $[c] \in V_\sim(\neg A)$. Thus $\overline{V_\sim(A)} \subseteq V_\sim(\neg A)$.
- Otherwise $[c] \in V_\sim(A)$ then $\langle \nu_\Sigma \rangle (c \wedge A) \in L^*$. If $[c] \in V_\sim(\neg A)$, then $\langle \nu_\Sigma \rangle (c \wedge \neg A) \in L^*$. As $(\Sigma 2)$ is in L^* so is $[\nu_\Sigma](c \Rightarrow \neg A) \in L^*$ and thus $\langle \nu_\Sigma \rangle (A \wedge \neg A) \in L^*$. As L^* is consistent, this is impossible and thus $V_\sim(\neg A) \subseteq \overline{V(A)}$.

Thus $V_\sim(\neg A) = \overline{V_\sim(A)}$.

The complete, and much more precise, proof can be found in the appendix.

4.4 Decidability

Theorem 5 (Decidability). *The validity problem of $\mathcal{C}2\mathcal{PDL}$ is decidable.*

The first step towards proving Theorem 5 is to find two semi-decision procedures: one that stops when called upon if the formula given as argument is valid and another one that stops if the formula is not valid. As a formula has to be one of the two, the decision procedure will stop.

We start with the validity semi-decision procedure. The deductive system is a good starting point but there are problems with the rules (Ind) and (Cov) as they both quantify on infinite sets (the integers and the names respectively). We thus drop them to form a new logic that generates the same set of valid formulae but whose validity problem is decidable:

Definition 10. *Let $\mathcal{FDS}$ be the deductive system obtained from $\mathcal{DS}$ by dropping the rules (Ind) and (Cov) and adding the axiom (ind): $(A \wedge [\alpha^*](A \Rightarrow [\alpha]A)) \Rightarrow [\alpha^*]A$. Let $\vdash_F$ denote provability in $\mathcal{FDS}$. We call $\mathcal{LFDS}$ the set of $\mathcal{C}2\mathcal{PDL}$-formulae $\{A | \vdash_F A\}$.*

As $\mathcal{FDS}$ is ω-rule-free, it is obvious that $\mathcal{LFDS}$ is a recursively enumerable set. We now have to prove that every formula of $\mathcal{LFDS}$ is also a valid formula of $\mathcal{C}2\mathcal{PDL}$.

Lemma 2. *Let A be a formula of $\mathcal{C}2\mathcal{PDL}$, if $\vdash_F A$ then $\vdash A$*

The proof of Lemma 2 simply amounts to proving that $\vdash ind$.

The semi-decision procedure that decides whether a formula is valid is based on Lemma 2.

To find an invalidity semi-decision procedure, we use the following theorem.

Theorem 6. *Let A be a formula of $\mathcal{C}2\mathcal{PDL}$, if $\nvdash_F A$ then, for some finite model $\mathcal{M}$, $\mathcal{M} \nvDash A$.*

To prove Theorem 6 we build a canonical quasi-model.

Definition 11. *We name canonical quasi-model the model* $\mathcal{M}_c = (M_c, R_c, V_c)$ *where:*

- M_c *is the set of all maximal consistent sets of formulae*
- *for every program* α *and for all* $u, v \in M_c$, $u\ R_c(\alpha)\ v$ *iff, for every formula* A, *if* $[\alpha]A \in u$ *then* $A \in v$
- *for every atomic proposition* ϕ, $V_c(\phi) = \{u \in M_c | \phi \in u\}$
- *for every name* i, $V_c(i) = \{u \in M_c | i \in u\}$

$\mathcal{M}_c$ is still not a model but it's possible to obtain a finite model from it by doing a filtration [22] as defined below:

Definition 12. *Let* $\mathcal{M} = (M, R, \chi, V)$ *be a model and let* Γ *be any set of formulae closed under sub-formulae. We define the equivalence relation* $\sim_\Gamma$ *on* M *by:*
$\forall s, t \in M. s \sim_\Gamma t$ *iff* $\forall \phi \in \Gamma$, *(*$s \vDash \phi$ *iff* $t \vDash \phi$*).*

We note $[s]_\Gamma$ *the equivalence class of* s *with respect to* $\sim_\Gamma$. *The structure* $\mathcal{M}_\Gamma = (M_\Gamma, R_\Gamma, \chi_\Gamma, V_\Gamma)$ *is called filtration of* $\mathcal{M}_c$ *with respect to* Γ *if:*

- $M_\Gamma := \{[s]_\Gamma | s \in M_c\}$
- *for every program* $\alpha \in \Gamma$, *if* $sR_c(\alpha)t$, *then* $[s]_\Gamma R_\Gamma(\alpha)[t]_\Gamma$
- *for every program* $\alpha \in \Gamma$, *if* $[s]_\Gamma R_\Gamma(\alpha)[t]_\Gamma$, *then for all formulae* A, $[\alpha]A \in s \cap \Gamma$ *only if* $A \in t$
- *for every name in* $o \in \Gamma$, *if* $o \in s$, $[s]_\Gamma \in \chi_\Gamma(o)$
- *for every atomic proposition* $\phi_0 \in \Gamma$, $V_\Gamma(\phi_0) = \{[s]_\Gamma | s \in V_c(\phi_0)\}$

The proof that $\mathcal{M}_\Gamma$ is a model is not very involved, as it is a simple check of all the conditions, but it will not be reported here for lack of space. The proof that $\mathcal{M}_\Gamma \not\Vdash A$ is less obvious and it rests mainly on the fact that if $\not\Vdash_F A$, then there exists a maximal set of formulae u not containing A. Thus $[u]_\Gamma \not\Vdash A$.

We introduce the Fisher-Ladner closure of a set of formulae which is a set closed under sub-formulae that we will use for the filtration of the canonical quasi-model.

Definition 13. *The Fischer-Ladner closure of a set of formulae* Ξ *is the smallest set* $\mathcal{FL}$ *that satisfies:*

- $\Xi \subseteq \mathcal{FL}$
- $\mathcal{FL}$ *is closed under sub-formulae*
- *If* $[\alpha \cup \beta]A \in \mathcal{FL}$, $[\alpha]A \in \mathcal{FL}$ *and* $[\beta]A \in \mathcal{FL}$
- *If* $[\alpha; \beta]A \in \mathcal{FL}$, $[\alpha][\beta]A \in \mathcal{FL}$
- *If* $[\alpha^*]A \in \mathcal{FL}$, $[\alpha][\alpha^*]A \in \mathcal{FL}$
- *If* $[\alpha^-]A \in \mathcal{FL}$, $[\alpha]\neg[\alpha^-]A \in \mathcal{FL}$

It is obvious that the Fisher-Ladner closure of a finite set of formulae is itself finite and that it is closed under sub-formulae.

We can now prove that we obtain that way a finite model.

Lemma 3. *If Γ is such that $|\Gamma| = n$, where $|\Gamma|$ is the cardinality of Γ that is the number of formulae it contains, then $|\mathcal{M}_\Gamma| \leq 2^n$.*

The proof of Lemma 3 is obvious as there are at most 2^n equivalence classes for n formulae.

We can now go back to sketch the proof of Theorem 6. We can exhibit a finite model $\mathcal{M}$, the canonical quasi-model filtrated by the Fisher-Ladner closure of $\{A\}$ such that if $\not\vdash_F A$ then $\mathcal{M} \not\models A$.

Theorem 6 is used to prove that A is invalid. The procedure tries all finite models and stops when it finds one such that $\not\models A$. On the other hand, Lemma 2 gives us the assurance that, if A is valid, it will be generated eventually by $\mathcal{FDS}$. We can thus decide whether or not a $\mathcal{C}2\mathcal{PDL}$-formula A is valid. As usual, we can also prove that a formula A is satisfiable by proving that $\neg A$ is invalid.

The complete proof can be found in the appendix.

Another possible approach would be to use the Hybrid μ-calculus[21]. This logic as almost the same constructors as $\mathcal{C}2\mathcal{PDL}$ but replaces the closure with the μ and ν constructors of μ calculus. It is known to be decidable. The translation of the closure in μ-calculus is simple and well-known. The only difficulty is the use of the sets in ν_S. This can be tackled by introducing new atomic propositions that label the nodes in S and only them. One also has to had in the formula that nodes named with elements of Σ_2 are such that none of the atomic propositions label them and they have neither incoming or outgoing edges.

5 Related work

We recall that our goal in defining $\mathcal{C}2\mathcal{PDLS}$ was to introduce a logic that would be both decidable and expressive enough to characterize basic substitutions over graphs. Furthermore, we wanted the logic to be able to speak about named nodes, and thus to contain nominals, and to be able to express reachability, and thus contain the Kleene star.

Some expressive logics have been introduced that are able to deal with actions over graphs. In [4], the authors introduced two ways to extend 'static' modal logics with actions that are similar to ours. The first one allows to modify node and edge labelling globally while the second one modifies them locally. The first one is proven to have the same status w.r.t. completeness and decidability as the original 'static' modal logic. The second one yields undecidable logics. Among the crucial differences between their work and ours is that they allow actions, that they introduce, and programs, that describe the models, to interact while we separate them. Furthermore, their models are rooted in that the exact position at which one is in the graph is key. We do not care on the other hand. The second logic being undecidable does not meet our requirements and the first

one, by allowing only global modifications, prevents us from explicitly stating where the modifications occur.

In [6], the authors introduced a quite expressive logic that extends both logics of [4] by allowing global and local modifications of the labelings. This logic is obviously very expressive but its validity problem is undecidable.

Some approaches introduce actions that are relevant for the subject they deal with but cannot really be used when one tries, as we do, to define graph transformations. Public Announcement Logic [18], for instance, deals with multi-agent epistemic logic and adds operators for public communications of message α that removes from the models all nodes that do not satisfy α. This approach is generalized by van Ditmarsch et al. [23]. In a different direction, Fernandez-Duque et al. [13] introduced an epistemic logic allowing to forget information. These logics target some particular classes of graphs which limit their use for graph transformations in general.

In [7], Balbiani et al. introduced the Dynamic Logic of Propositional Assignment. This logic allows to dynamically assign propositional values. The expressive power of this logic and $\mathcal{C}2\mathcal{PDLS}$ are not comparable in the sense that there are formulae that can be expressed in one logic but not in the other one and the other way round. Furthermore, with $\mathcal{C}2\mathcal{PDLS}$ one may reason about evolving graph structures which is not that obvious in [7].

In [3] an operator is proposed to extend modal logic. It allows one to swap an edge of the graph, in addition to the usual operators of modal logic, that is to consider the graph where an edge $\{v, w\}$ would be replaced by an edge $\{w, v\}$. The logic presented in [3] and $\mathcal{C}2\mathcal{PDLS}$ are both targeted to reason on graph transformations but differ on the properties that can be expressed.

In [11], we have studied the same substitutions in the present paper, minus the creation and deletion of nodes and the global redirection of edges, and looked at whether or not some Description Logics are closed under them. Once again, the main goal is to identify logics which are decidable and can also express substitutions.

In [1], Ahmetaj et al. studied the logic $\mathcal{ALCHOIQ}br$. It is an unusual Description Logic that has been extended so that substitutions can be written as part of the formulae . It is then used to solve planning problems. $\mathcal{ALCHOIQ}br$ is finitely decidable but lacks the Kleene star.

One could hope to merge the gap between dynamic logics and description logics to obtain a logic containing both regular role expressions and counting quantifiers. This is particularly interesting as some important graph-like structures are defined using both. The logic that is reached, though, will not be decidable as it has been proven, in [10], that logics with regular role expressions and counting quantifiers are undecidable.

6 Conclusion

We introduced a new extension of dynamic logics, we called $\mathcal{C}2\mathcal{PDLS}$, to define properties of evolving graphs. Our main goal in doing so was to provide a logic

able to express two different ideas that are usually associated with the dynamic part of dynamic logics. First, the use of classical constructors of dynamic logics (such as union, intersection, composition, ... of programs) to express complex conditions on the way the various nodes of graphs can be connected. Meanwhile, other constructs borrowed from Hoare logics and called substitutions are used to express conditions on future states of graphs after performing some actions. These differ from the usual programs of dynamic logic in that the effect of each substitution in term of model is well-defined and not left to be chosen when the model is built. In order to avoid confusion, these two dynamic aspects were completely separated out with different notations.

In addition to the usual constructs of dynamic logic, we split the nodes of the considered graphs into two sets: one contains the nodes that are actually part of the graph at the inception of the formulae, the other contains all nodes that do not belong to the graph. This second set of nodes is intended to store nodes that will be created by future transformations.

Once the syntax of $\mathcal{C}2\mathcal{PDLS}$ and its models have been properly defined, we looked at the properties of $\mathcal{C}2\mathcal{PDLS}$. We proved that the substitutions do not increase the expressive power of the logic and provided a rewriting system allowing to translate a formula with substitutions (that is in $\mathcal{C}2\mathcal{PDLS}$) to a substitution-free formula (that is in $\mathcal{C}2\mathcal{PDL}$). We provided a reasoning system that we proved to be both sound and complete. We also proved that the validity (and thus also the satisfiability) problem in $\mathcal{C}2\mathcal{PDL}$ (and consequently in $\mathcal{C}2\mathcal{PDLS}$) is decidable.

An interesting way to continue this line of work would be to extend the substitutions toward a more dynamic structure in which substitutions would not only allow one action to be performed but full programs as is usual in dynamic logic and then to see whether or not it would be possible to make logical programs (those used to model the arcs of the graph) and the graph transformations interact.

