Theoretical Computer Science 966–967 (2023) 113928

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

On counting propositional logic and Wagner's hierarchy ☆,☆☆

Melissa Antonelli [a,b,*], Ugo Dal Lago [a,c], Paolo Pistone [a]

[a] Department of Computer Science and Engineering, University of Bologna, Mura Anteo Zamboni, 7, 40127, Bologna, Italy
[b] HIIT Helsinki Institute for Information Technology, Pietari Kalmi katu, 5, 00560, Helsinki, Finland
[c] INRIA, Université Côte d'Azur, 2004 Rte des Lucioles, 06902, Valbonne, France

ARTICLE INFO

Article history:
Received 28 February 2022
Accepted 28 April 2023
Available online 24 May 2023

Keywords:
Quantified propositional logic
Counting quantifiers
Counting hierarchy
Computational complexity

ABSTRACT

We introduce an extension of classical propositional logic with counting quantifiers. These forms of quantification make it possible to express that a formula is true *in a certain portion* of the set of all its interpretations. Beyond providing a sound and complete proof system for this logic, we show that validity problems for counting propositional logic can be used to capture counting complexity classes. More precisely, we show that the complexity of the decision problems for validity of prenex counting formulas perfectly matches the appropriate levels of Wagner's counting hierarchy.

1. Introduction

Among the many intriguing relationships existing between logic and computer science, we can certainly mention those connecting classical propositional logic (**PL**, for short), on the one hand, and computational complexity, the theory of programming languages, and several other branches of theoretical computer science, on the other. For instance, it is well-known that **PL** provides the first example of a non-trivial NP-complete problem, as shown by [14], while formal systems for classical and intuitionistic propositional logic correspond to type systems for λ-calculi and related formalisms, see [20,49]. These lines of research have further evolved in various directions, resulting in active sub-areas of computer science. In particular, variations of propositional logic have been put in relation with complexity classes other than P and NP, as well as with type systems other than the simply typed λ-calculus. For example, the complexity of deciding *quantified* propositional formulas is known to match the appropriate level of the polynomial hierarchy (PH, for short), see [36,37,50,57,12].

Of course, not *all* aspects of the theory of computation have found a precise logical counterpart, at least so far. Among the missing ones, several of them are somehow related to probabilistic computation. An example that we find particularly interesting is given by Wagner's hierarchy of *counting complexity classes* (CH for short), see [53,54,56,55]. Indeed, a somehow logical approach to such classes, which are deeply connected to probabilistic complexity classes like PP, has been developed in [29], with tools from descriptive complexity and finite model theory. Yet, to the best of the authors' knowledge, a counterpart of counting classes in the realm of (quantified) propositional logic, in analogy with what happens with the polynomial hierarchy, has not been proposed.

☆ This article belongs to Section A: Algorithms, automata, complexity and games, Edited by Paul Spirakis.

☆☆ Supported by ANR PRC project "PPS - Probabilistic Programming Semantics", ANR-19-CE48-0014. The first author thanks Helsinki Institute for Information Technology HIIT for supporting her work since 2023.

* Corresponding author.
E-mail addresses: melissa.antonelli@helsinki.fi (M. Antonelli), ugo.dallago@unibo.it (U. Dal Lago), paolo.pistone2@unibo.it (P. Pistone).

https://doi.org/10.1016/j.tcs.2023.113928

In this paper we show that such a counterpart exists, in the form of a new quantified propositional logic, that we call *counting* propositional logic. The main feature of this logic is the presence of counting quantifiers, that is, quantifiers designed to *count* the number of values of the bound propositional variables satisfying the argument formula. Such formulas have a natural semantics arising from the observation that the set of valuations of a classical propositional formula forms a *measurable* subset of the Cantor space, $2^{\mathbb{N}}$. Moreover, we will introduce a sound and complete proof theory, in the form of a single-sided sequent calculus.

With the goal of providing the reader with a gentle introduction to the subtle aspects related to this novel form of quantification, we first introduce a simpler logic, called $\mathbf{CPL}_0$, in which, intuitively, quantifiers count models of the *whole* formula, and only later the full logic **CPL**, in which quantifiers can count models of *groups* of variables occurring in a formula. Indeed, both logics are deeply related to counting complexity classes: checking the validity of formulas of $\mathbf{CPL}_0$ is shown to be in the class $\mathsf{P}^{\sharp\mathtt{SAT}}$ of problems having a polytime solution for a machine with an oracle for the counting problem $\sharp\mathtt{SAT}$ (which asks to count the number of models of a given Boolean formula); instead, the problem of checking validity for formulas of **CPL** characterizes the whole counting hierarchy. Actually, in the spirit of the correspondence between quantified propositional logic and the polynomial hierarchy, we prove that deciding (a special kind of) prenex normal forms of **CPL** is complete for the appropriate level of CH.

Structure of the paper This article is a longer version of the eponymous paper [1]. The presentation is structured as follows. First, in Section 2 we introduce the syntax and semantics of $\mathbf{CPL}_0$. Then, in Section 3, we show that the $\mathbf{CPL}_0$-decision problem lies in the complexity class $\mathsf{P}^{\sharp\mathtt{SAT}}$. In Section 4 we introduce a sequent calculus for $\mathbf{CPL}_0$ and establish both its soundness and completeness. In Section 5 we introduce the more expressive logic **CPL** and its semantics. Section 6 is devoted to establishing the link between counting logic and complexity theory, by relating the decision problem for **CPL** with CH. Finally, in Section 7, we define a sound and complete proof system for **CPL**. We conclude the paper by considering some related works in the literature, in Section 8, and by pointing to possible directions of our future study, in Section 9.

2. Univariate counting propositional logic: syntax and semantics

In this section we introduce the syntax and semantics of a univariate version of counting propositional logic, called $\mathbf{CPL}_0$. Although the expressive power of this fragment is limited, it provides a first glimpse of the main ingredients of its close relative, the system **CPL**, introduced in Section 5. Furthermore, validity for formulas of $\mathbf{CPL}_0$ is proved to be a problem in the class $\mathsf{P}^{\sharp\mathtt{SAT}}$.

In the semantics of standard propositional logic, the interpretation of a formula A is a truth-value, obtained by evaluating all propositional variables in A as true or false, according to *some* valuation. The fundamental observation behind our approach to counting quantification is that the set of *all* valuations making A true can itself be taken as a semantics of A, actually a *quantitative* semantics, since this set is *measurable*. Indeed, as propositional formulas may have an arbitrary number of propositional variables, a valuation can be taken as an element of $2^{\mathbb{N}}$. Hence, for any formula A of propositional logic we may take as its interpretation the set $[\![A]\!] \subseteq 2^{\mathbb{N}}$ made of all maps $f \in 2^{\mathbb{N}}$ "making A true". Such sets belong to the standard Borel algebra over $2^{\mathbb{N}}$, $\mathscr{B}(2^{\mathbb{N}})$: atomic propositions are interpreted as *cylinder sets* [10] of the following form:

$$\mathsf{Cyl}(i) = \{f \in 2^{\mathbb{N}} \mid f(i) = 1\} \qquad (i \in \mathbb{N})$$

Non-atomic propositions are interpreted in a natural way by relying on the standard σ-algebra operations of complementation, finite intersection and finite union. Since the sets $[\![A]\!]$ are measurable, and $\mathscr{B}(2^{\mathbb{N}})$ is endowed with a canonical *probability* measure (which gives measure $\frac{1}{2}$ to all cylinders), it makes sense, semantically, to ask whether "A is true with probability *at least* q" or "A is true with probability *strictly less than* q". Defining a logic capable of expressing these conditions is precisely our goal.

In order to express quantitative conditions like the ones mentioned above, we introduce two quantifiers, $\mathbf{C}^q$ and $\mathbf{D}^q$ (inspired from the counting operators from [56,55]), where q ranges over $\mathbb{Q}_{[0,1]}$. Intuitively, the formula $\mathbf{C}^q A$ (resp. $\mathbf{D}^q A$) expresses that A is satisfied by a portion of interpretations greater (resp. strictly smaller) than the set of all the possible ones. For example, the formula $\mathbf{C}^{1/2} A$ expresses the fact that A is satisfied by *at least half* of its valuations. In other words, A is true with probability at least $\frac{1}{2}$. Similarly, the formula $\mathbf{D}^{3/4} A$ expresses that A is satisfied by *strictly less than* three-fourths of its valuations. In other words, the probability for A to be true is strictly smaller than $\frac{3}{4}$.

Definition 1 *(Formula of $\mathbf{CPL}_0$).* The *formulas of $\mathbf{CPL}_0$* are defined by the following grammar:

$$A, B ::= \mathbf{i} \mid \neg A \mid A \wedge B \mid A \vee B \mid \mathbf{C}^q A \mid \mathbf{D}^q A$$

where $\mathbf{i} \in \mathbb{N}$ and $q \in \mathbb{Q}_{[0,1]}$.

Let $\mathscr{C}$ be the set of all cylinder sets, and let $\sigma(\mathscr{C})$ indicate the σ-algebra generated by $\mathscr{C}$, that is the smallest σ-algebra containing $\mathscr{C}$ (which is included in the standard Borel σ-algebra over $2^{\mathbb{N}}$). Moreover, let μ denote the standard cylinder measure over $\sigma(\mathscr{C})$, which can be defined as the *unique* measure on $\sigma(\mathscr{C})$ such that $\mu\big(\mathsf{Cyl}(i)\big) = \frac{1}{2}$, see [10].

Definition 2 *(Semantics of $\mathbf{CPL}_0$).* For each formula A of $\mathbf{CPL}_0$, its *interpretation* is the measurable set $[\![A]\!] \in \mathscr{B}(2^{\mathbb{N}})$ defined in an inductive way as follows:

$$[\![\mathbf{i}]\!] = \mathrm{Cyl}(i) \qquad [\![\mathbf{C}^q A]\!] = \begin{cases} 2^{\mathbb{N}} & \text{if } \mu([\![A]\!]) \geq q \\ \emptyset & \text{otherwise} \end{cases}$$
$$[\![\neg A]\!] = 2^{\mathbb{N}} - [\![A]\!]$$
$$[\![A \wedge B]\!] = [\![A]\!] \cap [\![B]\!] \qquad [\![\mathbf{D}^q A]\!] = \begin{cases} 2^{\mathbb{N}} & \text{if } \mu([\![A]\!]) < q \\ \emptyset & \text{otherwise.} \end{cases}$$
$$[\![A \vee B]\!] = [\![A]\!] \cup [\![B]\!]$$

A formula A is *valid* when $[\![A]\!] = 2^{\mathbb{N}}$. Two formulas A, B are *logically equivalent* (noted $A \equiv B$) when $[\![A]\!] = [\![B]\!]$.

The examples below can help clarifying the semantics of $\mathbf{CPL}_0$.

Example 1. Let us consider the formula $\mathbf{C}^{1/2}A$, where $A = B \vee C$, $B = \mathbf{0} \wedge \neg\mathbf{1}$ and $C = \neg\mathbf{0} \wedge \mathbf{1}$. The measurable sets $[\![B]\!]$ and $[\![C]\!]$ both have measure $\frac{1}{4}$ and are disjoint. Hence, $\mu([\![A]\!]) = \mu([\![B]\!]) + \mu([\![C]\!]) = \frac{1}{2}$, which means that $[\![\mathbf{C}^{1/2}A]\!] = 2^{\mathbb{N}}$, and so the formula $\mathbf{C}^{1/2}A$ is valid.

Example 2. Let us consider the formula $\mathbf{D}^1 A$, where $A = B \vee C$, $B = (\mathbf{0} \wedge \neg\mathbf{1}) \vee \mathbf{2}$ and $C = (\neg\mathbf{0} \wedge \mathbf{1}) \vee \mathbf{2}$. Both sets $[\![B]\!]$ and $[\![C]\!]$ have measure $\frac{5}{8}$ (in fact, 5 of their 8 possible models are satisfying ones). Nevertheless, they are not disjoint, as $[\![B]\!] \cap [\![C]\!] = [\![\mathbf{2}]\!] = \mathrm{Cyl}(2)$, which has measure $\frac{1}{2}$. Hence, $\mu\big([\![A]\!]\big) = \mu\big([\![B]\!]\big) + \mu\big([\![C]\!]\big) - \mu\big(\mathrm{Cyl}(2)\big) = \frac{3}{4}$. So, the formula $\mathbf{D}^1 A$ is valid.

One can easily check that $[\![\mathbf{C}^0 A]\!] = 2^{\mathbb{N}}$ and $[\![\mathbf{D}^0 A]\!] = \emptyset$. Moreover, the two counting quantifiers are inter-definable, as can be easily shown semantically:

$$\mathbf{C}^q A \equiv \neg\mathbf{D}^q A \qquad \mathbf{D}^q A \equiv \neg\mathbf{C}^q A. \tag{1}$$

Observe that they are not dual in the sense of standard modal operators: $\mathbf{C}^q A$ is *not* equivalent to $\neg\mathbf{D}^q\neg A$. Notably, using these quantifiers, it is even possible to express that a formula A is satisfied with probability *strictly greater than q* or *no smaller than q*, as (resp.) $\mathbf{D}^{(1-q)}\neg A$ and $\mathbf{C}^{(1-q)}\neg A$.

3. Univariate counting logic: correspondence with $\mathbf{P}^{\sharp\mathtt{SAT}}$

In order to investigate the complexity of the validity problem for the formulas of $\mathbf{CPL}_0$, let us start by introducing the notion of Boolean formula:

Definition 3 *(Boolean formula). Boolean formulas* are defined by the following grammar:

$$\mathscr{b}, \mathscr{c} ::= x_i \mid \top \mid \bot \mid \neg\mathscr{b} \mid \mathscr{b} \wedge \mathscr{c} \mid \mathscr{b} \vee \mathscr{c},$$

where $i \in \mathbb{N}$. The *interpretation* of a Boolean formula $\mathscr{b}$, $[\![\mathscr{b}]\!] \in \mathscr{B}(2^{\mathbb{N}})$, is inductively defined as follows:

$$[\![x_i]\!] = \mathrm{Cyl}(i) \qquad [\![\neg\mathscr{b}]\!] = 2^{\mathbb{N}} - [\![\mathscr{b}]\!]$$
$$[\![\top]\!] = 2^{\mathbb{N}} \qquad [\![\mathscr{b} \wedge \mathscr{c}]\!] = [\![\mathscr{b}]\!] \cap [\![\mathscr{c}]\!]$$
$$[\![\bot]\!] = \emptyset \qquad [\![\mathscr{b} \vee \mathscr{c}]\!] = [\![\mathscr{b}]\!] \cup [\![\mathscr{c}]\!].$$

In the following, we will consider semantic properties of Boolean formulas of the form $\mu([\![\mathscr{b}]\!]) \triangleright q$, where $\triangleright \in \{\geq, >, \leq, <, =\}$, $\mathscr{b}$ is a Boolean formula and $q \in \mathbb{Q}_{[0,1]}$.

The measure of a Boolean formula, $\mu\big([\![\mathscr{b}]\!]\big)$ can be related to the number $\sharp\mathtt{SAT}(\mathscr{b})$ of the valuations making $\mathscr{b}$ true.

Lemma 1. *For any Boolean formula $\mathscr{b}$, in which exactly n distinct propositional variables occur,* $\mu([\![\mathscr{b}]\!]) = \sharp\mathtt{SAT}(\mathscr{b}) \cdot 2^{-n}$.

Proof. Any valuation $\theta : \{x_0, \ldots, x_{n-1}\} \to 2$ is associated to a measurable set $X(\theta) \in \mathscr{B}(2^{\mathbb{N}})$ by letting $X(\theta) = \{f \mid \forall_{i<n} f(i) = \theta(x_i)\} = \bigcap_{i=0}^{n-1} \mathrm{Cyl}(i)^{\theta(x_i)}$, where $\mathrm{Cyl}(i)^{\theta(x_i)}$ is $\mathrm{Cyl}(i)$ if $\theta(x_i) = 1$ and $\overline{\mathrm{Cyl}(i)}$ otherwise. Observe that $\mu(X(\theta)) = 2^{-n}$. For any $\mathscr{b}$, we have that $[\![\mathscr{b}]\!] = \bigcup_{\theta \vDash \mathscr{b}} X(\theta)$ (this is easily checked by induction on $\mathscr{b}$). Since for all distinct θ, θ', $X(\theta) \cap X(\theta') = \emptyset$, we conclude that $\sharp\mathtt{SAT}(\mathscr{b}) \cdot 2^{-n} = \sum_{\theta \vDash \mathscr{b}} 2^{-n} = \sum_{\theta \vDash \mathscr{b}} \big(\mu(X(\theta))\big) = \mu\big(\bigcup_{\theta \vDash \mathscr{b}} X(\theta)\big) = \mu([\![\mathscr{b}]\!])$. $\square$

$$
\begin{array}{ll}
\texttt{Bool}(\mathbf{i}) = x_i & \texttt{Val}(A \wedge B) = \texttt{Val}(A) \ \texttt{and} \ \texttt{Val}(B) \\
\texttt{Bool}(A \wedge B) = \texttt{Bool}(A) \wedge \texttt{Bool}(B) & \texttt{Val}(A \vee B) = \texttt{Val}(A) \ \texttt{or} \ \texttt{Val}(B) \\
\texttt{Bool}(A \vee B) = \texttt{Bool}(A) \vee \texttt{Bool}(B) & \texttt{Val}(\neg A) = \texttt{not} \ \texttt{Val}(A) \\
\texttt{Bool}(\neg A) = \neg \texttt{Bool}(A) & \texttt{Val}(\mathbf{C}^q A) = \ \texttt{let} \ \mathscr{b} = \texttt{Bool}(A) \ \texttt{in} \\
\texttt{Bool}(\mathbf{C}^q A) = \texttt{Val}(\mathbf{C}^q A) & \qquad \texttt{let} \ n = \sharp\mathsf{vars}(\mathscr{b}) \ \texttt{in} \\
\texttt{Bool}(\mathbf{D}^q A) = \texttt{Val}(\mathbf{D}^q A) & \qquad \dfrac{\sharp\texttt{SAT}(\mathscr{b})}{2^n} \geq q \\
 & \texttt{Val}(\mathbf{D}^q A) = \ \texttt{let} \ \mathscr{b} = \texttt{Bool}(A) \ \texttt{in} \\
 & \qquad \texttt{let} \ n = \sharp\mathsf{vars}(\mathscr{b}) \ \texttt{in} \\
 & \qquad \dfrac{\sharp\texttt{SAT}(\mathscr{b})}{2^n} < q
\end{array}
$$

Fig. 1. Algorithms $\texttt{Bool}(\cdot)$ and $\texttt{Val}(\cdot)$.

We now show that the validity of a formula of $\mathbf{CPL}_0$ can be decided by a polytime algorithm having access to an oracle for the problem $\sharp\texttt{SAT}$ of counting the models of a Boolean formula. Indeed, checking that a quantified formula, such as $\mathbf{C}^q A$ or $\mathbf{D}^q A$, is valid can be done by invoking an oracle that provides measurements of the form $\mu(\llbracket \mathscr{b} \rrbracket)$, for any Boolean formula $\mathscr{b}$. As shown by Lemma 1, these measurements correspond to actually *counting* the number of valuations satisfying the corresponding formula, and can be thus taken as oracles for $\sharp\texttt{SAT}$.

A formula A of $\mathbf{CPL}_0$ is *closed* if it is either of the form $\mathbf{C}^q B$ or $\mathbf{D}^q B$, or it is a negation, conjunction, or disjunction of closed formulas. It can be easily checked by induction on the structure of closed formulas that for any closed A, either $\llbracket A \rrbracket = 2^{\mathbb{N}}$ or $\llbracket A \rrbracket = \emptyset$. We define, by mutual recursion, two polytime algorithms $\texttt{Bool}$ and $\texttt{Val}$: for each formula A of $\mathbf{CPL}_0$, $\texttt{Bool}(A)$ computes a Boolean formula $\mathscr{b}_A$ such that $\llbracket A \rrbracket = \llbracket \mathscr{b}_A \rrbracket$, and, for all closed formula A, $\texttt{Val}(A) = 1$ if and only if $\llbracket A \rrbracket = 2^{\mathbb{N}}$ and $\texttt{Val}(A) = 0$ if and only if $\llbracket A \rrbracket = \emptyset$. Let $\sharp\mathsf{vars}(\mathscr{b})$ denote the number of propositional variables in the Boolean formula $\mathscr{b}$. The two algorithms are defined in Fig. 1. Notice that the algorithm $\texttt{Val}$ makes use of a $\sharp\texttt{SAT}$ oracle. Recall that the class $\mathsf{P}^{\sharp\texttt{SAT}}$ is made of those problems which can be decided in polytime having access to a $\sharp\texttt{SAT}$ oracle, see [4]. One can easily be convinced that the algorithms $\texttt{Bool}$ and $\texttt{Val}$ both belong to $\mathsf{P}^{\sharp\texttt{SAT}}$, which leads to the following:

Proposition 1. *Validity of closed $\mathbf{CPL}_0$-formulas is in $\mathsf{P}^{\sharp\texttt{SAT}}$.*

4. Univariate counting logic: proof theory

We introduce a one-sided, single-succedent sequent calculus, called $\mathbf{CLK}_0$, and prove it sound and complete with respect to the semantics of $\mathbf{CPL}_0$. The language of this calculus is constituted by *labelled* formulas of the form $\mathscr{b} \rightarrowtail A$ or $\mathscr{b} \leftarrowtail A$, where $\mathscr{b}$ and A are respectively a Boolean formula and a formula of $\mathbf{CPL}_0$. Intuitively, $\rightarrowtail$ (resp. $\leftarrowtail$) expresses an inclusion relation (indeed, an implication) between a Boolean formula and a formula of $\mathbf{CPL}_0$: the labelled expression $\mathscr{b} \rightarrowtail A$ (resp., $\mathscr{b} \leftarrowtail A$) says that the set of valuations satisfying $\mathscr{b}$ is included in (resp., includes) the set of valuations satisfying A. In the following, we will use $\mathscr{b} \vDash \mathscr{c}$ for $\llbracket \mathscr{b} \rrbracket \subseteq \llbracket \mathscr{c} \rrbracket$.