References

1. Shqiponja Ahmetaj, Diego Calvanese, Magdalena Ortiz, and Mantas Simkus. Managing change in graph-structured data using description logics. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, pages 966–973, 2014.
2. C Areces, P Blackburn, and M Marx. The computational complexity of hybrid temporal logics. *Logic Journal of IGPL*, 8(5):653–679, 2000.
3. Carlos Areces, Raul Fervari, and Guillaume Hoffmann. Swap logic. *Logic Journal of the IGPL*, 22(2):309–332, 2014.
4. Guillaume Aucher, Philippe Balbiani, Luis Fariñas del Cerro, and Andreas Herzig. Global and local graph modifiers. *Electronic Notes in Theoretical Computer Science*, 231:293 – 307, 2009. Proceedings of the 5th Workshop on Methods for Modalities (M4M5 2007).
5. Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

6. Philippe Balbiani, Rachid Echahed, and Andreas Herzig. A modal logic for term-graph rewriting. *CoRR*, abs/1003.4369, 2010.
7. Philippe Balbiani, Andreas Herzig, and Nicolas Troquard. Dynamic logic of propositional assignments: A well-behaved variant of PDL. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25-28, 2013*, pages 143–152, 2013.
8. Patrick Blackburn, Johan F. A. K. van Benthem, and Frank Wolter. *Handbook of Modal Logic, Volume 3 (Studies in Logic and Practical Reasoning)*. Elsevier Science Inc., New York, NY, USA, 2006.
9. Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*. Cambridge University Press, 2001. Cambridge Books Online.
10. Piero A. Bonatti. On the undecidability of description and dynamic logics with recursion and counting. In *IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003*, pages 331–336, 2003.
11. Jon Haël Brenas, Rachid Echahed, and Martin Strecker. On the closure of description logics under substitutions.
12. Jon Haël Brenas, Rachid Echahed, and Martin Strecker. A hoare-like calculus using the SROIQ σ logic on transformations of graphs. In *Theoretical Computer Science - 8th IFIP TC 1/WG 2.2 International Conference, TCS 2014, Rome, Italy, September 1-3, 2014. Proceedings*, pages 164–178, 2014.
13. David Fernández Duque, Ángel Nepomuceno-Fernández, Enrique Sarrión-Morillo, Fernando Soler-Toscano, and Fernando R. Velázquez-Quesada. Forgetting complex propositions. *CoRR*, abs/1507.01111, 2015.
14. Michael J. Fischer and Richard E. Ladner. Propositional dynamic logic of regular programs. *J. Comput. Syst. Sci.*, 18(2):194–211, 1979.
15. David Harel, Jerzy Tiuryn, and Dexter Kozen. *Dynamic Logic*. MIT Press, Cambridge, MA, USA, 2000.
16. C. A. R. Hoare. An axiomatic basis for computer programming. *Commun. ACM*, 12(10):576–580, 1969.
17. Solomon Passy and Tinko Tinchev. An essay in combinatory dynamic logic. *Inf. Comput.*, 93(2):263–332, 1991.
18. Jan Plaza. Logics of public communications. *Synthese*, 158(2):165–179, 2007.
19. Christopher M. Poskitt and Detlef Plump. Verifying monadic second-order properties of graph programs. In *Graph Transformation - 7th International Conference, ICGT 2014, Held as Part of STAF 2014, York, UK, July 22-24, 2014. Proceedings*, pages 33–48, 2014.
20. Grzegorz Rozenberg, editor. *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*. World Scientific, 1997.
21. Ulrike Sattler and Moshe Y. Vardi. The hybrid μ-calculus. In *Proceedings of the First International Joint Conference on Automated Reasoning*, IJCAR '01, pages 76–91, London, UK, UK, 2001. Springer-Verlag.
22. K. Segerberg. A completeness theorem in the modal logic of programs. In *Universal algebra and applications*, pages 31–46. PWN-Polish Scientific Publishers, 1982.
23. Hans P. van Ditmarsch, Wiebe van der Hoek, and Barteld P. Kooi. Dynamic epistemic logic and knowledge puzzles. In *Conceptual Structures: Knowledge Architectures for Smart Applications, 15th International Conference on Conceptual Structures, ICCS 2007, Sheffield, UK, July 22-27, 2007, Proceedings*, pages 45–58, 2007.

Appendix

In these appendices, we will report the proofs of the soundness (in Sect. 6.1) and completeness (in Sect. 6.2) theorems. We also prove (in Sect. 6.3) that the validity problem is decidable and (in Sect. 6.4) that the rules introduced in Sect. 2 are correct.

We will need a few more definitions in order to shorten a few expressions.

Definition 14. *We say that, given two sets of formulae, $\mathcal{S}_1$ and $\mathcal{S}_2$, $\mathcal{S}_1 \models \mathcal{S}_2$ if $\forall \phi \in \mathcal{S}_2$, $(\bigwedge_{\psi \in \mathcal{S}_1} \psi) \Rightarrow \phi$ is valid. A set of formulae S is said to have a model $\mathcal{M} = (M, R, \chi, V)$ if, for every $\phi \in S$, there exists an element m of M such that $m \models \phi$. In order to clarify which model is being used, we may write $\mathcal{M}, m \vdash A$ instead of just $m \vdash A$.*

6.1 Soundness

Theorem 2 (Soundness) . *Let A be a $\mathcal{C}2\mathcal{PDL}$ formula, if $\vdash A$ then $\models A$.*

The proof is a straightforward induction on $\vdash$:

(Bool) Every node of the model $\mathcal{M}$ satisfies the boolean tautology A. Hence, $\mathcal{M} \models A$.

($\Box$) Let m be a node of the model $\mathcal{M}$ then:
- Either $\exists m'$ such that $(m, m') \in R(\alpha)$ and $m' \in V(A \wedge \neg B)$ and thus $m \in V(\langle\alpha\rangle(A \wedge \neg B))$,
- or $\forall m'.(m, m') \notin R(\alpha)$ or $m' \in V(\neg A \vee B)$. Then either $\exists m''$ such that $(m, m'') \in R(\alpha)$ and $m'' \in V(\neg A)$ and thus $m \in V(\langle\alpha\rangle\neg A)$,
- or $\forall m'.(m, m') \notin R(\alpha)$ or $m' \in V(B)$ and then $m \in V([\alpha]B)$.

In all possible cases, $m \in V(\langle\alpha\rangle(A \wedge \neg B) \vee \langle\alpha\rangle\neg A \vee [\alpha]B))$. Thus $m \in V([\alpha](A \Rightarrow B) \Rightarrow ([\alpha]A \Rightarrow [\alpha]B))$.

(;) Let m be a node of the model $\mathcal{M}$ then:
- Either $\exists m', m''$ such that $(m, m') \in R(\alpha)$ and $(m', m'') \in R(\beta)$ and $m'' \in V(A)$, that is $m \in V(\langle\alpha\rangle\langle\beta\rangle A)$, and thus $\exists m''$ such that $(m, m'') \in R(\alpha; \beta)$ and $m'' \in V(A)$ that is $m \in V(\langle\alpha; \beta\rangle A)$,
- or $\forall m', m''$, $(m, m') \notin R(\alpha)$ or $(m', m'') \notin R(\beta)$ or $m'' \notin V(A)$, that is $m \in V([\alpha][\beta]\neg A)$, and thus $\forall m''.(m, m'') \notin R(\alpha; \beta)$ or $m'' \notin V(A)$ that is $m \in V([\alpha; \beta]\neg A)$.

In all possible cases, $m \in V((\langle\alpha\rangle\langle\beta\rangle A \wedge \langle\alpha; \beta\rangle A) \vee ([\alpha][\beta]\neg A \wedge [\alpha; \beta]\neg A))$. Thus $m \in V(\langle\alpha\rangle\langle\beta\rangle A \Leftrightarrow \langle\alpha; \beta\rangle A)$.

($\cup$) Let m be a node of the model $\mathcal{M}$ then:
- Either $\exists m'$ such that $(m, m') \in R(\alpha)$ and $m' \in V(A)$, that is $m \in V(\langle\alpha\rangle A)$ and $m \in V(\langle\alpha\rangle A \vee \langle\beta\rangle A)$, and thus $\exists m'$ such that $(m, m') \in R(\alpha \cup \beta)$ and $m' \in V(A)$, that is $m \in V(\langle\alpha \cup \beta\rangle A)$,
- or $\exists m'$ such that $(m, m') \in R(\beta)$ and $m' \in V(A)$, that is $m \in V(\langle\beta\rangle A)$ and $m \in V(\langle\alpha\rangle A \vee \langle\beta\rangle A)$, and thus $\exists m'$ such that $(m, m') \in R(\alpha \cup \beta)$ and $m' \in V(A)$, that is $m \in V(\langle\alpha \cup \beta\rangle A)$,

– or $\forall m'$, either $(m,m') \notin R(\alpha)$ and $(m,m') \notin R(\beta)$ or $m' \notin V(A)$, that is $m \in V([\alpha]\neg A \wedge [\beta]\neg A)$, and thus $\forall m'$, $(m,m') \notin R(\alpha \cup \beta)$ or $m' \notin V(A)$ that is $m \in V([\alpha \cup \beta]\neg A)$.

In all possible cases, $m \in V(((\langle\alpha\rangle A \vee \langle\beta\rangle A) \wedge \langle\alpha \cup \beta\rangle A) \vee ([\alpha]\neg A \wedge [\beta]\neg A \wedge [\alpha \cup \beta]\neg A))$. Thus $m \in V(\langle\alpha \cup \beta\rangle A \Leftrightarrow \langle\alpha\rangle A \vee \langle\beta\rangle A)$

(?) Let m be a node of the model $\mathcal{M}$ then:

– Either $m \in V(A)$ and $m \in V(B)$, that is $\exists m' = m$ such that $(m,m') \in R(A?)$ and $m' \in V(B)$ and thus $m \in V(\langle A?\rangle B)$, and then $m \in V(A \wedge B)$,

– or $m \notin V(A)$ or $m \notin V(B)$, that is $\forall m'.(m,m') \notin R(A?)$ or $m' \notin V(B)$ and thus $m \in [A?]\neg B$, and then $m \in V(\neg A \vee \neg B)$

In all possible cases, $m \in V((\langle A?\rangle B \wedge A \wedge B) \vee ([A?]\neg B \wedge (\neg A \vee \neg B)))$. Thus $m \in V(\langle A?\rangle B \Leftrightarrow A \wedge B)$

(*) Let m be a node of the model $\mathcal{M}$ then:

– Either $\exists k, m'$ such that $(m,m') \in R(\alpha^k)$ and $m' \in V(A)$, that is $m \in V(\langle\alpha^*\rangle A)$, and then either $k = 0$ thus $m \in V(A)$ or $k \geq 1$ and $\exists m''$ such that $(m,m'') \in R(\alpha)$ and $(m'',m') \in R(\alpha^{k-1})$ thus $m \in V(\langle\alpha\rangle\langle\alpha^*\rangle A)$,

– or $\forall k$, $\forall m'$, $(m,m') \notin R(\alpha^k)$ or $m' \notin V(A)$, that is $m \in V([\alpha^*]\neg A)$. In particular, $m \notin V(A)$ and $\forall m''$, either $(m,m'') \notin R(\alpha)$ or $\forall k'$, $\forall m^{(3)}$, $(m'',m^{(3)}) \notin R(\alpha^k)$ or $m^{(3)} \notin V(A)$, that is $m \in V(\neg A \wedge [\alpha][\alpha^*]\neg A)$.

In all possible cases, $m \in V((\langle\alpha^*\rangle A \wedge (A \vee \langle\alpha\rangle\langle\alpha^*\rangle A) \vee ([\alpha^*]\neg A \wedge \neg A \wedge [\alpha][\alpha^*]\neg A))$. Thus $m \in V(\langle\alpha^*\rangle A \Leftrightarrow A \vee \langle\alpha\rangle\langle\alpha^*\rangle A)$

(-) Let m be a node of the model $\mathcal{M}$ then:

– Either $m \in V(A)$ and then $\forall m', ((m,m') \notin R(\alpha)$ or $\exists m'' = m.$ such that $(m',m'') \in V(A))$. Thus $m \in V([\alpha]\langle\alpha^-\rangle A)$,

– or $m \notin V(A)$ and thus $m \in V(\neg A)$.

In all cases, $m \in V([\alpha]\langle\alpha^-\rangle A \vee \neg A)$ thus $m \in V(A \Rightarrow [\alpha]\langle\alpha^-\rangle A)$.

(Σ1) Let m be a node of the model $\mathcal{M}$, by definition of ν_Σ, $(m,\chi(c)) \in R(\nu_\Sigma)$ thus $m \in V(\langle\nu_\Sigma\rangle c)$

(Σ2) Let m be a node of the model $\mathcal{M}$ then:

– either $\chi(c) \notin V(A)$ and thus $m \in V(\langle\nu_\Sigma\rangle(c \wedge \neg A))$ but then $\forall m''.\ m'' \notin V(c) = \{\chi(c)\}$ or $m'' \notin V(A)$ thus $m \in V([\nu_\Sigma](\neg c \vee \neg A)$. Thus $m \in V([\nu_\Sigma](\neg c \vee \neg A) \wedge \langle\nu_\Sigma\rangle(c \wedge \neg A))$,

– or $\chi(c) \in V(A)$ and thus $\forall m', m' \notin V(c)$ or $m' \in V(A)$ thus $m \in V([\nu_\Sigma](\neg c \vee A)$. But then $\exists m'' = \chi(c)$ such that $m'' \in V(c \wedge A)$ and thus $m \in V(\langle\nu_\Sigma\rangle(c \wedge A))$ thus $m \in V([\nu_\Sigma](\neg c \vee A) \wedge \langle\nu_\Sigma\rangle(c \wedge A))$.

In all possible cases, $m \in V(([\nu_\Sigma](\neg c \vee \neg A) \wedge \langle\nu_\Sigma\rangle(c \wedge \neg A)) \wedge ([\nu_\Sigma](\neg c \vee A) \wedge \langle\nu_\Sigma\rangle(c \wedge A)))$ that is $m \in V(\langle\nu_\Sigma\rangle(c \wedge A) \Leftrightarrow [\nu_\Sigma](c \Rightarrow A))$

(ν_S1) Let m be a node of the model $\mathcal{M}$, S be a subset of Σ and c, d be elements of S. Then:

– Either $m \in V(c) = \{\chi(c)\}$ and, as $(\chi(c),\chi(d)) \in \chi(S)^2$, $(m,\chi(d)) \in R(\nu_S)$. Moreover as $\chi(d) \in V(d)$, $m \in V(\langle\nu_S\rangle d)$ and thus $m \in V(\neg c \vee \langle\nu_S\rangle d)$,

– or $m \notin V(c)$ and thus $m \in V(\neg c \vee \langle\nu_S\rangle d)$.

In all possible cases, $m \in V(\neg c \vee \langle\nu_S\rangle d)$ that is $m \in V(c \Rightarrow \langle\nu_S\rangle d)$

(ν_S2) Let m be a node of the model $\mathcal{M}$, S be a subset of Σ and c, d be such that $\{c,d\} \not\subseteq S$. Then:

– Either $c \notin S$ and then:
  - Either $m \in V(c) = \{\chi(c)\}$ which, as $\forall m'\ (\chi(c), m') \notin \chi(S)^2$, means $m \in V([\nu_S]\neg d)$ thus $m \in V(\neg c \vee [\nu_S]\neg d)$,
  - or $m \notin V(c)$ thus $m \in V(\neg c \vee [\nu_S]\neg d)$,
– or $c \in S$ and $d \notin S$ and then:
  - Either $m \in V(c) = \{\chi(c)\}$ which, as $(\chi(c), \chi(d)) \notin \chi(S)^2, \forall m'.(m, m') \notin R(\nu_S)$ or $m' \notin V(d)$ thus $m \in V(\neg c \vee [\nu_S]\neg d)$,
  - or $m \notin V(c)$ and thus $m \in V(\neg c \vee [\nu_S]\neg d)$.

In all possible cases, $m \in V(\neg c \vee [\nu_S]\neg d)$ that is $m \in V(c \Rightarrow [\nu_S]\neg d)$

(ν_S3) Let m be a node of the model $\mathcal{M}$ then:
– Either $\exists m', m''$ such that $(m, m') \in R(\nu_S)$, $(m', m'') \in R(\nu_S)$ and $m'' \in V(A)$, that is $m \in V(\langle\nu_S\rangle\langle\nu_S\rangle A)$, and then, as $(m, m'') \in \chi(S)^2$, $(m, m'') \in R(\nu_S)$ and thus $m \in V(\langle\nu_S\rangle A)$,
– or $\forall m', m''.\ (m, m') \notin R(\nu_S)$ or $(m', m'') \notin R(\nu_S)$ or $m'' \notin V(A)$. But then, $\forall m'.\ (m, m') \notin R(\nu_S)$ or $\forall m''.\ (m', m'') \notin R(\nu_S)$ or $m'' \notin V(A)$, that is $m \in V([\nu_S][\nu_S]\neg A)$.

In all possible cases, $m \in V([\nu_S][\nu_S]\neg A \vee \langle\nu_S\rangle A)$ that is $m \in V(\langle\nu_S\rangle\langle\nu_S\rangle A \Rightarrow \langle\nu_S\rangle A)$

(ν_S4) Let m be a node of the model $\mathcal{M}$ then:
– Either $m \in V(A)$ and then:
  - either $m \in \chi(S)$ and $\forall m'$ such that $(m, m') \in R(\nu_S)$ then $m' \in \chi(S)$ and thus $(m', m) \in R(\nu_S)$ thus $\forall m'.\ (m, m') \notin R(\nu_S)$ or $\exists m'' = m$ such that $(m', m'') \in R(\nu_S)$ and $m'' \in V(A)$, that is $m \in V([\nu_S]\langle\nu_S\rangle A)$,
  - or $m \notin \chi(S)$ and then $\forall m'.\ (m, m') \notin R(\nu_S)$, that is $m \in V([\nu_S]\langle\nu_S\rangle A)$
– or $m \notin V(A)$ and thus $m \in V(\neg A)$.

In all possible cases, $m \in V(\neg A \vee [\nu_S]\langle\nu_S\rangle A)$ that is $m \in V(A \Rightarrow [\nu_S]\langle\nu_S\rangle A)$

(ν_Σ1) Let m be a node of the model $\mathcal{M}$ then:
– Either $m \in V(A)$ and then $\exists m' = m$ such that $(m, m') \in R(\nu_\Sigma)$ and $m' \in V(A)$ thus $m \in V(\langle\nu_\Sigma\rangle A)$,
– or $m \notin V(A)$ and then $m \in V(\neg A)$

In all possible cases, $m \in V(\neg A \vee \langle\nu_\Sigma\rangle A)$ that is $m \in V(A \Rightarrow \langle\nu_\Sigma\rangle A)$.

(ν_Σ2) Let m be a node of the model $\mathcal{M}$ then:
– Either $m \in V(\langle\alpha\rangle A)$ and then $\exists m'$ such that $(m, m') \in R(\alpha)$ and $m' \in V(A)$ but then $(m, m') \in R(\nu_\Sigma)$ that is $m \in V(\langle\nu_\Sigma\rangle A)$,
– or $m \notin V(\langle\alpha\rangle A)$ and then $m \in V([\alpha]\neg A)$

In all possible cases, $m \in V([\alpha]\neg A \vee \langle\nu_\Sigma\rangle A)$ that is $m \in V(\langle\alpha\rangle A \Rightarrow \langle\nu_\Sigma\rangle A)$

($\Sigma_2$1) Let m be a node of the model $\mathcal{M}$, $\phi \in \Phi_0$ then:
– Either $m \in V(c) = \{\chi(c)\}$ and then as $V(\phi) \subseteq \chi(\Sigma_1)$ and $\chi(\Sigma_1) \cap \chi(\Sigma_2) = \emptyset$, $m \notin V(\phi)$ and thus $m \in V(\neg\phi)$,
– or $m \notin V(c)$ and thus $m \in V(\neg c)$.

In all possible cases, $m \in V(\neg c \vee \neg\phi)$ that is $m \in V(c \Rightarrow \neg\phi)$

($\Sigma_2$2) Let m be a node of the model $\mathcal{M}$, $\alpha \in \Pi_0$ then:
– Either $m \in V(c)$ and then as $R(\alpha) \subseteq \chi(\Sigma_1)^2$ and $\chi(\Sigma_1) \cap \chi(\Sigma_2) = \emptyset$, $\forall m'$, $(m, m') \notin R(\alpha)$ and $(m', m) \notin R(\alpha)$ thus $m \in V([\alpha]\bot \wedge [\alpha^-]\bot)$,
– or $m \notin V(c)$ and thus $m \in V(\neg c)$.

In all possible cases, $m \in V(\neg c \vee [\alpha]\bot \wedge [\alpha^-]\bot)$ that is $m \in V(c \Rightarrow [\alpha]\bot \wedge [\alpha^-]\bot)$

(MP) Assume $\vdash A$ and $\vdash A \Rightarrow B$, then $\forall m$, $m \in V(A)$ and $m \in V(\neg A \vee B)$ thus $m \in V(B)$. That is $\vdash B$.

(Ind) Assume $\vdash [\gamma][\alpha^k]A$, for all $k\langle\omega$, then $\forall m$, $m \in V([\gamma][\alpha^k]A$, for all k, that is $\forall m'$ such that $(m, m') \in R(\gamma)$, $m' \in V([\alpha^k]A)$ for all k. As $\forall m''$, $(m', m'') \in R(\alpha^k)$ for some k or $(m', m'') \notin \bigcup_k R(\alpha^k) = R(\alpha^*)$, that is $(m', m'') \notin R(\alpha^*)$ or $m'' \in V(A)$. Thus $m' \in V([\alpha^*]A)$ and thus $m \in V([\gamma][\alpha^*]A)$. That is $\vdash [\gamma][\alpha^*]A$

(Cov) Assume $\vdash [\gamma]\neg c$, for all $c \in \Sigma$, then $\forall m$, $m \in V([\gamma]\neg c)$. Thus $\forall m'$, $(m, m') \notin R(\gamma)$ or $m' \notin V(c)$ for all c. But, as $\chi(\Sigma) = M$, $\exists c'$ such that $m' = \chi(c')$ Thus $\forall m'$, $(m, m') \notin R(\gamma)$. Thus $m \in V([\gamma]\bot)$. That is $\vdash [\gamma]\bot$.

(Nec) Assume $\vdash A$, then $\forall m$, $m \in V(A)$, then $\forall m', m''$, $(m', m'') \notin R(\nu_\Sigma)$ or $m'' \in V(A)$, thus $m' \in V([\nu_\Sigma]A)$. That is $\vdash [\nu_\Sigma]A$

6.2 Completeness

Theorem 3 (Completeness) . *Let A be a $\mathcal{C}2\mathcal{PDL}$ formula, if $\vDash A$ then $\vdash A$.*

This proof is much more involved and requires several definitions and lemmata.

Definition 15. *A logic (over $\mathcal{DS}$) is any set of $\mathcal{C}2\mathcal{PDL}$ formulae L such that:*

- *L contains all axioms of $\mathcal{DS}$*
- *L is closed under (MP), (Ind), (Cov) and (Nec).*

Definition 16. *Let L be a logic, an L-theory is any set $T \subseteq \Phi$ such that:*

- *$L \subseteq T$*
- *T is closed under (MP), (Ind) and (Cov).*

Definition 17. *A logic L (resp. a theory T) is consistent if $\bot \notin L$ (resp. $\bot \notin T$).*

Definition 18. *A formula A is said to be closed if $\exists B$ such that $\vdash A \leftrightarrow \langle\nu_\Sigma\rangle B$.*

Definition 19. *A logic L (resp. a theory T) is maximal if $\forall A \in \Phi$ such that A is closed, either $A \notin L$ or $A \in L$ (resp. $\forall A \in \Phi$, either $A \notin T$ or $A \in T$).*

Definition 20. *By $log(\Gamma, A)$ (resp. $th(\Gamma, A)$), we denote the least logic (resp. theory) containing $\Gamma \cup \{A\}$.*

Lemma 4 (Deduction lemma for theories). *Let T be an L-theory, $A, B \in \Phi$, then $A \Rightarrow B \in T$ iff $B \in th(T, A)$.*

Proof. $\Leftarrow$ Assume $B \in th(T, A)$ and let T_0 be $\{D | A \Rightarrow D \in T\}$. As $A \Rightarrow A \in T$, $A \in T_0$. Let's prove that T_0 is an L-theory:
- $\forall D \in T$, $\neg A \vee D \in T$ and thus $L \subseteq T \subseteq T_0$.

– Assume $D_0 \in T_0$ and $D_0 \Rightarrow D_1 \in T_0$. As $A \vee \neg A \in T$, $(A \wedge (D \vee \neg D)) \vee \neg A \in T$ thus $(A \wedge \neg D) \vee \neg A \vee (A \wedge D) \in T$ that is $(A \Rightarrow D) \Rightarrow (A \Rightarrow (A \wedge D)) \in T$. By replacing D with D_0, we obtain $(A \Rightarrow D_0) \Rightarrow (A \Rightarrow (A \wedge D_0)) \in T$ and, as $A \Rightarrow D_0 \in T$ and T is closed under (MP), $A \Rightarrow (A \wedge D_0) \in T$. By replacing D with $D_0 \Rightarrow D_1$, we obtain $(A \Rightarrow (D_0 \Rightarrow D_1)) \Rightarrow (A \Rightarrow (A \wedge (D_0 \Rightarrow D_1))) \in T$, $A \Rightarrow (D_0 \Rightarrow D_1) \in T$ and, as T is closed under (MP), $A \Rightarrow (A \wedge (D_0 \Rightarrow D_1)) \in T$. Similarly, $(A \wedge \neg D_1) \vee \neg A \vee \neg D_1 \in T$ thus $(A \wedge \neg D_1 \wedge (D_0 \vee \neg D_0)) \vee \neg A \vee \neg D_1 \in T$ thus $(A \wedge (\neg A \vee (D_0 \wedge \neg D_1))) \vee (A \wedge (\neg A \vee \neg D_0)) \vee \neg A \vee \neg D_1 \in T$ that is $(A \Rightarrow (A \wedge (D_0 \Rightarrow D_1))) \Rightarrow ((A \Rightarrow (A \wedge D_0)) \Rightarrow (A \wedge D_1)) \in T$. Then, as $(A \Rightarrow (A \wedge (D_0 \Rightarrow D_1))) \Rightarrow ((A \Rightarrow (A \wedge D_0)) \Rightarrow (A \wedge D_1)) \in T$, $A \Rightarrow (A \wedge (D_0 \Rightarrow D_1)) \in T$ and T is closed by (MP), $(A \Rightarrow (A \wedge D_0)) \Rightarrow (A \wedge D_1) \in T$. As $(A \Rightarrow (A \wedge D_0)) \Rightarrow (A \wedge D_1) \in T$, $(A \Rightarrow (A \wedge D_0)) \in T$ and T is closed under (MP), $A \wedge D_1 \in T$ thus $D_1 \in T_0$ thus T_0 is stable by (MP).
– Assume $\forall k \langle \omega, [\gamma][\alpha^k]D \in T_0$, then $\forall k \langle \omega, A \Rightarrow [\gamma][\alpha^k]D \in T$ and thus $\forall k \langle \omega, [A?;\gamma][\alpha^k]D \in T$. But, as T is stable by (Ind), $[A?;\gamma][\alpha^*]D \in T$ and thus $A \Rightarrow [\gamma][\alpha^*]D \in T$ that is $[\gamma][\alpha^*]D \in T_0$. Thus T_0 is stable by (Ind).
– Assume $\forall c \in \Sigma, [\gamma]\neg c \in T_0$, then $\forall c \in \Sigma, A \Rightarrow [\gamma]\neg c \in T$ and thus $\forall c \in \Sigma, [A?;\gamma]\neg c \in T$. But, as T is stable by (Cov), $[A?;\gamma]\bot \in T$ and thus $A \Rightarrow [\gamma]\bot \in T$ that is $[\gamma]\bot \in T_0$. Thus T_0 is stable by (Cov).

Thus T_0 is an L-theory. As $th(T, A)$ is the smallest theory containing $T \cup A$ and T_0 contains $T \cup A$, $th(T, A) \subseteq T_0$ but, as $B \in th(T, A)$ then $B \in T_0$ that is $A \Rightarrow B \in T$.

$\Rightarrow$ Assume $A \Rightarrow B \in T$ then, as $th(T, A)$ is stable by (MP) and $A \in th(T, A)$, $B \in th(T, A)$.

Lemma 5. *If $\forall B$, ($B \in \Gamma \cup A$ implies $[\nu_\Sigma]B \in th(\Gamma, A)$), then $th(\Gamma, A)$ is a logic.*

Proof. As $th(\Gamma, A)$ is a theory, $L \subseteq th(\Gamma, A)$ and thus all axioms of $\mathcal{DS}$ are contained in $th(\Gamma, A)$ and $th(\Gamma, A)$ is closed under (MP), (Cov) and (Ind).

Assume $C \in th(\Gamma, A)$. We prove by induction that $[\nu_\Sigma]C \in th(\Gamma, A)$:

– If $C \in Gamma \cup A$, then $[\nu_\Sigma]C \in th(\Gamma, A)$.
– If C results from the application of (MP) on $D \in th(\Gamma, A)$ and $D \Rightarrow C \in th(\Gamma, A)$, from the induction hypothesis, $[\nu_\Sigma]D \in th(\Gamma, A)$ and $[\nu_\Sigma]D \Rightarrow C \in th(\Gamma, A)$. But $th(\Gamma, A)$ contains the axiom $\Box$ where $A = D$, $B = C$ and $\alpha = \nu_S$, that is $[\nu_\Sigma](D \Rightarrow C) \Rightarrow ([\nu_\Sigma]D \Rightarrow [\nu_\Sigma]C) \in th(\Gamma, A)$. Then, as $th(\Gamma, A)$ is stable under (MP) with $A = [\nu_\Sigma](D \Rightarrow C)$ and $B = ([\nu_\Sigma]D \Rightarrow [\nu_\Sigma]C)$, $[\nu_\Sigma]D \Rightarrow [\nu_\Sigma]C \in th(\Gamma, A)$. Then, as $th(\Gamma, A)$ is stable under (MP) with $A = [\nu_\Sigma]D$ and $B = [\nu_\Sigma]C$, $[\nu_\Sigma]C \in th(\Gamma, A)$.
– If C results from the application of (Ind) on $\forall k \langle \omega . [\gamma][\alpha^k]D \in th(\Gamma, A)$ then $C = [\gamma][\alpha^*]D$. From the induction hypothesis, $[\nu_\Sigma;\gamma][\alpha^k]D \in th(\Gamma, A)$. Then, as $th(\Gamma, A)$ is stable under (Ind), $[\nu_\Sigma;\gamma][\alpha^*]D \in th(\Gamma, A)$ and thus $[\nu_\Sigma]C \in th(\Gamma, A)$.