Definition 4 *(Labelled formula).* A *labelled formula* is an expression of one of the forms $\mathscr{b} \rightarrowtail A$, $\mathscr{b} \leftarrowtail A$, where $\mathscr{b}$ is Boolean formula and A is a $\mathbf{CPL}_0$-formula. A *labelled sequent* is a sequent of the form $\vdash L$, where L is a labelled formula.

The rules of the calculus $\mathbf{CLK}_0$ include semantic conditions, called *external hypotheses*, which express semantic properties of Boolean formulas or conditions to be checked inside $\mathscr{B}(2^{\mathbb{N}})$.

Definition 5 *(External hypothesis).* An *external hypothesis* is either an expression of the form $\mathscr{b} \vDash \mathscr{c}$ or of the form $\mu(\llbracket \mathscr{b} \rrbracket) \triangleright q$, where $\triangleright \in \{\geq, >, \leq, <, =\}$, $\mathscr{b}, \mathscr{c}$ are Boolean formulas and $q \in \mathbb{Q}_{[0,1]}$.

As we have seen, the measure of the interpretation of a Boolean formula is related to the number $\sharp\texttt{SAT}(\mathscr{b})$ of the valuations making $\mathscr{b}$ true.

The proof system $\mathbf{CLK}_0$ is defined by the rules displayed in Fig. 2. Let us call *μ-rules* the two rules $\mathsf{R}^{\rightarrowtail}_{\mu}$ and $\mathsf{R}^{\leftarrowtail}_{\mu}$. Observe that in the last four rules from Fig. 2 the Boolean formula $\mathscr{b}$ in the conclusion is chosen arbitrarily. This is coherent with the semantics of counting formulas, which are interpreted as either $2^{\mathbb{N}}$ or $\emptyset$, i.e. (resp.) superset or subset of *any* given set. The use of external hypotheses, i.e. of genuinely semantic conditions, as premisses of syntactic rules might seem somehow

Initial Sequents

$$\dfrac{\mathscr{b} \vDash x_i}{\vdash \mathscr{b} \rightarrowtail \mathbf{i}}\ \mathsf{Ax1} \qquad\qquad \dfrac{x_i \vDash \mathscr{b}}{\vdash \mathscr{b} \leftarrowtail \mathbf{i}}\ \mathsf{Ax2}$$

Boolean Rules

$$\dfrac{\vdash \mathscr{c} \rightarrowtail A \qquad \vdash \mathscr{d} \rightarrowtail A \qquad \mathscr{b} \vDash \mathscr{c} \vee \mathscr{d}}{\vdash \mathscr{b} \rightarrowtail A}\ \mathsf{R}^{\rightarrowtail}_{\cup}$$

$$\dfrac{\vdash \mathscr{c} \leftarrowtail A \qquad \vdash \mathscr{d} \leftarrowtail A \qquad \mathscr{c} \wedge \mathscr{d} \vDash \mathscr{b}}{\vdash \mathscr{b} \leftarrowtail A}\ \mathsf{R}^{\leftarrowtail}_{\cap}$$

Logical Rules

$$\dfrac{\vdash \mathscr{c} \leftarrowtail A \qquad \mathscr{b} \vDash \neg \mathscr{c}}{\vdash \mathscr{b} \rightarrowtail \neg A}\ \mathsf{R}^{\rightarrowtail}_{\neg} \qquad\qquad \dfrac{\vdash \mathscr{c} \rightarrowtail A \qquad \neg \mathscr{c} \vDash \mathscr{b}}{\vdash \mathscr{b} \leftarrowtail \neg A}\ \mathsf{R}^{\leftarrowtail}_{\neg}$$

$$\dfrac{\vdash \mathscr{b} \rightarrowtail A}{\vdash \mathscr{b} \rightarrowtail A \vee B}\ \mathsf{R1}^{\rightarrowtail}_{\vee} \qquad\qquad \dfrac{\vdash \mathscr{b} \rightarrowtail B}{\vdash \mathscr{b} \rightarrowtail A \vee B}\ \mathsf{R2}^{\rightarrowtail}_{\vee}$$

$$\dfrac{\vdash \mathscr{b} \leftarrowtail A \qquad \vdash \mathscr{b} \leftarrowtail B}{\vdash \mathscr{b} \leftarrowtail A \vee B}\ \mathsf{R}^{\leftarrowtail}_{\vee} \qquad\qquad \dfrac{\vdash \mathscr{b} \rightarrowtail A \qquad \vdash \mathscr{b} \rightarrowtail B}{\vdash \mathscr{b} \rightarrowtail A \wedge B}\ \mathsf{R}^{\rightarrowtail}_{\wedge}$$

$$\dfrac{\vdash \mathscr{b} \leftarrowtail A}{\vdash \mathscr{b} \leftarrowtail A \wedge B}\ \mathsf{R}^{\leftarrowtail}_{\wedge} \qquad\qquad \dfrac{\vdash \mathscr{b} \leftarrowtail B}{\vdash \mathscr{b} \leftarrowtail A \wedge B}\ \mathsf{R}^{\leftarrowtail}_{\wedge}$$

Counting Rules

$$\dfrac{\mu(\llbracket \mathscr{b} \rrbracket) = 0}{\vdash \mathscr{b} \rightarrowtail A}\ \mathsf{R}^{\rightarrowtail}_{\mu} \qquad\qquad \dfrac{\mu(\llbracket \mathscr{b} \rrbracket) = 1}{\vdash \mathscr{b} \leftarrowtail A}\ \mathsf{R}^{\leftarrowtail}_{\mu}$$

$$\dfrac{\vdash \mathscr{c} \rightarrowtail A \qquad \mu(\llbracket \mathscr{c} \rrbracket) \geq q}{\vdash \mathscr{b} \rightarrowtail \mathbf{C}^q A}\ \mathsf{R}^{\rightarrowtail}_{\mathbf{C}} \qquad\qquad \dfrac{\vdash \mathscr{c} \leftarrowtail A \qquad \mu(\llbracket \mathscr{c} \rrbracket) < q}{\vdash \mathscr{b} \leftarrowtail \mathbf{C}^q A}\ \mathsf{R}^{\leftarrowtail}_{\mathbf{C}}$$

$$\dfrac{\vdash \mathscr{c} \leftarrowtail A \qquad \mu(\llbracket \mathscr{c} \rrbracket) < q}{\vdash \mathscr{b} \rightarrowtail \mathbf{D}^q A}\ \mathsf{R}^{\rightarrowtail}_{\mathbf{D}} \qquad\qquad \dfrac{\vdash \mathscr{c} \rightarrowtail A \qquad \mu(\llbracket \mathscr{c} \rrbracket) \geq q}{\vdash \mathscr{b} \leftarrowtail \mathbf{D}^q A}\ \mathsf{R}^{\leftarrowtail}_{\mathbf{D}}$$

Fig. 2. Proof System for $\mathbf{CPL}_0$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x_0 \vDash x_0}{\vdash x_0 \rightarrowtail \mathbf{0}}\ \mathsf{Ax1}}{\vdash x_0 \wedge \neg x_1 \rightarrowtail \mathbf{0}}\ \mathsf{R}^{\rightarrowtail}_{\cup} \qquad \dfrac{\dfrac{\dfrac{x_1 \vDash x_1}{\vdash x_1 \leftarrowtail \mathbf{1}}\ \mathsf{Ax2}}{\vdash \neg x_1 \rightarrowtail \neg \mathbf{1}}\ \mathsf{R}^{\rightarrowtail}_{\neg}}{\vdash x_0 \wedge \neg x_1 \rightarrowtail \neg \mathbf{1}}\ \mathsf{R}^{\rightarrowtail}_{\cup}}{\vdash x_0 \wedge \neg x_1 \rightarrowtail \mathbf{0} \wedge \neg \mathbf{1}}\ \mathsf{R}^{\rightarrowtail}_{\wedge}}{\vdash x_0 \wedge \neg x_1 \rightarrowtail (\mathbf{0} \wedge \neg \mathbf{1}) \vee (\neg \mathbf{0} \wedge \mathbf{1})}\ \mathsf{R1}^{\rightarrowtail}_{\vee} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x_0 \vDash x_0}{\vdash x_0 \leftarrowtail \mathbf{0}}\ \mathsf{Ax2}}{\vdash \neg x_0 \rightarrowtail \neg \mathbf{0}}\ \mathsf{R}^{\rightarrowtail}_{\neg}}{\vdash \neg x_0 \wedge x_1 \rightarrowtail \neg \mathbf{0}}\ \mathsf{R}^{\rightarrowtail}_{\cup} \qquad \dfrac{\dfrac{x_1 \vDash x_1}{\vdash x_1 \rightarrowtail \mathbf{1}}\ \mathsf{Ax1}}{\vdash \neg x_0 \wedge x_1 \rightarrowtail \mathbf{1}}\ \mathsf{R}^{\rightarrowtail}_{\cup}}{\vdash \neg x_0 \wedge x_1 \rightarrowtail \neg \mathbf{0} \wedge \mathbf{1}}\ \mathsf{R}^{\rightarrowtail}_{\wedge}}{\vdash \neg x_0 \wedge x_1 \rightarrowtail (\mathbf{0} \wedge \neg \mathbf{1}) \vee (\neg \mathbf{0} \wedge \mathbf{1})}\ \mathsf{R2}^{\rightarrowtail}_{\vee}}{\dfrac{\vdash (x_0 \wedge \neg x_1) \vee (\neg x_0 \wedge x_1) \rightarrowtail (\mathbf{0} \wedge \neg \mathbf{1}) \vee (\neg \mathbf{0} \wedge \mathbf{1})}{\vdash \top \rightarrowtail \mathbf{C}^{1/2}\big((\mathbf{0} \wedge \neg \mathbf{1}) \vee (\neg \mathbf{0} \wedge \mathbf{1})\big)}\ \mathsf{R}^{\rightarrowtail *}_{\mathbf{C}}}\ \mathsf{R}^{\rightarrowtail}_{\cup}$$

*as $\mu\big(\llbracket (x_0 \wedge \neg x_1) \vee (\neg x_0 \wedge x_1) \rrbracket\big) \geq \frac{1}{2}$

Fig. 3. Derivation of $\vdash \top \rightarrowtail \mathbf{C}^{1/2}\big((\mathbf{0} \wedge \neg \mathbf{1}) \vee (\neg \mathbf{0} \wedge \mathbf{1})\big)$ in $\mathbf{CLK}_0$.

unsatisfactory. Such premisses correspond to the idea that, when searching for a proof of a counting formula, one might need to query an *oracle* for values of the form $\mu(\llbracket \mathscr{b} \rrbracket)$ (in fact, by Lemma 1, an oracle for $\sharp\mathtt{SAT}(\mathscr{b})$). Derivations in $\mathbf{CLK}_0$ are defined in the standard way. Let $\vdash_{\mathbf{CLK}_0} L$ indicate that $\vdash L$ is derivable by the rules in Fig. 2. The *height* of a derivation in $\mathbf{CLK}_0$ is defined, as usual, as the greatest number of successive applications of rules in it, where initial sequents and μ-rules have height 0. In Fig. 3 we provide an example of derivation in $\mathbf{CLK}_0$.