– If C results from the application of (Cov) on $\forall c \in \Sigma.[\gamma]\neg c \in th(\Gamma, A)$ then $C = [\gamma]\bot$. From the induction hypothesis, $\forall c \in \Sigma.[\nu_\Sigma;\gamma]\neg c \in th(\Gamma, A)$. Then, as $th(\Gamma, A)$ is stable under (Cov), $[\nu_\Sigma;\gamma]\bot \in th(\Gamma, A)$ and thus $[\nu_\Sigma]C \in th(\Gamma, A)$.

Thus $th(\Gamma, A)$ is stable under (Nec), thus $th(\Gamma, A)$ is a logic.

Lemma 6. *Let L be a logic and A be a closed formula, $th(L, A) = log(L, A)$*

Proof. $\subseteq$ Let $B \in L \cup A$, if $B \in L$ by the stability of L under (MP), $[\nu_\Sigma]B \in L \subseteq th(L, A)$.
Otherwise, $B = A$ then, as A is closed, $\exists C$ such that $B \leftrightarrow \langle\nu_\Sigma\rangle C \in L$ and thus $B \Rightarrow \langle\nu_\Sigma\rangle C \in L \subseteq th(L, A)$. Then, as $th(L, A)$ is stable under (MP), $B \in th(L, A)$ and $B \Rightarrow \langle\nu_\Sigma\rangle C$ $inth(L, A)$, $\langle\nu_\Sigma\rangle C \in th(L, A)$. Then, as $th(L, A)$ contains $(\nu_S 4)$ with $S = \Sigma$ and $A = \langle\nu_\Sigma\rangle C$, it yields $\langle\nu_\Sigma\rangle C \Rightarrow [\nu_\Sigma]\langle\nu_\Sigma\rangle\langle\nu_\Sigma\rangle C$ and thus $[\nu_\Sigma]\langle\nu_\Sigma\rangle\langle\nu_\Sigma\rangle C \in th(\Gamma, A)$. Then, as $th(L, A)$ contains $(\nu_S 3)$ with $S = \Sigma$ and $A = C$, it yields $\langle\nu_\Sigma\rangle\langle\nu_\Sigma\rangle C \Rightarrow \langle\nu_\Sigma\rangle C \in th(\Gamma, A)$. Thus $[\nu_\Sigma]\langle\nu_\Sigma\rangle C \in th(\Gamma, A)$ and thus $[\nu_\Sigma]B \in th(\Gamma, A)$. Thus, from Lemma 5, $th(L, A)$ is a logic containing $L \cup A$. Thus $log(L, A) \subseteq th(L, A)$.
$\supseteq$ By definition, a logic is a theory and $log(L, A)$ contains $L \cup A$ thus $th(L, A) \subseteq log(L, A)$.

Lemma 7 (Deduction lemma for logics). *Let L be a logic and A be a closed formula. Then $A \Rightarrow B \in L$ iff $B \in log(L, A)$*

Proof. As L is an L-theory, from Lemma 4, $A \Rightarrow B \in L$ iff $B \in th(L, A)$. But, from Lemma 6, $log(L, A) = th(L, A)$ and thus $A \Rightarrow B \in L$ iff $B \in log(L, A)$

Lemma 8 (Separation theorem for theories). *Let T be a theory, $A \notin T$. Then there exists a maximal theory T^* such that $T \subseteq T^*$ and $A \notin T^*$.*

Proof. Let $T_0 = th(T, \neg A)$. As $A \notin T$, $\neg A \Rightarrow \bot \notin T$. Then, from Lemma 4, $\bot \notin T_0$ and thus T_0 is consistent. Let $B_0, B_1, \ldots,$ be an enumeration of Φ. By induction on n, we construct a chain $T_0 \subseteq T_1 \subseteq \ldots$ of consistent theories. Their union will yield the required T^*. The induction hypothesis is that T_n is a consistent theory. It is the case for T_0.

– If $th(T_n, B_n)$ is consistent, then $T_{n+1} = th(T_n, B_n)$ is consistent.
– If $th(T_n, B_n)$ is not consistent, then $\bot \in th(T_n, B_n)$ and, from Lemma 4, $B_n \Rightarrow \bot \in T_n$ and thus $\not B_n \in T_n$. Then:
  • Either $B_n \neq [\gamma]0$ and $B_n \neq [\gamma][\alpha^*]A$ and then $T_{n+1} = T_n$ is consistent,
  • or $B_n = [\gamma]0$. Let $B_{n,c} = [\gamma]\neg c$, if $\forall c \in \Sigma$, $B_{n,c} \in T_n$ then, because T_n is stable by (Cov), and $\neg B_n \in T_n$ then T_n is inconsistent which, due to the induction hypothesis, is not the case. Thus $\exists c \in \Sigma$ such that $B_{n,c} \notin T_n$. Then, from Lemma 4, $T_{n+1} = th(T_n, \neg B_{n,c})$ is consistent,
  • or $B_n = [\gamma][\alpha^*]A$. Let $B_{n,k} = [\gamma][\alpha^k]A$, if $\forall k\langle\omega$, $B_{n,k} \in T_n$ then, because T_n is stable by (Ind), and $\neg B_n \in T_n$ T_n is inconsistent which, due to the induction hypothesis, is not the case. Thus $\exists k\langle\omega$ such that $B_{n,k} \notin T_n$. Then, from Lemma 4, $T_{n+1} = th(T_n, \neg B_{n,k})$ is consistent.

Let $T^* = \bigcup\{T_n | n\langle\omega\}$. We have:

- $L \subseteq T \subseteq T_0 \subseteq T^*$
- Let C_0, C_1 be such that $C_0 \in T^*$ and $C_0 \Rightarrow C_1 \in T^*$, then $\exists k_0, k_1 \langle\omega$ such that $C_0 \in T_{k_0}$ and $C_0 \Rightarrow C_1 \in T_{k_1}$ that is $C_0 \in T_{max(k_0,k_1)}$ and $C_0 \Rightarrow C_1 \in T_{max(k_0,k_1)}$. As $T_{max(k_0,k_1)}$ is closed under (MP), $C_1 \in T_{max(k_0,k_1)}$ and thus $C_1 \in T^*$. Then T^* is closed under (MP).
- As $\forall k$, $\neg A \in T_0 \subseteq T_k$ and T_k is consistent, $A \notin T^*$
- Assume $\bot \in T^*$ then, as $\bot \Rightarrow A$ is a boolean tautology, both $\bot \in T^*$ and $\bot \Rightarrow A \in T^*$. As T^* is closed under (MP), then $A \in T^*$. As it is not the case, T^* is consistent.
- By construction, $\forall B \in \Phi$ either $B \in T^*$ or $\neg B \in T^*$
- Let $D_c = [\gamma]\neg c$ and $D = [\gamma]0 = B_n$. Suppose $\forall c \in \Sigma. D_c \in T^*$ and $D \notin T^*$ then, by construction, for some $c_0 \in \Sigma$, $\neg B_{n,c_0} \in T_{n+1} \subseteq T^*$ but then $\neg B_{n,c_0} \in T^*$ and $B_{n,c_0} \in T^*$ which is impossible as T^* is consistent. Thus T^* is closed under (Cov).
- Let $D_k = [\gamma][\alpha^k]A$ and $D = [\gamma][\alpha^*]A = B_n$. Suppose $\forall k\langle\omega. D_k \in T^*$ and $D \notin T^*$ then, by construction, for some $k_0 \in \Sigma$, $\neg B_{n,k_0} \in T_{n+1} \subseteq T^*$ but then $\neg B_{n,k_0} \in T^*$ and $B_{n,k_0} \in T^*$ which is impossible as T^* is consistent. Thus T^* is closed under (Ind).

Thus T^* is a maximal theory and $T \subseteq T^*$ and $A \notin T^*$.

Definition 21. $\mathcal{L}_T = \{A | [\nu_\Sigma]A \in T\}$.

Lemma 9. *If T is a maximal L-theory, then $\mathcal{L}_T$ is a maximal logic and $\mathcal{L}_T$ is the greatest logic included in T.*

Proof.
- Let $\mathcal{A}$ be an axiom of $\mathcal{DS}$ then $\mathcal{A} \in L$ as L is a logic. As L is closed under (MP), $[\nu_\Sigma]\mathcal{A} \in L$. As $L \subseteq T$, $[\nu_\Sigma]\mathcal{A} \in T$ and thus $\mathcal{A} \in \mathcal{L}_T$
- Assume $C \in \mathcal{L}_T$ and $C \Rightarrow D \in \mathcal{L}_T$ then $[\nu_\Sigma]C \in T$ and $[\nu_\Sigma](C \Rightarrow D) \in T$. As $L \subseteq T$ and L contains $(\Box)$ and T is closed under (MP), $[\nu_\Sigma]C \Rightarrow [\nu_\Sigma]D \in T$ and then, by (MP), $[\nu_\Sigma]D \in T$ and thus $D \in \mathcal{L}_T$. Thus $\mathcal{L}_T$ is closed under (MP).
- Assume $\forall k\langle\omega.[\gamma][\alpha^k]A \in \mathcal{L}_T$, then $\forall k\langle\omega.[\nu_\Sigma;\gamma][\alpha^k]A \in T$. As T is closed under (Ind), $[\nu_\Sigma;\gamma][\alpha^*]A \in T$ and thus $[\gamma][\alpha^k]A \in \mathcal{L}_T$. Thus $\mathcal{L}_T$ is closed under (Ind).
- Assume $\forall c \in \Sigma.[\gamma]\neg c \in \mathcal{L}_T$, then $\forall c \in \Sigma.[\nu_\Sigma;\gamma]\neg c \in T$. As T is closed under (Cov), $[\nu_\Sigma;\gamma]\bot \in T$ and thus $[\gamma]\bot \in \mathcal{L}_T$. Thus $\mathcal{L}_T$ is closed under (Cov).
- Assume $C \in \mathcal{L}_T$ then $[\nu_\Sigma]C \in T$. Assume $\langle\nu_\Sigma\rangle[\nu_\Sigma]C \in T$ then as $L \subseteq T$ and L contains $(\nu_S 3)$ with $S = \Sigma$ and $A = \neg C$, $[\nu_\Sigma]C \Rightarrow [\nu_\Sigma][\nu_\Sigma]C \in T$. As T is closed under (MP), $[\nu_\Sigma][\nu_\Sigma]C \in T$ and thus $[\nu_\Sigma]C \in T$. Thus $\mathcal{L}_T$ is closed under (Nec).
- Let C be a closed formula. As T is maximal:
  - either $C \in T$ that is $existsB.\langle\nu_\Sigma\rangle B \in T$. As $L \subseteq T$ and L contains $(\nu_S 4)$ for $S = \Sigma$ and $A = \langle\nu_\Sigma\rangle B$, $\langle\nu_\Sigma\rangle B \Rightarrow [\nu_\Sigma]\langle\nu_\Sigma\rangle\langle\nu_\Sigma\rangle B \in T$ and then, by (MP), $[\nu_\Sigma]\langle\nu_\Sigma\rangle\langle\nu_\Sigma\rangle B \in T$. As $L \subseteq T$ and L contains $(\nu_S 3)$

for $S = \Sigma$ and $A = B$ and L is closed under (Nec), $[\nu_\Sigma]\langle\nu_\Sigma\rangle\langle\nu_\Sigma\rangle B \Rightarrow [\nu_\Sigma]\langle\nu_\Sigma\rangle B \in T$ and then, by (MP), $[\nu_\Sigma]\langle\nu_\Sigma\rangle B \in T$ and thus $\langle\nu_\Sigma\rangle B \in T$ that is $C \in \mathcal{L}_T$,
  - or $\neg C \in T$ that is $\exists B.[\nu_\Sigma]\neg B \in T$ and thus $\neg B \in \mathcal{L}_T$. But, as $\mathcal{L}_T$ is closed under (Nec), $[\nu_\Sigma]\neg B \in \mathcal{L}_T$ and thus $\neg C \in \mathcal{L}_T$.

  That is $\mathcal{L}_T$ is maximal.
- Let L' be a logic such that $L' \subseteq T$. Let $A' \in L'$. As L' is closed under (Nec), $[\nu_\Sigma]A' \in L'$ and thus $[\nu_\Sigma]A' \in T$ that is $A' \in \mathcal{L}_T$. Thus $\forall L' \subseteq T, L' \subseteq \mathcal{L}_T$.
- Let $C \in \mathcal{L}_T$ then $[\nu_\Sigma]C \in T$. As $L \subseteq T$ and L contains $(\nu_\Sigma 1)$ with $A = \neg C$, $\neg C \Rightarrow \langle\nu_\Sigma\rangle\neg C \in T$ or, written in another way, $[\nu_\Sigma]C \Rightarrow C \in T$. As T is closed under (MP), $C \in T$. Thus $\mathcal{L}_T \subseteq T$.

  Thus $\mathcal{L}_T$ is a maximal logic and it is the greatest included in T.

Lemma 10 (Separation lemma for logics). *Let L be a logic, $A \notin L$. Then there exists a maximal logic L^* such that $L \subseteq L^*$ and $A \notin L^*$.*

Proof. L is an L-theory thus, from Lemma 8, there exists a maximal theory T^* such that $L \subseteq T^*$ and $A \notin T^*$. Then, from Lemma 9, $\mathcal{L}_{T^*}$ is a maximal logic such that $\mathcal{L}_{T^*} \subseteq T^*$. Assume $A \in \mathcal{L}_{T^*}$ then $A \in T^*$ which is not the case. Thus $\mathcal{L}_{T^*}$ is a maximal L-logic, that is $L \subseteq \mathcal{L}_{T^*}$, and $A \notin \mathcal{L}_{T^*}$.

Lemma 11 (Lindenbaum lemma). *If L is a consistent logic (resp. T is a consistent theory) then there exists a maximal consistent logic L^* (resp. a maximal consistent theory T^*) such that $L \subseteq L^*$ (resp. $T \subseteq T^*$).*

Proof. It is a direct consequence of the Separation lemmata with $A = \bot$.

Lemma 12. *If L is a consistent logic, then L has a model.*

Proof. From Lemma 11, there exists a maximal consistent logic L^* such that $L \subseteq L^*$. Let's define $c \sim d = \langle\nu_\Sigma\rangle(c \wedge d) \in L^*$.

- As L contains $(\Sigma 1)$, so does L^* and thus $\langle\nu_\Sigma\rangle(c \wedge c) \in L^*$. Thus $c \sim c$ that is $\sim$ is reflexive.
- Assume $c \sim d$. As $\wedge$ is commutative, $\langle\nu_\Sigma\rangle(c \wedge d) \leftrightarrow \langle\nu_\Sigma\rangle(d \wedge c) \in L^*$ and thus, by (MP), $d \sim c$ that is $\sim$ is symmetric.
- Assume $c \sim d$ and $d \sim e$. Then, as L^* contains $(\Sigma 2)$ and is closed under (MP), $[\nu_\Sigma](d \Rightarrow e) \in L^*$ and $[\nu_\Sigma](c \Rightarrow d) \in L^*$. Then, as $[\nu_\Sigma](c \Rightarrow d) \Rightarrow ([\nu_\Sigma](d \Rightarrow e) \Rightarrow ([\nu_\Sigma](c \Rightarrow d) \wedge [\nu_\Sigma](d \Rightarrow e)))$ is a boolean tautology and thus in L^*, by applying (MP) twice, $[\nu_\Sigma](c \Rightarrow d) \wedge [\nu_\Sigma](d \Rightarrow e) \in L^*$. But, as $([\nu_\Sigma](c \Rightarrow d) \wedge [\nu_\Sigma](d \Rightarrow e)) \Rightarrow [\nu_\Sigma]((c \Rightarrow d) \wedge (d \Rightarrow e))$ is a boolean tautology and thus in L^*, by applying (MP), $[\nu_\Sigma]((c \Rightarrow d) \wedge (d \Rightarrow e)) \in L^*$ and thus $[\nu_\Sigma](c \Rightarrow e) \in L^*$. Then, from $(\Sigma 2)$ using (MP), $\langle\nu_\Sigma\rangle(c \wedge e) \in L^*$ and thus $c \sim e$ that is $\sim$ is transitive.

Thus $\sim$ is an equivalence relation. Let $[c] = \{d | c \sim d\}$, we construct $M_\sim = \Sigma_{/\sim}$, $\chi_\sim(c) = [c]$, $V_\sim(A) = \{[c] | \langle\nu_\Sigma\rangle(c \wedge A) \in L^*\}$ and $R^*_\sim(\alpha) = \{([c],[d]) | \langle\nu_\Sigma\rangle(c \wedge \langle\alpha\rangle d) \in L^*\}$. Let's show that $\mathcal{M} = (M_\sim, \chi_\sim, V_\sim, R_\sim)$ is a model.

– $\chi_\sim$ is obviously onto.
– Assume $m \in \chi_\sim(\Sigma_2) \cap \chi_\sim(\Sigma_1)$ that is $\exists c_2 \in \Sigma_2, c_1 \in \Sigma_1$ such that $[c_2] = [c_1]$ that is $\langle\nu_\Sigma\rangle c_2 \wedge c_1 \in L^*$. As $\Sigma_0 \cup \Sigma_1 = \emptyset$, $c_0 \notin \Sigma_1$ and thus, from $(\Sigma_S 2)$, $c_0 \Rightarrow [\nu_{\Sigma_1}]\neg c_1$. Meanwhile, from $(\Sigma_S 1)$, $c_1 \Rightarrow \langle\nu_\Sigma\rangle c_1$. Thus $\langle\nu_\Sigma\rangle(\langle\nu_{\Sigma_1}\rangle c_1 \wedge [\nu_\Sigma]\neg c_1) \in L^*$ which is false. As L^* is consistent, $\chi_\sim(\Sigma_2) \cap \chi_\sim(\Sigma_1) = \emptyset$.
– Let $i \in \Sigma$, then $V_\sim(i) = \{[c] | \langle\nu_\Sigma\rangle(c \wedge i) \in L^*\} = \{[i]\} = \{\chi_\sim(i)\}$.
– Let $\phi_0 \in \Phi_0$, assume $\exists c_2 \in \Sigma_2$ such that $[c_2] \in V_\sim(\phi_0)$ then $\langle\nu_\Sigma\rangle(c_2 \wedge \phi_0) \in {}^*$. As $\Sigma_2 1$ is in L^*, $c_2 \Rightarrow \neg\phi_0$ and thus $\langle\nu_\Sigma\rangle(\neg\phi_0 \wedge \phi_0) \in {}^*$ which is impossible as L^* is consistent. Thus $V_\sim(\phi_0) \in \mathcal{P}(\chi_\sim(\Sigma_1))$.
– • Assume $[c] \in \overline{V_\sim(A)}$ then $\langle\nu_\Sigma\rangle(c \wedge A) \notin L^*$. As L^* is maximal, $[\nu_\Sigma](c \Rightarrow \neg A) \in L^*$. As $(\Sigma 2)$ is in L^* so is $\langle\nu_\Sigma\rangle(c \wedge \neg A) \notin L^*$ and thus $[c] \in V_\sim(\neg A)$. Thus $\overline{V_\sim(A)} \subseteq V_\sim(\neg A)$.
  • Otherwise $[c] \in V_\sim(A)$ then $\langle\nu_\Sigma\rangle(c \wedge A) \in L^*$. If $[c] \in V_\sim(\neg A)$, then $\langle\nu_\Sigma\rangle(c \wedge \neg A) \in L^*$. As $(\Sigma 2)$ is in L^* so is $[\nu_\Sigma](c \Rightarrow \neg A) \in L^*$ and thus $\langle\nu_\Sigma\rangle(A \wedge \neg A) \in L^*$. As L^* is consistent, this is impossible and thus $V_\sim(\neg A) \subseteq \overline{V_\sim(A)}$.
  Thus $V_\sim(\neg A) = \overline{V_\sim(A)}$.
– • Assume $[c] \in V_\sim(A) \cup V_\sim(B)$ then:
    ∗ Either $[c] \in V_\sim(A)$ and then $\langle\nu_\Sigma\rangle(c \wedge A) \in L^*$ and thus $\langle\nu_\Sigma\rangle(c \wedge (A \vee B)) \in L^*$. Thus $[c] \in V_\sim(A \vee B)$,
    ∗ or $[c] \in V_\sim(B)$ and then $\langle\nu_\Sigma\rangle(c \wedge B) \in L^*$ and thus $\langle\nu_\Sigma\rangle(c \wedge (A \vee B)) \in L^*$. Thus $[c] \in V_\sim(A \vee B)$.
  Thus $V_\sim(A) \cup V_\sim(B) \subseteq V_\sim(A \vee B)$
  • Otherwise $[c] \notin V_\sim(A) \cup V_\sim(B)$ that is $[c] \in \overline{V_\sim(A) \cup V_\sim(B)} = (\overline{V_\sim(A)}) \cap (\overline{V_\sim(B)}$. Thus $[c] \in \overline{V_\sim(A)}$ that is $[c] \in V_\sim(\neg A)$ from the previous point. Thus $\langle\nu_\Sigma\rangle(c \wedge \neg A) \in L^*$. Similarly, $[c] \in \overline{V_\sim(A)}$ and thus $\langle\nu_\Sigma\rangle(c \wedge \neg B) \in L^*$. From $(\Sigma 2)$, $[\nu_\Sigma](c \Rightarrow \neg B) \in L^*$ and thus $\langle\nu_\Sigma\rangle(c \wedge \neg(A \vee B)) \in L^*$. Thus $[c] \in V_\sim(\neg(A \vee B))$ that is $[c] \in \overline{V_\sim(A \vee B)}$ and thus $[c] \notin V_\sim(A \vee B)$. Thus $V_\sim(A \vee B) \subseteq V_\sim(A) \cup V_\sim(B)$
  Thus $V_\sim(A \vee B) = V_\sim(A) \vee V_\sim(B)$.
– • Assume $[c] \in \{s | \exists [d] \in M_\sim.((s,[d]) \in R_\sim(\alpha) \wedge [d] \in V_\sim(A)\}$ then $([c],[d]) \in R_\sim(\alpha)$ and $[d] \in V_\sim(A)$. Thus $\langle\nu_\Sigma\rangle(c \wedge \langle\alpha\rangle d) \in L^*$ and $\langle\nu_\Sigma\rangle(d \wedge A) \in L^*$. As $(\Sigma 2)$ in L^*, $[\nu_\Sigma](d \Rightarrow A) \in L^*$ and thus $\langle\nu_\Sigma\rangle(c \wedge \langle\alpha\rangle A) \in L^*$ that is $[c] \in V_\sim(\langle\alpha\rangle A)$. Thus $\{s | \exists [d] \in M_\sim.((s,[d]) \in R_\sim(\alpha) \wedge [d] \in V_\sim(A)\} \subseteq V_\sim(\langle\alpha\rangle A)$.
  • Otherwise $[c] \notin \{s | \exists [d] \in M_\sim.((s,[d]) \in R_\sim(\alpha) \wedge [d] \in V_\sim(A)\}$ then $\forall [d]$, $(([s],[d]) \notin R_\sim(\alpha)$ or $[d] \notin V_\sim(A))$ that is $\forall [d]$, $(\langle\nu_\Sigma\rangle(c \wedge \langle\alpha\rangle d) \notin L^*$ or $\langle\nu_\Sigma\rangle(d \wedge A) \notin L^*)$. Then, by maximality of L^*, $\forall [d]$, $(\neg\langle\nu_\Sigma\rangle(c \wedge \langle\alpha\rangle d) \in L^*$ or $\neg\langle\nu_\Sigma\rangle(d \wedge A) \in L^*$ that is $\forall [d]$, $([\nu_\Sigma](c \Rightarrow [\alpha]\neg d) \in L^*$ or $[\nu_\Sigma](d \Rightarrow \neg A) \in L^*$. Then $\forall [d]$, $([\nu_\Sigma](c \Rightarrow [\alpha](\neg d \vee \neg A)) \in L^*$ or, by $(\nu_\Sigma 2)$ and $(\nu_S 3)$, $[\nu_\Sigma; c?; \alpha](d \Rightarrow \neg A)) \in L^*$ that is $[\nu_\Sigma]c \Rightarrow [\alpha](\neg c \vee (d \Rightarrow \neg A)) \in L^*$. Thus $\forall [d]$, $[\nu_\Sigma; c?; \alpha; A?]\neg d \in L^*$. As L^* is stable under (Cov), $[\nu_\Sigma; c?; \alpha; A?]\bot \in L^*$ that is $[\nu_\Sigma](c \Rightarrow [\alpha]\neg A) \in L^*$. By $(\Sigma 2)$, $\langle\nu_\Sigma\rangle(c \wedge [\alpha]\neg A) \in L^*$ and thus $[c] \in V_\sim([\alpha]\neg A)$ that is $[c] \notin V_\sim(\langle\alpha\rangle A)$. Thus $V_\sim(\langle\alpha\rangle A) \subseteq \{s | \exists [d] \in M_\sim.((s,[d]) \in R_\sim(\alpha) \wedge [d] \in V_\sim(A)\}$.
  Thus $V_\sim(\langle\alpha\rangle A) = \{s | \exists [d] \in M_\sim.((s,[d]) \in R_\sim(\alpha) \wedge [d] \in V_\sim(A)\}$.

– Let $\alpha_0 \in \Pi_0$, let $c_2 \in \Sigma_2$, $c \in \Sigma$:
  - such that $([c_2],[c]) \in R_\sim(\alpha)$ then $\langle \nu_\Sigma \rangle (c_2 \wedge \langle \alpha_0 \rangle c) \in L^*$. But, from $(\Sigma_2 2)$, $\langle \nu_\Sigma \rangle (c_2 \wedge [\alpha_0]\bot) \in L^*$ and thus, from $(\Sigma 2)$, $[\nu_\Sigma](c_2 \Rightarrow [\alpha_0]\bot) \in L^*$. Thus $\langle \nu_\Sigma \rangle ([\alpha_0]\neg c \wedge \langle \alpha_0 \rangle c) \in L^*$ which is impossible as L^* is consistent.
  - such that $([c],[c_2]) \in R_\sim(\alpha)$ then $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha_0 \rangle c) \in L^*$. But, from $(\Sigma_2 2)$, $\langle \nu_\Sigma \rangle (c_2 \wedge [\alpha_0^-]\bot) \in L^*$ and thus, from $(\Sigma 2)$, $[\nu_\Sigma](c_2 \Rightarrow [\alpha_0^-]\bot) \in L^*$. Thus $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha_0 \rangle [\alpha_0^-]\bot) \in L^*$. But then, from $(-)$, $\langle \nu_\Sigma \rangle (c \wedge \bot) \in L^*$ which is impossible as L^* is consistent.

  Thus $\forall c_0 \in \Sigma_2, \forall c \in \Sigma$, $([c_0],[c]) \notin R_\sim(\alpha_0)$ and $([c],[c_0]) \notin R_\sim(\alpha_0)$. Thus $R_\sim(\alpha_0) \in \mathcal{P}(\chi_\sim(\Sigma_1))$.
– Let $S \subseteq \Sigma$:
  - Let $c_0, c_1 \in S$, from $(\Sigma_S 1)$, $\langle \nu_\Sigma \rangle (c_0 \wedge \langle \nu_S \rangle c_1) \in L^*$ and thus $([c_0],[c_1]) \in R_\sim(\nu_S)$. Thus $\chi_\sim(S)^2 \subseteq R_\sim(\nu_S)$.
  - Otherwise, $\{c_0, c_1\} \not\subseteq S$ and then, from $(\Sigma_S 2)$, $\langle \nu_\Sigma \rangle (c_0 \wedge [\nu_S]\neg c_1) \in L^*$. Assume $([c_0],[c_1]) \in R_\sim(\nu_S)$ then $\langle \nu_\Sigma \rangle (c_0 \wedge \langle \nu_S \rangle c_1) \in L^*$ and, from $(\Sigma 2)$, $[\nu_\Sigma](c_0 \Rightarrow \langle \nu_S \rangle c_1) \in L^*$ and thus $\langle \nu_\Sigma \rangle (\langle \nu_S \rangle c_1 \wedge [\nu_S]\neg c_1) \in L^*$ which is impossible as L^* is consistent. Thus $R_\sim(\nu_S) \subseteq \chi_\sim(S)^2$.

  Thus $\chi_\sim(S)^2 = R_\sim(\nu_S)$
–
  - Let $([c],[d]) \in R_\sim(\alpha) \cup R_\sim(\beta)$ then:
    * either $([c],[d]) \in R_\sim(\alpha)$ and thus $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \rangle d) \in L^*$ and then, from $(\cup)$; $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \cup \beta \rangle d) \in L^*$ and thus $([c],[d]) \in R_\sim(\alpha \cup \beta)$,
    * or $([c],[d]) \in R_\sim(\beta)$ and thus $\langle \nu_\Sigma \rangle (c \wedge \langle \beta \rangle d) \in L^*$ and then, from $(\cup)$; $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \cup \beta \rangle d) \in L^*$ and thus $([c],[d]) \in R_\sim(\alpha \cup \beta)$

    Thus $R_\sim(\alpha) \cup R_\sim(\beta) \subseteq R_\sim(\alpha \cup \beta)$.
  - Let $([c],[d]) \in R_\sim(\alpha \cup \beta)$ then $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \cup \beta \rangle d) \in L^*$. Then, from $(\cup)$, $\langle \nu_\Sigma \rangle (c \wedge (\langle \alpha \rangle d \vee \langle \beta \rangle d)) \in L^*$. Then:
    * either $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \rangle d) \in L^*$ and then $([c],[d]) \in R_\sim(\alpha)$,
    * or $\langle \nu_\Sigma \rangle (c \wedge \langle \beta \rangle d) \in L^*$ and then $([c],[d]) \in R_\sim(\beta)$,
    * or $\langle \nu_\Sigma \rangle (c \wedge [\alpha]\neg d \wedge [\beta]\neg d) \in L^*$ and thus, from $(\Sigma 2$, $[\nu_\Sigma](c \Rightarrow ([\alpha]\neg d \wedge [\beta]\neg d)) \in L^*$. Then $\langle \nu_\Sigma \rangle ((\langle \alpha \rangle d \vee \langle \beta \rangle d) \wedge [\alpha]\neg d \wedge [\beta]\neg d) \in L^*$ which is impossible as L^* is consistent.