4.1. Soundness and completeness

We will show that $\mathbf{CLK}_0$ is sound and complete with respect to the semantics of $\mathbf{CPL}_0$: a labelled formula is valid if and only if it is provable. Validity of labelled formulas and sequents is defined as follows:

Definition 6 *(Validity).* A labelled formula $\mathscr{b} \rightarrowtail A$ (resp. $\mathscr{b} \leftarrowtail A$) is *valid*, noted $\vDash \mathscr{b} \rightarrowtail A$ (resp. $\vDash \mathscr{b} \leftarrowtail A$), when $[\![\mathscr{b}]\!] \subseteq [\![A]\!]$ (resp. $[\![A]\!] \subseteq [\![\mathscr{b}]\!]$). A sequent $\vdash L$ is *valid*, denoted $\vDash L$, when L is a valid labelled formula.

Recall that validity of $\mathbf{CPL}_0$-formulae can be checked in polynomial time with an oracle on $\sharp\mathtt{SAT}$. The following useful result shows that, with the same complexity, any formula of $\mathbf{CPL}_0$ can be transformed into an equivalent Boolean formula.

Proposition 2. *For any formula A of $\mathbf{CPL}_0$ there exists a Boolean formula $\mathscr{b}_A$ such that $A \equiv \mathscr{b}_A$. Moreover, the computation of $\mathscr{b}_A$ from A can be done in polynomial time with an access to an oracle for $\sharp\mathtt{SAT}$.*

Proof. We construct the Boolean formula $\mathscr{b}_A$ by induction on A. The only non-trivial cases are $A = \mathbf{C}^q B$ and $A = \mathbf{D}^q B$. In the first case we let $\mathscr{b}_A = \top$ if $\mu([\![B]\!]) \geq q$ and $\mathscr{b}_A = \bot$ if $\mu([\![B]\!]) < q$. Similarly, in the second case, we let $\mathscr{b}_A = \top$ if $\mu([\![B]\!]) < q$ and $\mathscr{b}_A = \bot$ if $\mu([\![B]\!]) \geq q$. □

The soundness of $\mathbf{CLK}_0$ is proved by induction on the height of derivations.

Proposition 3 *(Soundness).* $\vdash_{\mathbf{CLK}_0} L$ *implies* $\vDash L$.

Proof. The proof is by induction on the height of the derivation $\vdash L$. We only consider a few relevant cases:

- $\mathsf{R}_{\cup}^{\rightarrowtail}$. Assume that the last rule applied is an instance of $\mathsf{R}_{\cup}^{\rightarrowtail}$ and that the derivation is in the following form:

$$\dfrac{\begin{matrix}\vdots\\ \vdash \mathscr{c} \rightarrowtail A\end{matrix} \qquad \begin{matrix}\vdots\\ \vdash \mathscr{d} \rightarrowtail A\end{matrix} \qquad \mathscr{b} \vDash \mathscr{c} \vee \mathscr{d}}{\vdash \mathscr{b} \rightarrowtail A}\ \mathsf{R}_{\cup}^{\rightarrowtail}$$

  By IH, $\vDash \mathscr{c} \rightarrowtail A$ and $\vDash \mathscr{d} \rightarrowtail A$, that is $[\![\mathscr{c}]\!] \subseteq [\![A]\!]$ and $[\![\mathscr{d}]\!] \subseteq [\![A]\!]$. Thus also $[\![\mathscr{c}]\!] \cup [\![\mathscr{d}]\!] \subseteq [\![A]\!]$ holds. Given the external hypothesis $\mathscr{b} \vDash \mathscr{c} \vee \mathscr{d}$, that is $[\![\mathscr{b}]\!] \subseteq [\![\mathscr{c}]\!] \cup [\![\mathscr{d}]\!]$. Then, $[\![\mathscr{b}]\!] \subseteq [\![A]\!]$ and so $\vDash \mathscr{b} \rightarrowtail A$.
- $\mathsf{R}_{\mathbf{C}}^{\rightarrowtail}$. Assume that the last rule applied is an instance of $\mathsf{R}_{\mathbf{C}}^{\rightarrowtail}$ and the derivation is in the following form:

$$\dfrac{\begin{matrix}\vdots\\ \vdash \mathscr{c} \rightarrowtail A\end{matrix} \qquad \mu([\![\mathscr{c}]\!]) \geq q}{\vdash \mathscr{b} \rightarrowtail \mathbf{C}^q A}\ \mathsf{R}_{\mathbf{C}}^{\rightarrowtail}$$

  By IH, $\vDash \mathscr{c} \rightarrowtail A$, that is $[\![\mathscr{c}]\!] \subseteq [\![A]\!]$. Since, given the external hypothesis, $\mu([\![\mathscr{c}]\!]) \geq q$, also $\mu([\![A]\!]) \geq q$ holds. Thus, $[\![\mathbf{C}^q A]\!] = 2^{\mathbb{N}}$ and for each $\mathscr{b}$, $[\![\mathscr{b}]\!] \subseteq [\![\mathbf{C}^q A]\!]$. Therefore, $\vDash \mathscr{b} \rightarrowtail \mathbf{C}^q A$.
- $\mathsf{R}_{\mathbf{C}}^{\leftarrowtail}$. Assume that the last rule applied is an instance of $\mathsf{R}_{\mathbf{C}}^{\leftarrowtail}$ and the derivation is in the following form:

$$\dfrac{\begin{matrix}\vdots\\ \vdash \mathscr{c} \leftarrowtail A\end{matrix} \qquad \mu([\![\mathscr{c}]\!]) < q}{\vdash \mathscr{b} \leftarrowtail \mathbf{C}^q A}\ \mathsf{R}_{\mathbf{C}}^{\leftarrowtail}$$

  By IH, $\vDash \mathscr{c} \leftarrowtail A$, that is $[\![A]\!] \subseteq [\![\mathscr{c}]\!]$. Since by hypothesis $\mu([\![\mathscr{c}]\!]) < q$, also $\mu([\![A]\!]) < q$ holds. Thus, $[\![\mathbf{C}^q A]\!] = \emptyset$ and so for every $\mathscr{b}$, $[\![\mathbf{C}^q A]\!] \subseteq [\![\mathscr{b}]\!]$. Therefore, $\vDash \mathscr{b} \leftarrowtail \mathbf{C}^q A$.
- $\mathsf{R}_{\mathbf{D}}^{\rightarrowtail}, \mathsf{R}_{\mathbf{D}}^{\leftarrowtail}$. These cases are treated as $\mathsf{R}_{\mathbf{C}}^{\leftarrowtail}, \mathsf{R}_{\mathbf{C}}^{\rightarrowtail}$. □

The proof of completeness is less straightforward. First, we introduce a *decomposition relation* $\rightsquigarrow$ between finite sets of sequents (that we will indicate as $\Phi, \Psi, \dots$), which allows one to decompose the validity of a sequent into that of a finite set of less complex sequents. The fundamental ingredients of the completeness proof are then expressed by the following three properties:

1. if Φ is $\rightsquigarrow$-normal, then $\vDash \Phi$ if and only if $\vdash_{\mathbf{CLK}_0} \Phi$;
2. if $\vDash \Phi$, then there is a $\rightsquigarrow$-normal Ψ such that $\Phi \rightsquigarrow \Psi$ and $\vDash \Psi$;
3. if $\vdash_{\mathbf{CLK}_0} \Psi$ and $\Phi \rightsquigarrow \Psi$, then $\vdash_{\mathbf{CLK}_0}$.

Using these, one can check that if $\vDash L$ holds, then by (2) $\vDash \Phi$ holds for some $\rightsquigarrow$-normal form Φ of $\vdash L$. Then, by (1), Φ is derivable in $\mathbf{CLK}_0$ and by (3) we conclude that $\vdash L$ is derivable in $\mathbf{CLK}_0$ as well. In order to establish properties 1.-3. above, some preliminary notions and lemmas are needed.

Definition 7 *(Basic formula and sequent).* A *basic formula* of $\mathbf{CLK}_0$ is a labelled formula, $\mathscr{b} \rightarrowtail A$ or $\mathscr{b} \leftarrowtail A$, in which the logical part A is atomic. A *basic sequent* of $\mathbf{CLK}_0$ is a sequent of the form $\vdash L$, where L is a basic formula.

The decomposition reduction $\rightsquigarrow$ is defined starting from the following relation:

Definition 8 *(Decomposition reduction, $\rightsquigarrow_0$).* The *decomposition reduction*, $\rightsquigarrow_0$, from a sequent to a set of sequents (both in the language of $\mathbf{CLK}_0$), is defined by the following decomposition rules:

$$\begin{aligned}
\text{if } \mathscr{b} \vDash \neg \mathscr{c},\ \vdash \mathscr{b} \rightarrowtail \neg A &\rightsquigarrow_0 \{\vdash \mathscr{c} \leftarrowtail A\}\\
\text{if } \neg \mathscr{c} \vDash \mathscr{b},\ \vdash \mathscr{b} \leftarrowtail \neg A &\rightsquigarrow_0 \{\vdash \mathscr{c} \rightarrowtail A\}\\
\text{if } \mathscr{b} \vDash \mathscr{c} \vee \mathscr{d},\ \vdash \mathscr{b} \rightarrowtail A \vee B &\rightsquigarrow_0 \{\vdash \mathscr{c} \rightarrowtail A, \vdash \mathscr{d} \rightarrowtail B\}\\
\vdash \mathscr{b} \leftarrowtail A \vee B &\rightsquigarrow_0 \{\vdash \mathscr{b} \leftarrowtail A, \vdash \mathscr{b} \leftarrowtail B\}\\
\vdash \mathscr{b} \rightarrowtail A \wedge B &\rightsquigarrow_0 \{\vdash \mathscr{b} \rightarrowtail A, \vdash \mathscr{b} \rightarrowtail B\}\\
\text{if } \mathscr{c} \wedge \mathscr{d} \vDash \mathscr{b},\ \vdash \mathscr{b} \leftarrowtail A \wedge B &\rightsquigarrow_0 \{\vdash \mathscr{c} \leftarrowtail A, \vdash \mathscr{d} \leftarrowtail B\}\\
\text{if } \mu(\llbracket \mathscr{c} \rrbracket) \geq q,\ \vdash \mathscr{b} \rightarrowtail \mathbf{C}^q A &\rightsquigarrow_0 \{\vdash \mathscr{c} \rightarrowtail A\}\\
\text{if } \mu(\llbracket \mathscr{c} \rrbracket) < q,\ \vdash \mathscr{b} \leftarrowtail \mathbf{C}^q A &\rightsquigarrow_0 \{\vdash \mathscr{c} \leftarrowtail A\}\\
\text{if } \mu(\llbracket \mathscr{c} \rrbracket) < q,\ \vdash \mathscr{b} \rightarrowtail \mathbf{D}^q A &\rightsquigarrow_0 \{\vdash \mathscr{c} \leftarrowtail A\}\\
\text{if } \mu(\llbracket \mathscr{c} \rrbracket) \geq q,\ \vdash \mathscr{b} \leftarrowtail \mathbf{D}^q A &\rightsquigarrow_0 \{\vdash \mathscr{c} \rightarrowtail A\}\\
\text{if } \mu(\llbracket \mathscr{b} \rrbracket) = 0,\ \vdash \mathscr{b} \rightarrowtail A &\rightsquigarrow_0 \{\}\\
\text{if } \mu(\llbracket \mathscr{b} \rrbracket) = 1,\ \vdash \mathscr{b} \leftarrowtail A &\rightsquigarrow_0 \{\}\\
\text{if } \mu(\llbracket \mathscr{b} \rrbracket) \neq 0, \vdash \mathscr{b} \rightarrowtail \mathbf{D}^0 A &\rightsquigarrow_0 \{\vdash \bot\}\\
\text{if } \mu(\llbracket \mathscr{b} \rrbracket) \neq 1,\ \vdash \mathscr{b} \leftarrowtail \mathbf{C}^0 A &\rightsquigarrow_0 \{\vdash \bot\}.
\end{aligned}$$

Observe that the rewriting rules are defined so that the application of a decomposition rule to an arbitrary sequent, $\vdash L$, leads to a set of sequents $\{\vdash L_1, \dots, \vdash L_n\}$, such that for every $i \in \{1, \dots, n\}$, the number of connectives occurring in the $\mathbf{CPL}_0$-formula of $\vdash L_i$ is (strictly) smaller than that of the $\mathbf{CPL}_0$-formula of $\vdash L$.