    Thus $R_\sim(\alpha \cup \beta) \subseteq R_\sim(\alpha) \cup R_\sim(\beta)$

  Thus $R_\sim(\alpha \cup \beta) = R_\sim(\alpha) \cup R_\sim(\beta)$
–
  - Let $([c],[d]) \in \{(s,t) | \exists [e].((s,[e]) \in R_\sim(\alpha)$ and $([e],t) \in R_\sim(\beta))\}$ then $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \rangle e) \in L^*$ and $\langle \nu_\Sigma \rangle (e \wedge \langle \beta \rangle d) \in L^*$ that is, from $(\Sigma 2)$, $[\nu_\Sigma](e \Rightarrow \langle \beta \rangle d) \in L^*$. Then $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \rangle \langle \beta \rangle d) \in L^*$. Then, from $(;)$, $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha ; \beta \rangle d) \in L^*$ and then $([c],[d]) \in R_\sim(\alpha ; \beta)$. Thus $\{(s,t) | \exists [e].((s,[e]) \in R_\sim(\alpha)$ and $([e],t) \in R_\sim(\beta))\} \subseteq R_\sim(\alpha ; \beta)$.
  - Let $([c],[d]) \notin \{(s,t) | \exists [e].((s,[e]) \in R_\sim(\alpha)$ and $([e],t) \in R_\sim(\beta))\}$ that is $\forall e.$ $(([c],[e]) \notin R_\sim(\alpha)$ or $([e],[d]) \notin R_\sim(\beta))$. Assume $\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \rangle e) \in L^*$ and $\langle \nu_\Sigma \rangle (e \wedge \langle \beta \rangle d) \in L^*$ then $([c],[e]) \in R_\sim(\alpha)$ and $([e],[d]) \in R_\sim(\beta)$ which is not the case. Thus $\forall e.$ $(\langle \nu_\Sigma \rangle (c \wedge \langle \alpha \rangle e) \notin L^*$ or $\langle \nu_\Sigma \rangle (e \wedge \langle \beta \rangle d) \notin L^*)$. As L^* is maximal, $\forall e.$ $([\nu_\Sigma](\neg c \vee [\alpha]\neg e) \in L^*$ or $[\nu_\Sigma](\neg e \vee [\beta]\neg d) \in L^*)$. Thus $\forall e.$ $([\nu_\Sigma](c \Rightarrow [\alpha](\langle \beta \rangle d \Rightarrow \neg e) \in L^*$ or, from $(\nu_\Sigma 2)$, $[\nu_\Sigma ; c?; \alpha](\langle \beta \rangle d \Rightarrow \neg e) \in Then^*)$. Thus $\forall e.$ $[\nu_\Sigma ; c?; \alpha ; (\langle \beta \rangle d)?]\neg e) \in L^*$. As L^* is closed under (Cov), $[\nu_\Sigma ; c?; \alpha ; (\langle \beta \rangle d)?]\bot \in L^*$ that is, from

(?) and ($\Sigma 2$), $[\nu_\Sigma](c \Rightarrow [\alpha](\langle\beta\rangle d \Rightarrow \bot)) \in L^*$. Assume $([c],[d]) \in R_\sim(\alpha;\beta)$ then $\langle\nu_\Sigma\rangle(c \wedge \langle\alpha\rangle\langle\beta\rangle d) \in L^*$. Thus $\langle\nu_\Sigma\rangle([\alpha](\langle\beta\rangle d \Rightarrow \bot) \wedge \langle\alpha\rangle\langle\beta\rangle d) \in L^*$ which is impossible as L^* is consistent. Thus $R_\sim(\alpha;\beta) \subseteq \{(s,t)|\exists [e].((s,[e]) \in R_\sim(\alpha)$ and $([e],t) \in R_\sim(\beta))\}$.

Thus $R_\sim(\alpha;\beta) = \{(s,t)|\exists [e].((s,[e]) \in R_\sim(\alpha)$ and $([e],t) \in R_\sim(\beta))\}$.

– • Let $([c],[d]) \in \{(s,t)|(t,s) \in R_\sim(\alpha)\}$ then $\langle\nu_\Sigma\rangle(d \wedge \langle\alpha\rangle c) \in L^*$. From ($\Sigma 1$), $\langle\nu_\Sigma\rangle c \in L^*$. Assume $\langle\nu_\Sigma\rangle(c \wedge [\alpha^-]\neg d) \in L^*$ then, from ($\Sigma 2$), $[\nu_\Sigma](c \Rightarrow [\alpha^-]\neg d) \in L^*$ and thus $\langle\nu_\Sigma\rangle(d \wedge \langle\alpha\rangle[\alpha^-]\neg d) \in L^*$. But, from ($-$), $\langle\nu_\Sigma\rangle(d \wedge \neg d) \in L^*$ which is impossible as L^* is consistent. Thus, as L^* is maximal, $[\nu_\Sigma](\neg c \vee \langle\alpha^-\rangle d) \in L^*$ that is $[\nu_\Sigma](c \Rightarrow \langle\alpha^-\rangle d) \in L^*$ and thus $([c],[d]) \in R_\sim(\alpha^-)$. Thus $\{(s,t)|(t,s) \in R_\sim(\alpha)\} \subseteq R_\sim(\alpha^-)$.

• Let $([c],[d]) \notin \{(s,t)|(t,s) \in R_\sim(\alpha)\}$. If $\langle\nu_\Sigma\rangle(d \wedge \langle\alpha\rangle c) \in L^*$, $([c],[d]) \in \{(s,t)|(t,s) \in R_\sim(\alpha)\}$ which is not the case thus, by maximality of L^*, $[\nu_\Sigma](\neg d \vee [\alpha]\neg c) \in L^*$. Assume $([c],[d]) \in R_\sim(\alpha^-)$, then $\langle\nu_\Sigma\rangle(c \wedge \langle\alpha^-\rangle d) \in L^*$ that is, using ($\Sigma 2$), $[\nu_\Sigma](c \Rightarrow \langle\alpha^-\rangle d) \in L^*$ and thus $[\nu_\Sigma](d \Rightarrow [\alpha]\langle\alpha^-\rangle d) \in L^*$. Then, from ($-$), $[\nu_\Sigma](d \Rightarrow \neg d) \in L^*$ that is, from ($\Sigma 2$), $\langle\nu_\Sigma\rangle(d \wedge \neg d) \in L^*$ which is impossible as L^* is consistent. Thus $R_\sim(\alpha^-) \subseteq \{(s,t)|(t,s) \in R_\sim(\alpha)\}$.

Thus $R_\sim(\alpha^-) = \{(s,t)|(t,s) \in R_\sim(\alpha)\}$.

– • Let $([c],[d]) \in R_\sim(\alpha^*)$. Then, $\langle\nu_\Sigma\rangle(c \wedge \langle\alpha^*\rangle d) \in L^*$. Assume $\forall k.\ \langle\nu_\Sigma\rangle(c \wedge \langle\alpha^k\rangle d) \notin L^*$ then, as L^* is maximal, $\forall k.\ [\nu_\Sigma](\neg c \wedge [\alpha^k]\neg d) \in L^*$ that is $[\nu_\Sigma; c?;][\alpha^k]\neg d \in L^*$. But L^* is closed under (Ind) and thus $[\nu_\Sigma; c?;][\alpha^*]\neg d \in L^*$ that is $[\nu_\Sigma](c \Rightarrow [\alpha^*]\neg d) \in L^*$ and thus $\langle\nu_\Sigma\rangle([\alpha^*]\neg d \wedge \langle\alpha^*\rangle d) \in L^*$ which is impossible as L^* is consistent. Thus $R_\sim(\alpha^*) \subseteq \bigcup_{k\langle\omega} R_\sim(\alpha^k)$

• Let's prove by induction that $\langle\alpha^k\rangle A \Rightarrow \langle\alpha^*\rangle A \in L^*$:
  * From ($*$), $A \Rightarrow \langle\alpha^*\rangle A \in L^*$ thus $\langle\alpha^0\rangle A \Rightarrow \langle\alpha^*\rangle A \in L^*$
  * Assume $\langle\alpha^k\rangle A \Rightarrow \langle\alpha^*\rangle A \in L^*$, then $[\alpha^{k+1}]\neg A \vee \langle\alpha^{k+1}\rangle A \in L^*$ being a tautology, thus $[\alpha^{k+1}]\neg A \vee \langle\alpha\rangle\langle\alpha^k\rangle A \in L^*$ and then, from the induction hypothesis, $[\alpha^{k+1}]\neg A \vee \langle\alpha\rangle\langle\alpha^*\rangle A \in L^*$. But, from ($*$), $\langle\alpha\rangle\langle\alpha^*\rangle A \Rightarrow \langle\alpha^*\rangle A \in L^*$ and thus $\langle\alpha^{k+1}\rangle A \Rightarrow \langle\alpha^*\rangle A \in L^*$.

Then, assume $([c],[d]) \in \bigcup_{k\langle\omega} R_\sim(\alpha^k)$. There exists k such that $([c],[d]) \in R_\sim(\alpha^k)$ and thus $\langle\nu_\Sigma\rangle(c \wedge \langle\alpha^k\rangle d) \in L^*$ but then, as $\langle\alpha^k\rangle A \Rightarrow \langle\alpha^*\rangle A \in L^*$, $\langle\nu_\Sigma\rangle(c \wedge \langle\alpha^*\rangle d) \in L^*$ and thus $([c],[d]) \in R_\sim(\alpha^*)$. Thus $\bigcup_{k\langle\omega} R_\sim(\alpha^k) \subseteq R_\sim(\alpha^*)$.

Thus $R_\sim(\alpha^*) = \bigcup_{k\langle\omega} R_\sim(\alpha^k)$.

– • Let $([c],[d]) \in R_\sim(A?)$ then $\langle\nu_\Sigma\rangle(c \wedge \langle A?\rangle d) \in L^*$. From (?), $\langle\nu_\Sigma\rangle(c \wedge A) \in L^*$ and $\langle\nu_\Sigma\rangle(c \wedge d) \in L^*$. Thus $[c] = [d]$ and $[c] \in V_\sim(A)$ thus $([c],[d]) \in \{(s,s)|s \in V_\sim(A)\}$. Thus $R_\sim(A?) \subseteq \{(s,s)|s \in V_\sim(A)\}$

• Let $([c],[d]) \in \{(s,s)|s \in V_\sim(A)\}$ then $[c] = [d]$ and $[c] \in V_\sim(A)$ thus $\langle\nu_\Sigma\rangle(c \wedge A) \in L^*$ and $\langle\nu_\Sigma\rangle(c \wedge d) \in L^*$ thus $\langle\nu_\Sigma\rangle(c \wedge \langle A?\rangle d) \in L^*$ and thus $([c],[d]) \in R_\sim(A?)$. Thus $\{(s,s)|s \in V_\sim(A)\} \subseteq R_\sim(A?)$.

Thus $R_\sim(A?) \subseteq \{(s,s)|s \in V_\sim(A)\}$.

Thus $\mathcal{M}$ is a model.

Let $A \in L$ then, as L is closed under (Nec), $[\nu_\Sigma]A \in L$ and as L contains ($\Sigma 1$), $\langle\nu_\Sigma\rangle c \in L$ thus $\langle\nu_\Sigma\rangle(c \wedge A) \in L \subseteq L^*$. Thus $\mathcal{M},[c] \vDash A$ which is the case for each $c \in \Sigma$. Thus $\mathcal{M} \vDash A$. Thus $\mathcal{M}$ is a model of L.

We can now prove the theorem itself:

Proof. Assume $\nvdash A$ then $A \notin \mathcal{LDS}$ and also $[\nu_\Sigma]A \notin \mathcal{LDS}$ thus $log(\mathcal{LDS}, \langle\nu_\Sigma\rangle\neg A)$ is consistent and thus, from Lemma 12, has a model $\mathcal{M}$. Thus $\mathcal{M} \vDash \langle\nu_\Sigma\rangle\neg A$ i.e. $\mathcal{M} \nvdash A$ and thus $\nvdash A$.

6.3 Decidability

The idea is similar to the one for combinatory $\mathcal{PDL}$, that is we prove that the ω-rules, (Ind) and (Cov), can be replaced so that the set of valid formulae is recursively enumerable. Then, we prove that if a formula is satisfiable then it is satisfied by a finite model. In this case, there is a procedure, that may not stop, deciding if the formula is valid and another one, that may not stop either, deciding if the formula is invalid. As the formula can't be both, one of them will reach a result eventually.

Definition 22. *Let $\mathcal{FDS}$ be the deductive system obtained from $\mathcal{DS}$ by dropping the rules (Ind) and (Cov) and adding the axiom (ind): $(A \wedge [\alpha^*](A \Rightarrow [\alpha]A)) \Rightarrow [\alpha^*]A$. Let $\vdash_F$ denote provability in $\mathcal{FDS}$. We call $\mathcal{LFDS}$ the set of $\mathcal{C}2\mathcal{PDL}$-formulae $\{A | \vdash_F A\}$.*

$\mathcal{LFDS}$ is a recursively enumerable set.

Lemma 13. *Let A be a $\mathcal{C}2\mathcal{PDL}$-formula, if $\vdash_F A$ then $\vdash A$*

Proof. It amounts to prove that $\vdash ind$. Let $\gamma = (A \wedge [\alpha^*](A \rightarrow [\alpha]A))?$. Assume there exists $k < \omega$ such that $\langle\gamma\rangle\langle\alpha^k\rangle\neg A$ that is $A \wedge [\alpha^*](A \rightarrow [\alpha]A) \wedge \langle\alpha^k\rangle\neg A$. From $(*)$, applied k times, one obtains $A \wedge \bigwedge_{0\leq l<k}[\alpha^l](A \rightarrow [\alpha]A) \wedge [\alpha^k](A \rightarrow [\alpha]A \wedge [\alpha][\alpha^*](A \rightarrow [\alpha]A)) \wedge \langle\alpha^k\rangle\neg A$. Then, $A \wedge \bigwedge_{0\leq l<k}[\alpha^l](A \rightarrow [\alpha]A)$ yields $\bigwedge_{0\leq l\leq k}[\alpha^l]A$ and thus, adding $\langle\alpha^k\rangle\neg A$ an impossibility is reached.