The set-decomposition reduction, $\rightsquigarrow$, relating sets of sequents, is defined starting from $\rightsquigarrow_0$ as follows:

Definition 9 *(Set decomposition, $\rightsquigarrow$).* The *set-decomposition reduction*, $\rightsquigarrow$, from a set of sequents to another set of sequents in $\mathbf{CLK}_0$ is defined as follows:

$$\frac{\vdash L_i \rightsquigarrow \{\vdash L_{i_1}, \dots, \vdash L_{i_m}\}}{\{\vdash L_1, \dots, \vdash L_i, \dots, \vdash L_n\} \rightsquigarrow \{\vdash L_1, \dots, \vdash L_{i_1}, \dots, \vdash L_{i_m}, \dots, \vdash L_n\}}$$

Otherwise said, $\rightsquigarrow$ is the natural lifting of $\rightsquigarrow_0$ to a relation on sets.

Each predicate concerning one sequent can be naturally generalized to sets of sequents by stipulating that a predicate holds for the set when it holds for every sequent in the set. In order to make the presentation clearer, given a sequent $\vdash L$, we call its *corresponding set* the set including only this sequent as its element, i.e. $\{\vdash L\}$. The notion of $\rightsquigarrow$-normal form is defined in a standard way:

Definition 10 *($\rightsquigarrow$-Normal form).* A sequent is a *$\rightsquigarrow$-normal form* if no decomposition rewriting reduction rule, $\rightsquigarrow_0$, can be applied on it. A set of sequents is in *$\rightsquigarrow$-normal form* if it cannot be reduced by any $\rightsquigarrow$ set-rewriting rule.

The first step consists in showing that $\rightsquigarrow$ is strongly normalizing, and so that each decomposition process terminates. A couple of auxiliary definitions are needed.

Definition 11 *(Number of connectives,* cn*).* For any labelled formula L, we define its *number of connectives* $\mathsf{cn}(L)$ as follows:

$$\mathsf{cn}(\mathscr{b} \rightarrowtail \mathbf{i}) = \mathsf{cn}(\mathscr{b} \leftarrowtail \mathbf{i}) = \mathsf{cn}(\mathbf{i}) = 1$$

$$\mathsf{cn}(\mathscr{b} \rightarrowtail \neg A) = \mathsf{cn}(\mathscr{b} \leftarrowtail \neg A) = \mathsf{cn}(\neg A) = 1 + \mathsf{cn}(A)$$

$$\mathsf{cn}(\mathscr{b} \rightarrowtail A \diamond B) = \mathsf{cn}(\mathscr{b} \leftarrowtail A \diamond B) = \mathsf{cn}(A \diamond B) = 1 + \mathsf{cn}(A) + \mathsf{cn}(B)$$

$$\mathsf{cn}(\mathscr{b} \rightarrowtail \square A) = \mathsf{cn}(\mathscr{b} \leftarrowtail \square A) = \mathsf{cn}(\square A) = 1 + \mathsf{cn}(A)$$

where $\diamond \in \{\wedge, \vee\}$ and $\square \in \{\mathbf{C}^q, \mathbf{D}^q\}$. Notice that $\mathsf{cn}(L)$ only counts the connectives in the $\mathbf{CPL}_0$-formula of L (for this reason, we sometimes note it also as $\mathsf{cn}(A)$. Given a sequent $\vdash L$, we define its number of connectives as $\mathsf{cn}(\vdash L) = \mathsf{cn}(L)$.

Definition 12 *(Set measure,* ms*).* Given a set of sequents $\Phi = \{\vdash L_1, \ldots, \vdash L_n\}$, its *measure* is defined as $\mathsf{ms}(\Phi) = \sum_{i=1}^{n} \mathsf{cn}(\vdash L_i)$.

Lemma 2. *The reduction $\rightsquigarrow$ is strongly normalizing*

Proof. That every set of sequents is $\rightsquigarrow$-strongly normalizing is proved by showing that, if $\Phi \rightsquigarrow_0 \Psi$, then $\mathsf{ms}(\Phi) > \mathsf{ms}(\Psi)$. The proof is based on exhaustive inspection of all possible forms of $\rightsquigarrow$-reduction, applicable to the given set, that is by dealing with all possible forms of $\rightsquigarrow_0$-reduction of one of the $\vdash L_i$, where $i \in \{1, \ldots, m\}$. Thus, we will take an arbitrary $\vdash L_i$ to be the "active" sequent of $\rightsquigarrow$. Let us consider all the possible forms of $\rightsquigarrow_0$ on which $\rightsquigarrow$ can be based:

- $L_i = \mathscr{b} \rightarrowtail \neg A$. Assume that $\vdash L_i$ is the active sequent in the given $\rightsquigarrow$-decomposition and that $\rightsquigarrow$ is based on the $\rightsquigarrow_0$ below:

$$\vdash \mathscr{b} \rightarrowtail \neg A \quad \rightsquigarrow_0 \quad \{\vdash \mathscr{c} \leftarrowtail A\}$$

where $\mathscr{b} \vDash \neg \mathscr{c}$. Thus,

$$\begin{aligned}
&\mathsf{ms}\big(\{\vdash L_1, \ldots \vdash \mathscr{c} \leftarrowtail A, \ldots \vdash L_m\}\big) \\
&\qquad = \mathsf{cn}(L_1) + \ldots + \mathsf{cn}(\mathscr{c} \leftarrowtail A) + \ldots + \mathsf{cn}(L_m) \\
&\qquad = \mathsf{cn}(L_1) + \ldots + \mathsf{cn}(L) + \ldots + \mathsf{cn}(L_m) \\
&\qquad = \mathsf{cn}(L_1) + \ldots + \mathsf{cn}(\neg A) - 1 + \ldots + \mathsf{cn}(L_m) \\
&\qquad < \mathsf{cn}(L_1) + \ldots + \mathsf{cn}(\mathscr{b} \rightarrowtail \neg A) + \ldots + \mathsf{cn}(L_m) \\
&\qquad = \mathsf{ms}\big(\{\vdash L_1, \ldots \vdash \mathscr{b} \rightarrowtail \neg A, \ldots \vdash L_m\}\big).
\end{aligned}$$

- $L_i = \mathscr{b} \leftarrowtail \neg A$. Equivalent to the case above.
- $L_i = \mathscr{b} \rightarrowtail A \vee B$. Assume that $\vdash L_i$ is the active sequent of $\rightsquigarrow$ and that $\rightsquigarrow$ is based on the following $\rightsquigarrow_0$:

$$\vdash \mathscr{b} \rightarrowtail A \vee B \quad \rightsquigarrow_0 \quad \{\vdash \mathscr{c} \rightarrowtail A, \vdash \mathscr{d} \rightarrowtail B\}$$

where $\mathscr{b} \vDash \mathscr{c} \vee \mathscr{d}$. Thus,

$$\begin{aligned}
&\mathsf{ms}\big(\{\vdash L_1, \ldots \vdash \mathscr{c} \rightarrowtail A, \vdash \mathscr{d} \rightarrowtail B, \ldots \vdash L_m\}\big) \\
&\qquad = \mathsf{cn}(L_1) + \ldots + \mathsf{cn}(\mathscr{c} \rightarrowtail A) + \mathsf{cn}(\mathscr{d} \rightarrowtail B) + \ldots + \mathsf{cn}(L_m) \\
&\qquad = \mathsf{cn}(L_1) + \ldots + \mathsf{cn}(A) + \mathsf{cn}(B) + \ldots + \mathsf{cn}(L_m) \\
&\qquad < \mathsf{cn}(L_1) + \ldots + \mathsf{cn}(A) + \mathsf{cn}(B) + 1 + \ldots + \mathsf{cn}(L_m) \\
&\qquad = \mathsf{cn}(L_1) + \ldots + \mathsf{cn}(\mathscr{b} \rightarrowtail A \vee B) + \ldots + \mathsf{cn}(L_m) \\
&\qquad = \mathsf{ms}\big(\{\vdash L_1, \ldots \vdash \mathscr{b} \rightarrowtail A \vee B, \ldots \vdash L_m\}\big).
\end{aligned}$$

- $L_i = \mathscr{b} \leftarrowtail A \vee B$, $L_i = \mathscr{b} \rightarrowtail A \wedge B$, $L_i = \mathscr{b} \leftarrowtail A \wedge B$. Analogous to the case above.
- $L_i = \mathscr{b} \rightarrowtail \mathbf{C}^q A$. Assume that $\vdash L_i$ is the active sequent of the given $\rightsquigarrow$-decomposition and that $\rightsquigarrow$ is in its turn based on the $\rightsquigarrow_0$-decomposition below:

$$\vdash \mathscr{b} \rightarrowtail \mathbf{C}^q A \quad \rightsquigarrow_0 \quad \{\vdash \mathscr{c} \rightarrowtail A\}$$

where $\mu\big(\llbracket \mathscr{c} \rrbracket\big) \geq q$. Then,

$$\begin{aligned}
&\mathsf{ms}\big(\{\vdash L_1, \dots \vdash \mathscr{c} \rightarrowtail A, \dots \vdash L_m\}\big)\\
&\qquad = \mathsf{cn}(L_1) + \dots + \mathsf{cn}(\mathscr{c} \rightarrowtail A) + \dots + \mathsf{cn}(L_m)\\
&\qquad = \mathsf{cn}(L_1) + \dots + \mathsf{cn}(A) + \dots + \mathsf{cn}(L_m)\\
&\qquad < \mathsf{cn}(L_1) + \dots + \mathsf{cn}(A) + 1 + \dots + \mathsf{cn}(L_m)\\
&\qquad = \mathsf{cn}(L_1) + \dots + \mathsf{cn}(\mathscr{b} \rightarrowtail \mathbf{C}^q A) + 1 + \dots + \mathsf{cn}(L_m)\\
&\qquad = \mathsf{ms}\big(\{\vdash L_1, \dots \vdash \mathscr{b} \rightarrowtail \mathbf{C}^q A, \dots \vdash L_m\}\big).
\end{aligned}$$

- $L_i = \mathscr{b} \leftarrowtail \mathbf{C}^q A$, $L_i = \mathscr{b} \rightarrowtail \mathbf{D}^q A$, $L_i = \mathscr{b} \leftarrowtail \mathbf{D}^q A$. Similar to the case above.
- $L_i = \mathscr{b} \rightarrowtail A$. Assume that $\vdash L_i$ is the active sequent in the given $\rightsquigarrow$-decomposition and that $\rightsquigarrow$ is based on the $\rightsquigarrow_0$-decomposition below:

$$\vdash \mathscr{b} \rightarrowtail A \quad \rightsquigarrow_0 \quad \{\}$$

  where $\mu\big([\![\mathscr{b}]\!]\big) = 0$. Since for Definitions 11 and 12, $\mathsf{cn}\big(\{\}\big) = 0$ and $\mathsf{cn}(\vdash \mathscr{b} \leftarrowtail A) > 0$, clearly $\mathsf{cn}(\{\}) < \mathsf{cn}(\vdash \mathscr{b} \leftarrowtail A)$.
- $L_i = \mathscr{b} \leftarrowtail A$, $L_i = \mathscr{b} \rightarrowtail \mathbf{D}^0 A$, $L_i = \mathscr{b} \leftarrowtail \mathbf{C}^0 A$. Analogous to the case above. □

The Lemma below establishes Claim 1.

Lemma 3. *For all $\rightsquigarrow$-normal sequent $\vdash L$, $\vDash L$ if and only if $\vdash_{\mathbf{CLK}_0} L$.*

Proof. First observe that every $\rightsquigarrow$-normal sequent is basic (by inspection of possible cases). Let now $\vdash L$ be a basic sequent.

$\Rightarrow$ Assume that $\vDash L$. Since L is a basic formula, there are two main cases - Ax1 and Ax2 - which are both trivial.
$\Leftarrow$ Assume that $\vdash L$ is derivable in $\mathbf{CLK}_0$. Then, by Proposition 3, $\vDash L$. □

All the given results can be extended from sequents to sets in a natural way, obtaining in particular that, if a set of sequents is $\rightsquigarrow$-normal, i.e. each of its sequents is $\rightsquigarrow$-normal, then it is valid if and only if it is derivable, namely its sequents are valid if and only if they are derivable.

In order to prove Claim 2. we need to show that validity is existentially preserved through $\rightsquigarrow$-decomposition.

Lemma 4. *Each valid sequent has a valid $\rightsquigarrow$-normal form.*

Proof. Let $\vdash L$ be an arbitrary valid sequent. It is shown that $\vdash L$ has a valid $\rightsquigarrow$-normal form. Since $\rightsquigarrow_0$ is strongly normalizing, it suffices to check that for each possible sequent which is not $\rightsquigarrow_0$-normal, there is a $\rightsquigarrow_0$-reduction which preserves validity.

- Assume that $\vdash L$ is such that no $\rightsquigarrow_0$-reduction can be applied on it. Then, as observed in the proof of Lemma 3, the sequent is either empty or basic; in other words L is a basic formula. In both cases, the sequent is $\rightsquigarrow$-normal and, for hypothesis, valid.
- Assume now that $\vdash L$ is $\rightsquigarrow$-reducible. The argument is based on the exhaustive inspection of all possible forms of $\rightsquigarrow_0$-reduction, exploiting Remark 2. We only consider a few interesting cases:

  - $L = \mathscr{b} \rightarrowtail \neg A$. Let $\mathscr{c} = \neg \mathscr{b}$; we have $[\![\neg \mathscr{c}]\!] = 2^{\mathbb{N}} - [\![\mathscr{c}]\!] = 2^{\mathbb{N}} - (2^{\mathbb{N}} - [\![\mathscr{b}]\!]) = [\![\mathscr{b}]\!]$ and so, in particular, $\mathscr{b} \vDash \neg \mathscr{c}$. Let us consider the following well-defined $\rightsquigarrow_0$-decomposition (given $\mathscr{b} \vDash \neg \mathscr{c}$):

$$\vdash \mathscr{b} \rightarrowtail \neg A \quad \rightsquigarrow_0 \quad \{\vdash \mathscr{c} \leftarrowtail A\}.$$

    Since $\mathscr{b} \rightarrowtail \neg A$ is valid, $[\![\mathscr{b}]\!] \subseteq [\![\neg A]\!]$, whence $[\![A]\!] \subseteq [\![\neg \mathscr{b}]\!] = [\![\mathscr{c}]\!]$ that is $\vDash \mathscr{c} \leftarrowtail A$ as desired.
  - $L = \mathscr{b} \rightarrowtail A \vee B$. Using Remark 2, from the hypothesis $\vDash \mathscr{b} \rightarrowtail A \vee B$, that is, $[\![\mathscr{b}]\!] \subseteq [\![A]\!] \cup [\![B]\!]$, we deduce $[\![\mathscr{b}]\!] \subseteq [\![\mathscr{b}_A]\!] \cup [\![\mathscr{b}_B]\!]$, that is $\mathscr{b} \vDash \mathscr{b}_A \vee \mathscr{b}_B$. Let us consider the following reduction, which is well-defined (given $\mathscr{b} \vDash \mathscr{b}_A \vee \mathscr{b}_B$):

$$\vdash \mathscr{b} \rightarrowtail A \vee B \quad \rightsquigarrow_0 \quad \{\vdash \mathscr{b}_A \rightarrowtail A, \vdash \mathscr{b}_B \rightarrowtail B\}.$$

    Since $[\![\mathscr{b}_A]\!] = [\![A]\!]$ and $[\![\mathscr{b}_B]\!] = [\![B]\!]$ we then have $[\![\mathscr{b}_A]\!] \subseteq [\![A]\!]$ and $[\![\mathscr{b}_B]\!] \subseteq [\![B]\!]$. Therefore, $\vDash \mathscr{b}_A \rightarrowtail A$ and $\vDash \mathscr{b}_B \rightarrowtail B$, as desired.
  - $L = \mathscr{b} \leftarrowtail A \vee B$. Let us consider the following, well-defined $\rightsquigarrow_0$-decomposition:

$$\vdash \mathscr{b} \leftarrowtail A \vee B \quad \rightsquigarrow_0 \quad \{\vdash \mathscr{b} \leftarrowtail A, \vdash \mathscr{b} \leftarrowtail B\}.$$

    For hypothesis $\vDash \mathscr{b} \leftarrowtail A \vee B$, which is $[\![A]\!] \cup [\![B]\!] \subseteq [\![\mathscr{b}]\!]$. Then, by basic set theory, both $[\![A]\!] \subseteq [\![\mathscr{b}]\!]$ and $[\![B]\!] \subseteq [\![\mathscr{b}]\!]$, which is $\vDash \mathscr{b} \leftarrowtail A$ and $\vDash \mathscr{b} \leftarrowtail B$, as desired.

- $L = \mathscr{b} \rightarrowtail \mathbf{C}^q A$. There are two main sub-cases:

  - Let $\mu(\llbracket \mathscr{b} \rrbracket) = 0$. Then the sequent can be decomposed by means of the following well-defined $\rightsquigarrow_0$-decomposition:

    $$\vdash \mathscr{b} \rightarrowtail \mathbf{C}^q A \quad \rightsquigarrow_0 \quad \{\},$$

    where $\{\}$ is vacuously valid.
  - Let $\mu(\llbracket \mathscr{b} \rrbracket) \neq 0$. For hypothesis $\vDash \mathscr{b} \rightarrowtail \mathbf{C}^q A$, that is $\llbracket \mathscr{b} \rrbracket \subseteq \llbracket \mathbf{C}^q A \rrbracket$. Since $\llbracket \mathscr{b} \rrbracket \neq \emptyset$, also $\llbracket \mathbf{C}^q A \rrbracket \neq \emptyset$, that is $\llbracket \mathbf{C}^q A \rrbracket = 2^{\mathbb{N}}$, so $\mu(\llbracket A \rrbracket) \geq q$. Then $\mu(\llbracket \mathscr{b}_A \rrbracket) \geq q$ and the following $\rightsquigarrow_0$-decomposition is well-defined:

    $$\vdash \mathscr{b} \rightarrowtail \mathbf{C}^q A \quad \rightsquigarrow_0 \quad \{\mathscr{b}_A \rightarrowtail A\}.$$

    By construction $\llbracket \mathscr{b}_A \rrbracket = \llbracket A \rrbracket$ so, in particular, $\llbracket \mathscr{b}_A \rrbracket \subseteq \llbracket A \rrbracket$, that is $\vDash \mathscr{b}_A \rightarrowtail A$, as desired.

- $L = \mathscr{b} \leftarrowtail \mathbf{C}^q A$. There are two main sub-cases:

  - Let $\mu(\llbracket \mathscr{b} \rrbracket) = 1$. Then, the sequent can be decomposed by means of the following well-defined $\rightsquigarrow_0$-decomposition:

    $$\vdash \mathscr{b} \leftarrowtail \mathbf{C}^q A \quad \rightsquigarrow_0 \quad \{\},$$

    where $\{\}$ is vacuously valid.
  - Let $\mu(\llbracket \mathscr{b} \rrbracket) \neq 1$. By hypothesis $\vDash \mathscr{b} \leftarrowtail \mathbf{C}^q A$, that is $\llbracket \mathbf{C}^q A \rrbracket \subseteq \llbracket \mathscr{b} \rrbracket$. Since $\llbracket \mathscr{b} \rrbracket \neq 2^{\mathbb{N}}$, $\llbracket \mathbf{C}^q A \rrbracket = \emptyset$. So, $\mu(\llbracket A \rrbracket) < q$. Then $\mu(\llbracket \mathscr{b}_A \rrbracket) < q$ and the following decomposition is well-defined:

    $$\vdash \mathscr{b} \leftarrowtail \mathbf{C}^q A \quad \rightsquigarrow_0 \quad \{\mathscr{b}_A \leftarrowtail A\}.$$

    For construction $\llbracket \mathscr{b}_A \rrbracket = \llbracket A \rrbracket$ so, in particular, $\llbracket A \rrbracket \subseteq \llbracket \mathscr{b}_A \rrbracket$, that is $\vDash \mathscr{b}_A \leftarrowtail A$ as desired. □

It now remains to establish Claim 3. Let $\vdash_{\mathbf{CLK}_0} \Phi$ indicate that $\vdash_{\mathbf{CLK}_0} L$ holds for all $L \in \Phi$.

Proposition 4. *Given two sets of sequents Φ and Ψ, if $\Phi \rightsquigarrow \Psi$, then $\vdash_{\mathbf{CLK}_0} \Psi$ implies $\vdash_{\mathbf{CLK}_0} \Phi$.*

Proof sketch. Assume $\Phi \rightsquigarrow \Psi$. Then, there is a $\vdash L \in \Phi$ such that it is the "active" sequent on which $\rightsquigarrow$ is based, that is $\rightsquigarrow$ is based on the $\rightsquigarrow_0$-decomposition below:

$$\vdash L \quad \rightsquigarrow_0 \quad \{\vdash L_1, \dots, \vdash L_n\}.$$

The proof is by straightforward inspection of all possible forms of $\rightsquigarrow_0$-reduction. □

Putting these results together, it is possible to conclude that $\mathbf{CLK}_0$ is complete with respect to the semantics of CPL_0.