Thus $\forall k < \omega$, $[\gamma][\alpha^k]A$ which, from (Ind), yields $[\gamma][\alpha^*]A$ that is $(A \wedge [\alpha^*](A \rightarrow [\alpha]A)) \rightarrow [\alpha^*]A$. Thus (ind) is a theorem of $\mathcal{DS}$.

This gives us the first of the two semi-decision procedure that we need: $\mathcal{FDS}$ generates only valid formulae and thus if it generates the $\mathcal{C}2\mathcal{PDL}$-formula A it is valid.

Definition 23. *The Fischer-Ladner closure of a set of formulae Ξ is the smallest set $\mathcal{FL}$ that satisfies:*

- $\Xi \subseteq \mathcal{FL}$
- *$\mathcal{FL}$ is closed under sub-formulae*
- *If $[\alpha \cup \beta]A \in \mathcal{FL}$, $[\alpha]A \in \mathcal{FL}$ and $[\beta]A \in \mathcal{FL}$*
- *If $[\alpha;\beta]A \in \mathcal{FL}$, $[\alpha][\beta]A \in \mathcal{FL}$*
- *If $[\alpha^*]A \in \mathcal{FL}$, $[\alpha][\alpha^*]A \in \mathcal{FL}$*
- *If $[\alpha^-]A \in \mathcal{FL}$, $[\alpha]\neg[\alpha^-]A \in \mathcal{FL}$*

Lemma 14. *The Fischer-Ladner closure of a finite set is finite.*

Definition 24. *We name canonical quasi-model the model* $\mathcal{M}_c = (M_c, R_c, V_c)$ *where:*

- M_c *is the set of all maximal consistent sets of formulae*
- *for every program* α *and for all* $u, v \in M_c$, $u\ R_c(\alpha)\ v$ *iff, for every formula* A, *if* $[\alpha]A \in u$ *then* $A \in v$
- *for every atomic proposition* ϕ, $V_c(\phi) = \{u \in M_c | \phi \in u\}$
- *for every name* i, $V_c(i) = \{u \in M_c | i \in u\}$

$\mathcal{M}_c$ is named a quasi-model because it is not a model. It is the template of the model we will create to prove the correctness though.

Lemma 15. *For all* $u \in M_c$ *and all formulae* A, $\mathcal{M}_c, u \vDash A$ *iff* $A \in u$.

Proof. This is done by induction on the complexity of A.

- If A is an atomic proposition or a name, this is true by construction.
- If $A = \neg B$, by induction, $\mathcal{M}_c, u \nvDash B$ iff $B \notin u$. As u is maximal, $B \notin u$ iff $A \in u$ and thus $A \in u$ iff $\mathcal{M}_c, u \vDash A$.
- If $A = B \vee C$, $\mathcal{M}_c, u \vDash A$ iff either $\mathcal{M}_c, u \vdash B$ or $\mathcal{M}_c, u \vDash C$ iff , by induction, either $B \in u$ or $C \in u$ iff, as u is maximal, $B \vee C \in u$.
- If $A = [\alpha]B$,
  - Assume $A \in u$ then for all v such that $u\ R_c(\alpha)\ v$, $B \in v$ by construction. By induction, $\mathcal{M}_c, v \vDash B$ and thus $\mathcal{M}_c, u \vDash A$
  - Assume $\mathcal{M}_c, u \vDash A$, then if v is such that $u\ R_c(\alpha)\ v$, $\mathcal{M}_c, v \vDash B$. By induction, $B \in v$. Assume $A \notin u$, A does not introduce additional constraints and thus $u \cup A$ is consistent. This is impossible as u is maximal. Thus $A \in u$.

Lemma 16. *A is a valid formula of $\mathcal{C}2\mathcal{PDLS}$ iff A is true in $\mathcal{M}_c$.*

Proof. If $\vdash A$, every maximal consistent set contains A and thus, from Lemma 15, $\mathcal{M}_c, u \vdash A$ for all $u \in M_c$. Thus A is true in $\mathcal{M}_c$. Otherwise, $\{\neg A\}$ is consistent and, from Lemma 10, can thus be extended to a maximal consistent set x. As $x \in M_c$ and $A \notin x$, $x \nvDash A$.

Lemma 17. $\forall \alpha, \beta,\ R_c(\alpha \cup \beta) = R_c(\alpha) \cup R_c(\beta)$.

Proof. Assume $uR_c(\alpha \cup \beta)v$ and $[\alpha \cup \beta]A \in u$. Then, as u is maximal $[\alpha]A \in u$ and $[\beta]A \in u$. Thus $R_c(\alpha) \cup R_c(\beta) \subseteq R_c(\alpha \cup \beta)$.

Assume neither $uR_c(\alpha)v$ nor $uR_c(\beta)v$. Then, there exists A and B such that $[\alpha]A \in u$, $A \notin v$, $[\beta]B \in u$ and $B \notin v$. Then $[\alpha](A \vee B) \in u$ and $[\beta](A \vee B) \in u$ and thus, by maximality of u, $[\alpha \cup \beta](A \vee B) \in u$. But $A \vee B \notin v$ thus it is impossible that $uR_c(\alpha \cup \beta)v$ and thus $R_c(\alpha \cup \beta) \subseteq R_c(\alpha) \cup R_c(\beta)$.

Lemma 18. $\forall \alpha, \beta,\ R_c(\alpha; \beta) = \{(x, y) | \exists z.(x, z) \in R_c(\alpha) \wedge (z, y) \in R_c(\beta)\}$.

Proof. Assume $uR_c(\alpha;\beta)v$ and $[\alpha;\beta]A \in u$. Then, as u is maximal $[\alpha][\beta]A \in u$. Thus $\{(x,y)|\exists z.(x,z) \in R_c(\alpha) \wedge (z,y) \in R_c(\beta)\} \subseteq R_c(\alpha;\beta)$.

Assume $uR_c(\alpha;\beta)v$. Let C_i be the formulae in v. We define a new set of formulae such that $B_0 = C_0$ and $B_i = B_{i+1} \wedge C_i$. We consider the set $\Delta = \{A : [\alpha]A \in u\} \cup \{\neg[\beta]\neg B_n : n\langle\omega\}$. Suppose Δ is inconsistent. Then there are $A_0, \ldots, A_n \in \{A : [\alpha]A \in u\}$ and $i_0, \ldots, i_m$ such that $\{A_0, \ldots, A_m, \neg[\beta]\neg B_{i_0}, \ldots, \neg[\beta]\neg B_{i_m}\}$ is an inconsistent set. Let $k = max(i_0, \ldots, i_m)$, then $\{A_0, \ldots, A_m, \neg[\beta]\neg B_k\}$ is inconsistent. Thus $\vDash A_0 \wedge \ldots \wedge A_m \Rightarrow [\beta]\neg B_k$ and thus $\vDash [\alpha]A_0 \wedge \ldots \wedge [\alpha]A_m \Rightarrow [\alpha][\beta]\neg B_k$. Then $[\alpha][\beta]\neg B_k \in u$ and thus $[\alpha;\beta]\neg B_k \in u$ and thus $\neg B_k \in v$. As v is consistent, $B_k \notin v$ which is contrary to the definition of B_k. Thus Δ is consistent. Thus, from Lemma 11, $\exists x$ such that $\Delta \subseteq x$. Then, by definition of $\mathcal{M}_c$, $uR_c(\alpha)x$ and $xR_c(\alpha)y$. Thus $R_c(\alpha;\beta) \subseteq \{(x,y)|\exists z.(x,z) \in R_c(\alpha) \wedge (z,y) \in R_c(\beta)\}$.

Lemma 19. $\forall \alpha, \beta,\ R_c(\alpha^-) = \{(x,y)|(y,x) \in R_c(\alpha)\}$.

Proof. Assume $uR_c(\alpha^-)v$. Pick A such that $[\alpha]A \in v$. It is then impossible that $[\alpha^-]\neg[\alpha]A \in u$ thus $\neg[\alpha^-]\neg[\alpha]A \in u$. Hence $A \in u$. Therefore $vR_c(\alpha)u$ and thus $\{(x,y)|(y,x) \in R_c(\alpha)\} \subseteq R_c(\alpha^-)$.

Assume $vR_c(u)u$. Pick A such that $[\alpha^-]A \in u$. It is then impossible that $\neg[\alpha^-]\neg[\alpha]A \in v$. Hence $A \in v$. Therefore $uR_c(\alpha^-)v$ and thus $R_c(\alpha^-) \subseteq \{(x,y)|(y,x) \in R_c(\alpha)\}$.

Definition 25. *Let $\mathcal{M} = (M, R, \chi, V)$ be a model and let Γ be any set of formulae closed under sub-formulae. We define the equivalence relation $\sim_\Gamma$ on M by:*
$s \sim_\Gamma t$ iff $\forall \phi \in \Gamma$, $(\mathcal{M}, s \vDash \phi$ iff $\mathcal{M}, t \vDash \phi)$.

We note $[s]_\Gamma$ the equivalence class of s with respect to $\sim_\Gamma$. The structure $\mathcal{M}_\Gamma = (M_\Gamma, R_\Gamma, \chi_\Gamma, V_\Gamma)$ is called filtration [22] of $\mathcal{M}_c$ with respect to Γ if:

- $M_\Gamma := \{[s]_\Gamma | s \in M_c\}$
- *for every program $\alpha \in \Gamma$, if $sR_c(\alpha)t$, then $[s]_\Gamma R_\Gamma(\alpha)[t]_\Gamma$*
- *for every program $\alpha \in \Gamma$, if $[s]_\Gamma R_\Gamma(\alpha)[t]_\Gamma$, then all formulae A, $[\alpha]A \in s \cap \Gamma$ only if $A \in t$*
- *for every name in $o \in \Gamma$, if $o \in s$, $[s]_\Gamma \in \chi_\Gamma(o)$*
- *for every atomic proposition $\phi_0 \in \Gamma$, $V_\Gamma(\phi_0) = \{[s]_\Gamma | s \in V_c(\phi_0)\}$*

Let's prove that χ_Γ is a function. Assume $o \in \Gamma$, $[s]_\Gamma$ and $[t]_\Gamma$ such that $[s]_\Gamma \in \chi_\Gamma(o)$ and $[t]_\Gamma \in \chi_\Gamma(o)$. Either $s \sim_\Gamma t$ and then $[s]_\Gamma = [t]_\Gamma$ or there is ϕ such that $\mathcal{M}, s \vDash \phi$ and $\mathcal{M}, t \nvDash \phi$. But then, s being maximal, $\langle\nu_{M_\Gamma}\rangle(o \sqcap \phi) \in s$ and $\langle\nu_{M_\Gamma}\rangle(o \sqcap \neg\phi) \in s$. This is impossible as s is consistent. Thus $\#(\chi_\Gamma(o)) \leq 1$. Moreover, $\langle\nu_{M_\Gamma}\rangle o$ is consistent and thus $\#(\chi_\Gamma(o)) = 1$. All nodes not named with names occurring in Γ can be unnamed and renamed so that χ_Γ is onto.

Lemma 20. *Let $\mathcal{M}_\Gamma$ be the filtration of $\mathcal{M}_c$ with respect to a Γ. Then for each formula $A \in \Gamma$ and all $s \in M_c$, $\mathcal{M}_c, s \vDash A$ iff $\mathcal{M}_\Gamma, [s]_\Gamma \vDash A$.*

Proof. The proof is by induction on A.

- For atomic propositions and names, this is by construction.
- If $A = \phi \wedge \psi$, by the induction hypothesis, $\mathcal{M}_c, s \vDash \phi$ iff $\mathcal{M}_\Gamma, [s]_\Gamma \vDash \phi$ and $\mathcal{M}_c, s \vDash \psi$ iff $\mathcal{M}_\Gamma, [s]_\Gamma \vDash \psi$ and thus $\mathcal{M}_c, s \vDash A$ iff $\mathcal{M}_\Gamma, [s]_\Gamma \vDash A$.
- If $A = \neg\phi$, by the induction hypothesis, $\mathcal{M}, s \vDash \phi$ iff $\mathcal{M}_\Gamma, [s]_\Gamma \vDash \phi$ and thus $\mathcal{M}, s \vDash A$ iff $\mathcal{M}_\Gamma, [s]_\Gamma \vDash A$.
- If $A = [\alpha]B$, then:
  - Assume $\mathcal{M}_c, s \vDash A$ then, from Lemma 15, $A \in s$ and thus, by construction, $B \in t$ for all t such that $sR_c(\alpha)t$. Then, by induction, $\mathcal{M}_\Gamma, [t]_\Gamma \vDash B$. Thus, for all $[t]_\Gamma$ such that $[s]_\Gamma \; R_\Gamma(\alpha)[t]_\Gamma, \mathcal{M}_\Gamma, [t]_\Gamma \vDash B$ thus $\mathcal{M}_\Gamma, [s]_\Gamma \vDash A$
  - Assume $\mathcal{M}_\Gamma, [s] \vDash A$. Then for all $t \in M_c$ with $sR_c(\alpha)t$, $\mathcal{M}_\Gamma, [t] \vDash B$ and thus, by induction, $\mathcal{M}_c, t \vDash B$. Thus $\mathcal{M}_c, s \vDash A$.

Lemma 21. *If Γ is such that $|\Gamma| = n$, that is finite, $|\mathcal{M}_\Gamma| \leq 2^n$.*

Proof. There are at most 2^n equivalence classes for n formulae.

Definition 26. *Let Γ be a set of formulae closed under sub-formulae. Let $\mathcal{M}^\dagger = \{M_\Gamma, R^\dagger, V_\Gamma, \chi_\Gamma\}$ be a model such that, for all atomic programs $\pi \in \Psi$, $[u]_\Gamma R^\dagger [v]_\Gamma$ iff $\exists u_0 \sim_\Gamma u \exists v_0 \sim_\Gamma v.(u_0 R(\pi) v_0)$.*

There can actually be a lot of them as there are no conditions on programs $\pi \notin \Gamma$.

Lemma 22. *$\forall \alpha \in \Gamma$, if $uR_c(\alpha)v$, then $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$.*

Proof. This is done by induction on α.

- If $\alpha \in \Pi_0$, this true by construction.
- If $\alpha = \beta \cup \delta$. Assume $uR_c(\alpha)v$ then, by Lemma 17, $uR_c(\beta)v$ or $uR_c(\delta)v$ so, by induction, $[u]_\Gamma R^\dagger(\beta)[v]_\Gamma$ or $[u]_\Gamma R^\dagger(\delta)[v]_\Gamma$. In either case, $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$.
- If $\alpha = \beta; \delta$. Assume $uR_c(\alpha)v$ then, by Lemma 18, there exists x such that $uR_c(\beta)x$ and $xR_c(\delta)v$ so, by induction, $[u]_\Gamma R^\dagger(\beta)[x]_\Gamma$ and $[x]_\Gamma R^\dagger(\delta)[v]_\Gamma$ that is $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$.
- If $\alpha = \beta^-$. Assume $uR_c(\alpha)v$ then, by Lemma 19, $vR_c(\beta)u$ so, by induction, $[v]_\Gamma R^\dagger(\beta)[u]_\Gamma$ that is $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$.
- If $\alpha = \beta^*$. Assume $uR_c(\alpha)v$ and not $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$. Since $\mathcal{M}^\dagger$ is a model with a finite universe, there exists $B = o_0 \vee \ldots \vee o_n$ such that $\forall w$, $B \in w$ iff $[u]_\Gamma R^\dagger(\alpha)[w]_\Gamma$. In particular, $B \in u$. Moreover, as $B \notin v$, $[\alpha]B \notin u$, then, from (*ind*), $[\beta^*](B \rightarrow [\beta]B) \notin u$. Thus, there exists x, y such that $uR_c(\beta^*)x$, $B \in x$, $xR_c(\beta)y$ and $B \notin y$. Then, $[u]_\Gamma R^\dagger(\beta^*)[x]_\Gamma$ and, by induction, $[x]_\Gamma R^\dagger(\beta)[y]_\Gamma$ and thus $[u]_\Gamma R^\dagger(\beta^*)[y]_\Gamma$. Thus $B \in y$ which is not possible.

Lemma 23. *$\forall \alpha \in \Gamma$, if $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$, then $\forall A \in \Gamma, [\alpha]A \in u \cap \Gamma$ only if $A \in v$.*

Proof. This is done by induction on α.

– Assume $\alpha = \beta \cup \delta$ and $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$. Take any A such that $[\beta \cup \delta]A \in u \cap \Gamma$. As Γ is closed under Fisher-Ladner conditions, $[\beta]A \in u \cap \Gamma$ and $[\delta]A \in u \cap \Gamma$. As $\mathcal{M}^\dagger$ is a model, either $[u]_\Gamma R^\dagger(\beta)[v]_\Gamma$ or $[u]_c R^\dagger(\delta)[v]_\Gamma$. By induction, in either case, $A \in v$.
– Assume $\alpha = \beta; \delta$ and $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$. Take any A such that $[\beta; \delta]A \in u \cap \Gamma$. Then, as Γ is closed under Fischer-Ladner conditions, $[\beta][\delta]A \in \Gamma$. Since $\mathcal{M}^\dagger$ is a model, there exists x such that $[u]_\Gamma R^\dagger(\beta)[x]_\Gamma$ and $[x]_c R^\dagger(\delta)[v]_\Gamma$. By induction, $[\delta]A \in x$ and thus $A \in v$.
– Assume $\alpha = \beta^-$ and $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$. Take any A such that $[\beta^-]A \in u \cap \Gamma$. As Γ is closed under Fischer-Ladner conditions, $[\beta]\neg[\beta^-]A \in \Gamma$. Assume $[\beta]\neg[\beta^-]A \in v \cup \Gamma$. As $\mathcal{M}^\dagger$ is a model, $[v]_\Gamma R^\dagger(\beta)[u]_\Gamma$ and thus, by induction, $\neg[\beta^-]A \in u$ which is impossible. Thus $\langle\beta\rangle[\beta^-]A \in v$ and thus $A \in v$.
– Assume $\alpha = \beta^*$ and $[u]_\Gamma R^\dagger(\alpha)[v]_\Gamma$. Take any A such that $[\beta^*]A \in u \cap \Gamma$. We prove that $\forall x, y, \forall i$, if $[x]_\Gamma (R^\dagger(\beta))^i [y]_\Gamma$ then $[\beta^*]A \in x$ only if $[\beta^*]A \in y$ by induction. The case $i = 0$ is trivial. Suppose the claim holds for n, $[x]_\Gamma (R^\dagger(\beta))^{n+1}[y]_\Gamma$ and $[\beta^*]A \in x$. Then, as Γ is closed under Fischer-Ladner conditions, $[\beta][\beta^*]A \in x$. But then there is z such that $[x]_\Gamma R^\dagger(\beta)[z]_\Gamma$ and $[z]_\Gamma (R^\dagger(\beta))^n [y]_\Gamma$. By induction on the first hypothesis, $[\beta^*]A \in z$ and, by induction on the second hypothesis, $[\beta^*]A \in y$. As $\mathcal{M}^\dagger$ is a model, $[u]_\Gamma R^\dagger(\beta^*)[v]_\Gamma$ implies that there is j such that $[u]_\Gamma (R^\dagger(\beta))^j [v]_\Gamma$ and thus $[\beta^*]A \in v$ and thus $A \in v$.

Lemma 24. *Let Γ be a finite set closed under Fischer-Ladner conditions. Then $\mathcal{M}^\dagger$ is a filtration of $\mathcal{M}_c$ under Γ.*

Proof. – $M_\Gamma := \{[s]_\Gamma | s \in M_c\}$ by construction
– for every program $\alpha \in \Gamma$, if $sR_c(\alpha)t$, then $[s]_\Gamma R_\Gamma(\alpha)[t]_\Gamma$ by Lemma 22.
– for every program $\alpha \in \Gamma$, if $[s]_\Gamma R_\Gamma(\alpha)[t]_\Gamma$, then all formulae A, $[\alpha]A \in s \cap \Gamma$ only if $A \in t$ by Lemma 23
– for every name in $o \in \Gamma$, if $o \in s$, $[s]_\Gamma \in \chi_\Gamma(o)$ by construction
– for every atomic proposition $\phi_0 \in \Gamma$, $V_\Gamma(\phi_0) = \{[s]_\Gamma | s \in V_c(\phi_0)\}$ by construction

Lemma 25. *Let A be a $\mathcal{C}2\mathcal{PDL}$-formula, if $\nvdash_F A$ then, for some finite model $\mathcal{M}$, $\mathcal{M} \not\Vdash A$.*

Proof. Assume $\nvdash_F A$ then, from Lemma 16, there exists x such that $A \notin x$. We call Γ the Fischer-Ladner closure of $\{A\}$. It is finite. We define accordingly $\mathcal{M}^\dagger$. By Lemma 24, $\mathcal{M}^\dagger$ is a filtration and, by Lemma 20, $\mathcal{M}^\dagger, [x]_\Gamma \not\Vdash A$ and $\mathcal{M}^\dagger$ is a model.

6.4 Equivalence of $\mathcal{C}2\mathcal{PDLS}$ and $\mathcal{C}2\mathcal{PDL}$

The following lemmata prove that the rules introduced in Sect. 2 to translate formulae of $\mathcal{C}2\mathcal{PDLS}$ to formulae of $\mathcal{C}2\mathcal{PDL}$ are correct that is that they conserve the valuations.

Lemma 26. *Let $\mathcal{R}$ be one of rules ϕ_1 - ϕ_7, A_r be the right-hand side and A_l be the left-hand side of $\mathcal{R}$. Given any model $\mathcal{M} = (M, R, \chi, V)$, $V(A_l) = V(A_r)$.*

Proof. Rule ϕ_1 : As $V(\top)$ is independent of the definition of V, R, Σ_1 and Σ_2, $V(\top\sigma) = V(\top)$.
Rule ϕ_2 : As nodes are never renamed, $V(o\sigma) = V(o)$.
Rule ϕ_3 : As $\sigma = [add(i_1, j_1, \alpha_0)]$, $\sigma = [del(i_1, j_1, \alpha_0)]$ or $\sigma = [add(i_2)]$ and neither modifies $V(\phi)$, $V(\phi\sigma) = V(\phi)$.
Rule ϕ_4 : As only $V(\phi')$ is modified, $V(\phi[add(i_1, \phi')]) = V(\phi)$.
Rule ϕ_5 : As $V(\phi[add(i_1, \phi)]) = V'(\phi) = \chi(i_1) \cup V(\phi)$, $V(\phi[add(i_1, \phi)]) = V(\phi \vee i_1)$.
Rule ϕ_6 : As only $V(\phi')$ is modified, $V(\phi[del(i_1, \phi')]) = V(\phi)$.
Rule ϕ_7 : As $V(\phi[del(i_1, \phi)]) = V'(\phi) = V(\phi) \cap \overline{\{\chi(i_1)\}}$, $V(\phi[del(i_1, \phi)]) = V(\phi \wedge \neg i_1)$.

Lemma 27. *Let $\mathcal{R}$ be one of rules α_1 - α_8, A_r be the right-hand side and A_l be the left-hand side of $\mathcal{R}$. Given any model $\mathcal{M} = (M, R, \chi, V)$, $R(A_l) = R(A_r)$.*

Proof. Rule α_1 : As $\sigma = [add(i_1, \phi)]$, $\sigma = [del(i_1, \phi)]$ or $\sigma = [add(i_1)]$ and neither modifies $R(\alpha_0)$, $R(\alpha_0[\sigma]) = R(\alpha_0)$.
Rule α_2 : As only $R(\alpha'_0)$ is modified, $R(\alpha_0[add(i_1, j_1, \alpha'_0)]) = R(\alpha_0)$.
Rule α_3 : As $R(\alpha_0[add(i_1, j_1, \alpha_0)]) = R'(\alpha_0) = R(\alpha_0) \cup \{\chi(i_1), \chi(j_1)\}$, $R(\alpha_0[add(i_1, j_1, \alpha_0)]) = R(\alpha_0 \cup (i_1?; \nu_{\Sigma_1}; j_1?)$.
Rule α_4 : As only $R(\alpha'_0)$ is modified, $R(\alpha_0[del(i_1, j_1, \alpha'_0)]) = R(\alpha_0)$.
Rule α_5 : As $R(\alpha_0[del(i_1, j_1, \alpha_0)]) = R'(\alpha_0) = R(\alpha_0) \cap \overline{\{\chi(i_1), \chi(j_1)\}}$, $R(\alpha_0[del(i_1, j_1\alpha_0)]) = R((\neg i_1)?; \alpha_0 \cup \alpha_0; (\neg j_1)?)$.
Rule α_6 As $R(\alpha_0[del(i_1)]) = R'(\alpha_0) = R(\alpha_0) \cap \overline{\{(\chi(i_1), m')\} \cup \{(m', \chi(i_1))\}}$, $R(\alpha_0[del(i_1)]) = R((\neg i_1?); \alpha_0; (\neg i_1)?)$.
Rule α_7 : As $R(\alpha_0[i_1 \gg j_1]) = R'(\alpha_0) = R(pha_0) \cup \{(m', j_1) | (m', i_1) \in R(\alpha_0)\} \cap \overline{\{(m', i_1) \in R(\alpha_0)\}}$, $R(\alpha_0[i_1 \gg j_1]) = R(\alpha_0; ((\neg i_1)? \cup (i_1?; \nu_{\Sigma_1}; j_1?))$.
Rule α_8 : As S is not modified by σ, $R(\nu_S\sigma) = R(\nu_S)$.

We now prove that the rewriting of the sets is correct.

Proof. Rule S_1 : As i_2 is added to Σ_1, $\chi(\Sigma_1[add(i_2)]) = \chi(\Sigma_1) \cup \{i_2\}$
Rule S_2 : As i_2 is deleted from Σ_1, $\chi(\Sigma_1[del(i_1)]) = \chi(\Sigma_1) \cap \overline{\{i_1\}}$
Rule S_3 : As i_1 is deleted from Σ_2, $\chi(\Sigma_2[add(i_2)]) = \chi(\Sigma_2) \cap \overline{\{i_2\}}$
Rule S_4 : As i_1 is added to Σ_2, $\chi(\Sigma_2[del(i_1)]) = \chi(\Sigma_2) \cup \{i_i\}$
Rule S_5 : As only Σ_1 and Σ_2 are modified by σ, $\chi((S_1 \cup S_2)\sigma) = \chi(S_1\sigma) \cup \chi(S_2\sigma)$
Rule S_6 : As only Σ_1 and Σ_2 are modified by σ, $\chi((S_1 \cap S_2)\sigma) = \chi(S_1\sigma) \cap \chi(S_2\sigma)$
Rule S_7 : As only Σ_1 and Σ_2 are modified by σ, $\chi(\overline{S_2}\sigma) = \chi(\overline{S_1\sigma})$
Rule S_8 : As only Σ_1 and Σ_2 are modified by σ, $\chi(\{i\}\sigma) \rightsquigarrow \{i\}$

We now do the same with the other constructors:

Lemma 28. *Let $\mathcal{R}$ be one of rules ϕ_8 - ϕ_{10}, A_r be the righthand side and A_l be the lefthand side of $\mathcal{R}$. Given any model $\mathcal{M} = (M, R, \chi, V)$, $V(A_l) = V(A_r)$.*

Proof. *Rule* ϕ_8 : $V((\neg A)\sigma') = V'(\neg A) = M \cap \overline{\{V'(A)\}} = V(\neg(A\sigma'))$.
Rule ϕ_9 : $V((A \vee B)\sigma') = V'(A \vee B) = V'(A) \cup V'(B) = V((A\sigma') \vee (B\sigma'))$.
Rule ϕ_{10} : $V((\langle\alpha\rangle A)\sigma') = V'(\langle\alpha\rangle A) = \{s|\exists T \in M.((s,t) \in R'(\alpha) \wedge t \in V'(A)\} = V(\langle\alpha\sigma\rangle(A\sigma))$.

Lemma 29. *Let* $\mathcal{R}$ *be one of rules* α_{10} - α_{14}, A_r *be the righthand side and* A_l *be the lefthand side of* $\mathcal{R}$. *Given any model* $\mathcal{M} = (M, R, \chi, V)$, $R(A_l) = R(A_r)$.

Proof. *Rule* α_{10} : $R((\alpha;\beta)\sigma) = R'(\alpha;\beta) = \{(s,t)|\exists v.((s,v) \in R'(\alpha) \wedge (v,t) \in R'(\beta)\} = R((\alpha\sigma);(\beta\sigma))$.
Rule α_{11} : $R((\alpha \cup \beta)\sigma) = R'(\alpha \cup \beta) = R'(\alpha) \cup R'(\beta) = R((\alpha\sigma) \cup (\beta\sigma))$.
Rule α_{12} : $R((\alpha^-)\sigma) = R'(\alpha^-) = \{(s,t)|(t,s) \in R'(\alpha)\} = R((\alpha\sigma)^-)$.
Rule α_{13} : $R((\alpha^*)\sigma) = R'(\alpha^*) = \bigcup_{k\leq\omega} R'(\alpha^k) = R((\alpha\sigma)^*)$.
Rule α_{14} : $R((A?)\sigma) = R'(A?) = \{(s,s)|s \in V'(A)\} = R((A\sigma)?)$.