Proposition 5 *(Completeness). $\vDash L$ implies $\vdash_{\mathbf{CLK}_0} L$.*

Proof. If the sequent $\vdash L$ is valid, by Lemma 4, it has a *valid* $\rightsquigarrow$-normal form. By Lemma 3, a $\rightsquigarrow$-normal form is valid if and only if it is derivable, so the given $\rightsquigarrow$-normal form must be derivable as well. Therefore, by Proposition 4, $\vdash L$ must be derivable in $\mathbf{CLK}_0$. □

Remark 1. As a consequence of the completeness theorem, the following cut-rule turns out to be derivable in $\mathbf{CLK}_0$:

$$\frac{\vdash \mathscr{c} \rightarrowtail \neg A \vee B \qquad \vdash \mathscr{d} \rightarrowtail A \qquad \mathscr{b} \vDash \mathscr{c} \wedge \mathscr{d}}{\vdash \mathscr{b} \rightarrowtail B}\ \mathsf{Cut}$$

5. Multivariate counting propositional logic: syntax and semantics

As is well-known, counting problems are not restricted to those in $\mathsf{P}^{\sharp\mathtt{SAT}}$. For instance, one can consider problems concerning relations between valuations of *different* groups of variables, like `MajMajSAT`, see [9,34,35]: given a formula A of PL containing two disjoint sets of variables, $\mathbf{x}$ and $\mathbf{y}$, this problem asks whether for at least half of the valuations of $\mathbf{x}$, at least half of the valuations of $\mathbf{y}$ makes A true.

To express these kinds of problems, we consider a language in which propositional atoms and counting quantifiers are *named*. We use $a, b, c, \dots$ for names. Counting quantifiers, indicated as $\mathbf{C}^q_a A$ or $\mathbf{D}^q_a A$, now depend on the number of valuations of propositional atoms *with name a* satisfying A.

Definition 13 *(Formula of **CPL**).* The *formulas of **CPL*** are defined by the following grammar:

$$A, B ::= \mathbf{i}_a \mid \neg A \mid A \wedge B \mid A \vee B \mid \mathbf{C}^q_a A \mid \mathbf{D}^q_a A$$

where $i \in \mathbb{N}$, a is a name, and $q \in \mathbb{Q}_{[0,1]}$.

The intuitive meaning of named quantifiers is that they count models *relative* to the corresponding bounded variables. Named quantifiers, $\mathbf{C}^q_a$ and $\mathbf{D}^q_a$, bind the occurrences of the name a in A. Formulas are thus taken modulo α-equivalence. Given a formula A of **CPL**, we let $\mathrm{FN}(A)$ indicate the set of names occurring *free* (i.e. not bound) in A.

Names can be used to distinguish between distinct groups of propositional variables. For example, the propositional formula $F = (x_1 \vee y_1) \wedge (x_2 \vee y_2)$, containing two groups of variables $\mathbf{x} = \{x_1, x_2\}$ and $\mathbf{y} = \{y_1, y_2\}$, can be expressed in **CPL** using two distinct names a, b as $G = (\mathbf{1}_a \vee \mathbf{1}_b) \wedge (\mathbf{2}_a \vee \mathbf{2}_b)$. Since the intuitive meaning of $\mathbf{C}^q_a A$ is that A is true in at least q of the valuations of the variables with name a, we can take the **CPL**-formula $\mathbf{C}^{1/2}_a \mathbf{C}^{1/2}_b G$ as expressing the `MajMajSAT` problem for F (which happens to have a positive answer, in this case).

While the formulas $\mathbf{C}^q_a A$ and $\mathbf{D}^q_a A$ have a rather intuitive meaning, the semantics of **CPL** is slightly subtler than in case of $\mathbf{CPL}_0$. First observe that the interpretation of a formula A now depends on the choice of a finite set of names $X \supseteq \mathrm{FN}(A)$, and yields a measurable set $[\![A]\!]_X$ belonging to the Borel algebra $\mathscr{B}\big((2^{\mathbb{N}})^X\big)$ (generated by the product topology over "X" copies of $2^{\mathbb{N}}$). Also in this case, there exists a canonical Lebesgue measure μ_X over $\mathscr{B}\big((2^{\mathbb{N}})^X\big)$ giving measure $\frac{1}{2}$ to all cylinders (see [10]) of the form

$$\mathrm{Cyl}(a, i) = \{f \in (2^{\mathbb{N}})^X \mid f(a)(i) = 1\} \qquad (a \in X, i \in \mathbb{N})$$

Hence, the quantifiers $\mathbf{C}^q_a$ and $\mathbf{D}^q_a$ must correspond to operations allowing one to pass, for any set of names X not containing a, from subsets of $(2^{\mathbb{N}})^{X \cup \{a\}}$ to subsets of $(2^{\mathbb{N}})^X$. To define such operations we need the following technical notion:

Definition 14 *(f-projection).* Let X, Y be two disjoint finite sets of names and $f \in (2^{\mathbb{N}})^X$. For all $\mathcal{X} \subseteq (2^{\mathbb{N}})^{X \cup Y}$, the *$f$-projection of $\mathcal{X}$* is the set $\Pi_f(\mathcal{X}) \subseteq (2^{\mathbb{N}})^Y$ defined as follows:

$$\Pi_f(\mathcal{X}) = \{g \in (2^{\mathbb{N}})^Y \mid f + g \in \mathcal{X}\},$$

where $(f+g)(a)$ is $f(a)$, if $a \in X$ and $g(a)$ if $a \in Y$.

Suppose X and Y are disjoint sets of names, with $\mathrm{FN}(A) \subseteq X \cup Y$. Then, if we fix a valuation $f \in (2^{\mathbb{N}})^X$ of the variables of A with names in X, the set $\Pi_f\big([\![A]\!]_{X \cup Y}\big)$ describes the set of valuations of the variables of A with names in Y which extend f.

In general, even if $\mathcal{X} \in \mathscr{B}((2^{\mathbb{N}})^{X \cup Y})$ is a Borel set, the projection $\Pi_f(\mathcal{X}) \subseteq (2^{\mathbb{N}})^X)$ needs not be Borel. Indeed, Π_f does not define a map from $\mathscr{B}((2^{\mathbb{N}})^{X \cup Y})$ to $\mathscr{B}((2^{\mathbb{N}})^X)$, but from $\Sigma^1_1((2^{\mathbb{N}})^{X \cup Y})$ to $\Sigma^1_1((2^{\mathbb{N}})^X)$, where $\Sigma^1_1((2^{\mathbb{N}})^X)$ indicates the class of *analytic* subsets of $(2^{\mathbb{N}})^X$ (see [27]). Importantly, the Lebesgue measure is always defined on $\Sigma^1_1((2^{\mathbb{N}})^X)$. Moreover, we will exploit the following result:

Lemma 5 *([27], Theorem 14.11 & Theorem 29.26). For any $\mathcal{X} \in \mathscr{B}((2^{\mathbb{N}})^{X \cup Y})$, with $X \cap Y = \emptyset$, and $r \in [0, 1]$, $\{f \in (2^{\mathbb{N}})^X \mid \mu(\Pi_f(\mathcal{X})) \geq r\} \in \mathscr{B}((2^{\mathbb{N}})^X)$.*

We can now define the semantics of **CPL**:

Definition 15 *(Semantics of **CPL**).* For each formula A of **CPL**, and finite set of names such that $X \supseteq \mathrm{FN}(A)$, the *interpretation* of A is a Borel set $[\![A]\!]_X \in \mathscr{B}((2^{\mathbb{N}})^X)$ inductively defined as follows:

$$\begin{aligned}
[\![\mathbf{i}_a]\!]_X &= \mathrm{Cyl}(a, i) & \qquad [\![\neg A]\!]_X &= (2^{\mathbb{N}})^X - [\![A]\!]_X \\
[\![A \wedge B]\!]_X &= [\![A]\!]_X \cap [\![B]\!]_X & \qquad [\![\mathbf{C}^q_a A]\!]_X &= \{f \mid \mu\big(\Pi_f([\![A]\!]_{X \cup \{a\}})\big) \geq q\} \\
[\![A \vee B]\!]_X &= [\![A]\!]_X \cup [\![B]\!]_X & \qquad [\![\mathbf{D}^q_a A]\!]_X &= \{f \mid \mu\big(\Pi_f([\![A]\!]_{X \cup \{a\}})\big) < q\}.
\end{aligned}$$

A formula A is *valid* when $[\![A]\!]_{\mathrm{FN}(A)} = (2^{\mathbb{N}})^{\mathrm{FN}(A)}$. Two formulas A, B are *logically equivalent* (noted $A \equiv B$) when $[\![A]\!]_{\mathrm{FN}(A) \cup \mathrm{FN}(B)} = [\![B]\!]_{\mathrm{FN}(A) \cup \mathrm{FN}(B)}$.

The well-definedness of the Borel sets $[\![A]\!]_X$ follows from Lemma 5, in the crucial cases of the sets $[\![\mathbf{C}^q_a A]\!]_X$ and $[\![\mathbf{D}^q_a A]\!]_X$. However, as a consequence of the Fundamental Lemma below, we will show at the end of this section a more direct proof of the fact that all the sets $[\![A]\!]_X$ are Borel. To have a grasp of the semantics of named quantifiers, consider the following example:

Example 3. Let A be the following formula of **CPL**:

$$A = \big(\mathbf{0}_a \wedge (\neg \mathbf{0}_b \wedge \mathbf{1}_b)\big) \vee (\neg \mathbf{0}_a \wedge \mathbf{0}_b \wedge \neg \mathbf{1}_b) \vee \big((\neg \mathbf{0}_a \wedge \mathbf{1}_a) \wedge \mathbf{1}_b\big).$$

The valuations $f \in (2^{\mathbb{N}})^{\{b\}}$ belonging to $[\![\mathbf{C}_a^{1/2} A]\!]_{\{b\}}$ are those which can be extended to valuations of all Boolean variables in A satisfying A in at least half of the cases. In such a simple situation, we can list all possible cases:

1. if $f(b)(0) = f(b)(1) = 1$, then A has $\frac{1}{4}$ chances of being true, since both $\neg \mathbf{0}_a$ and $\mathbf{1}_a$ must be true,
2. if $f(b)(0) = 1$, $f(b)(1) = 0$, then A has $\frac{1}{2}$ chances of being true, since $\neg \mathbf{0}_a$ must be true,
3. if $f(b)(0) = 0$, $f(b)(1) = 1$, then A has $\frac{3}{4}$ chances of being true, since either $\mathbf{0}_a$ or both $\neg \mathbf{0}_a$ and $\mathbf{1}_a$ must be true,
4. if $f(b)(0) = f(b)(1) = 0$, then A has 0 chances of being true.

Thus, $[\![\mathbf{C}_a^{1/2} A]\!]_{\{b\}}$ only contains the valuations which agree with cases 2. and 3. Therefore, $[\![\mathbf{C}_b^{1/2} \mathbf{C}_a^{1/2} A]\!]_{\emptyset} = 2^{\mathbb{N}}$, that is $\mathbf{C}_b^{1/2} \mathbf{C}_a^{1/2} A$ is valid, since half of the valuations of b has at least $\frac{1}{2}$ chances of being extended to a model of A.

The definition of the sets $[\![\mathbf{C}_a^q A]\!]_X$ and $[\![\mathbf{D}_a^q A]\!]_X$ is not very intuitive at first glance. We now provide an alternative characterization of these sets by means of *named* Boolean formulas, and show that they are measurable.

Definition 16 *(Named Boolean formula). Named Boolean formulas* are defined by the following grammar:

$$\mathscr{b}, \mathscr{c} := x_i^a \mid \top \mid \bot \mid \neg \mathscr{b} \mid \mathscr{b} \wedge \mathscr{c} \mid \mathscr{b} \vee \mathscr{c}.$$

The interpretation $[\![\mathscr{b}]\!]_X$ of the Boolean formula $\mathscr{b}$ with $\mathrm{FN}(\mathscr{b}) \subseteq X$ is defined in a straightforward way, mimicking Definition 2:

$$[\![x_i^a]\!]_X := \mathrm{Cyl}(a, i) \qquad [\![\neg \mathscr{b}]\!]_X := (2^{\mathbb{N}})^X - [\![\mathscr{b}]\!]_X$$
$$[\![\top]\!]_X := (2^{\mathbb{N}})^X \qquad [\![\mathscr{b} \wedge \mathscr{c}]\!]_X := [\![\mathscr{b}]\!]_X \cap [\![\mathscr{c}]\!]_X$$
$$[\![\bot]\!]_X := \emptyset^X \qquad [\![\mathscr{b} \vee \mathscr{c}]\!]_X := [\![\mathscr{b}]\!]_X \cup [\![\mathscr{c}]\!]_X.$$

Remark 2. A result analogous to Lemma 1 holds also for named Boolean formulas. A consequence of this fact is that the measure $\mu([\![\mathscr{b}]\!]_X)$ does not depend on the choice of X. For this reason, we will still indicate the measure of a named Boolean formula as $\mu([\![\mathscr{b}]\!])$, as we did for un-named formulas.

Let us now introduce the crucial notion of a-decomposition:

Definition 17 *(a-decomposition).* Let $\mathscr{b}$ be a named Boolean formula with free names in $X \cup \{a\}$. An *a-decomposition of* $\mathscr{b}$ is a Boolean formula $\mathscr{c} = \bigvee_{i=0}^{k-1} \mathscr{d}_i \wedge \mathscr{e}_i$ such that:

- $[\![\mathscr{b}]\!] = [\![\mathscr{c}]\!]$;
- $\mathrm{FN}(\mathscr{d}_i) \subseteq \{a\}$; and $\mathrm{FN}(\mathscr{e}_i) \subseteq X$,
- if $i \neq j$, then $[\![\mathscr{e}_i]\!] \cap [\![\mathscr{e}_j]\!] = \emptyset$.

The following lemma provides each Boolean formula with an a-decomposition.

Lemma 6. *Any named Boolean formula $\mathscr{b}$ with* $\mathrm{FN}(\mathscr{b}) \subseteq X \cup \{a\}$ *admits an a-decomposition in X.*

Proof. We will actually prove a stronger statement saying that any named Boolean formula $\mathscr{b}$ admits an a-decomposition $\bigvee_{i=1}^{k} \mathscr{d}_i \wedge \mathscr{e}_i$, such that $\vDash \bigvee_{i=1}^{j} \mathscr{e}_i$ holds. We proceed by induction on $\mathscr{b}$:

- if $\mathscr{b} = x_i^a$ or $\mathscr{b} = \neg x_i^a$, then $k = 1$, $\mathscr{d}_i = \mathscr{b}$ and $\mathscr{e}_i = \top$.
- if $\mathscr{b} = x_i^b$, where $b \neq a$, then $k = 2$, $\mathscr{d}_0 = \top$, $\mathscr{e}_1 = \bot$ and $\mathscr{e}_0 = \mathscr{b}$, $\mathscr{e}_1 = \neg \mathscr{b}$.
- if $\mathscr{b} = \mathscr{b}_1 \vee \mathscr{b}_2$ then, by IH, $\mathscr{b}_1 = \bigvee_{i=1}^{k_1} \mathscr{d}_i^1 \wedge \mathscr{e}_i^1$ and $\mathscr{b}_2 = \bigvee_{j=1}^{k_2} \mathscr{d}_j^2 \wedge \mathscr{e}_j^2$. Then,

$$\begin{aligned} \mathscr{b} &\equiv \Big(\bigvee_{i=0}^{k_1} \mathscr{d}_i^1 \wedge \mathscr{e}_i^1\Big) \vee \Big(\bigvee_{j=1}^{k_2} \mathscr{d}_j^2 \wedge \mathscr{e}_j^2\Big) \\ &\equiv \Big(\bigvee_{i=1}^{k_1} \mathscr{d}_i^1 \wedge \mathscr{e}_i^1 \wedge \top\Big) \vee \Big(\bigvee_{j=1}^{k_2} \mathscr{d}_j^2 \wedge \mathscr{e}_j^2 \wedge \top\Big) \end{aligned}$$

$$
\begin{aligned}
&\equiv \Big(\bigvee_{i=1}^{k_1} \mathscr{d}_i^1 \wedge \mathscr{e}_i^1 \wedge \bigvee_{j} \mathscr{e}_j^2\Big) \vee \Big(\bigvee_{j=1}^{k_2} \mathscr{d}_j^2 \wedge \mathscr{e}_j^2 \wedge \bigvee_{i} \mathscr{e}_i^1\Big) \\
&\equiv \Big(\bigvee_{i=1,j=1}^{k_1,k_2} (\mathscr{d}_i^1 \wedge \mathscr{e}_i^1 \wedge \mathscr{e}_j^2)\Big) \vee \Big(\bigvee_{j=1,i=1}^{k_2,k_1} (\mathscr{d}_j^2 \wedge \mathscr{e}_j^2 \wedge \mathscr{e}_i^1)\Big) \\
&\equiv \bigvee_{i=1,j=1}^{k_1,k_2} (\mathscr{d}_i^1 \vee \mathscr{d}_j^2) \wedge (\mathscr{e}_i^1 \wedge \mathscr{e}_j^2).
\end{aligned}
$$

Let $k = k_1 \cdot k_2$. We can identify any $l \le k-1$ with a pair (i, j), $i < k_1$ and $j < k_2$. Let $\mathscr{d}, j = \mathscr{d}_i^1 \vee \mathscr{d}_j^2$ and $\mathscr{e}_{i,j} = \mathscr{e}_i^1 \vee \mathscr{e}_j^2$. Then,

$$
\mathscr{b} \equiv \bigvee_{i=1,j=1}^{k_1,k_2} \mathscr{d}_{i,j} \wedge \mathscr{e}_{i,j}.
$$

Observe that for $(i, j) \neq (i', j')$, $\mathscr{e}_{i,j} \wedge \mathscr{e}_{i',j'} \equiv \bot$. Moreover, $\bigvee_{i,j} \mathscr{e}_{i,j} \equiv \bigvee_{i,j} \mathscr{e}_i^1 \vee \mathscr{e}_j^2 \equiv \big(\bigvee_i \mathscr{e}_i^1\big) \vee \big(\bigvee_j \mathscr{e}_j^2\big) \equiv \top \vee \top \equiv \top$.

- if $\mathscr{b} = \mathscr{b}_1 \wedge \mathscr{b}_2$, then, by IH, $\mathscr{b}_1 \equiv \bigvee_{i=1}^{k_1} \mathscr{d}_i^1 \wedge \mathscr{e}_i^1$ and $\mathscr{b}_2 \equiv \bigvee_{j=1}^{k_2} \mathscr{d}_j^2 \wedge \mathscr{e}_j^2$. Then,

$$
\begin{aligned}
\mathscr{b} &\equiv \Big(\bigvee_{i=1}^{k_1} \mathscr{d}_i^1 \wedge \mathscr{e}_i^1\Big) \wedge \Big(\bigvee_{j=1}^{k_2} \mathscr{d}_j^2 \wedge \mathscr{e}_j^2\Big) \\
&\equiv \bigvee_{i=1,j=1}^{k_1,k_2} \mathscr{d}_i^1 \wedge \mathscr{e}_i^1 \wedge \mathscr{d}_j^2 \wedge \mathscr{e}_j^2 \\
&\equiv \bigvee_{i=1,j=1}^{k_1,k_2} (\mathscr{d}_i^1 \wedge \mathscr{d}_j^2) \wedge (\mathscr{e}_i^1 \wedge \mathscr{e}_j^2).
\end{aligned}
$$

As in the case above, let $k_1 \cdot k_2$. We can identify any $l \le k-1$ with a pair (i, j), $i < k_1$ and $j < k_2$. □

It is worth observing that, while an a-decomposition of $\mathscr{b}$ always exists, it needs not be feasibly found, since this formula can be of exponential length with respect to $\mathscr{b}$. Yet, a-decompositions can be used to show that the interpretation of a quantified formula is a finite union of measurable sets:

Lemma 7 *(Fundamental Lemma). Let $\mathscr{b}$ be a named Boolean formula with* $\mathrm{FN}(\mathscr{b}) \subseteq X \cup \{a\}$ *and* $\mathscr{c} = \bigvee_{i=1}^{k} \mathscr{d}_i \wedge \mathscr{e}_i$ *be an a-decomposition of $\mathscr{b}$. Then, for all $q \in \mathbb{Q}_{[0,1]}$,*

$$
\begin{aligned}
\{f \in (2^{\mathbb{N}})^X \mid \mu\big(\Pi_f(\llbracket \mathscr{b} \rrbracket)\big) \ge q\} &= \bigcup \big\{\llbracket \mathscr{e}_i \rrbracket_X \mid \mu\big(\llbracket \mathscr{d}_i \rrbracket\big) \ge q\big\} \\
\{f \in (2^{\mathbb{N}})^X \mid \mu\big(\Pi_f(\llbracket \mathscr{b} \rrbracket)\big) < q\} &= \bigcup \big\{\llbracket \mathscr{e}_i \rrbracket_X \mid \mu\big(\llbracket \mathscr{d}_i \rrbracket\big) < q\big\}.
\end{aligned}
$$

Proof. We only prove the first equality, the second one being established in a similar way. First, note that if $q = 0$, then both sets are equal to $(2^{\mathbb{N}})^X$, so we can suppose $q > 0$.

- $\subseteq$ Suppose $\mu\big(\Pi_f(\llbracket \mathscr{b} \rrbracket_{X\cup\{a\}})\big) \ge q$. Then, $\Pi_f\big(\llbracket \mathscr{b} \rrbracket_{X\cup\{a\}}\big)$ is non-empty and from $\mathscr{b} \equiv \bigvee_i^k \mathscr{d}_i \wedge \mathscr{e}_i$, we deduce that there exists an $i \le k$ such that $f \in \llbracket \mathscr{e}_i \rrbracket_X$ and for all $g \in \llbracket \mathscr{d}_i \rrbracket_{\{a\}}$, $f + g \in \llbracket \mathscr{d}_i \wedge \mathscr{e}_i \rrbracket_{X\cup\{a\}}$. This implies then that $\llbracket \mathscr{d}_i \rrbracket_{\{a\}} \subseteq \Pi_f\big(\llbracket \mathscr{b} \rrbracket_{X\cup\{a\}}\big)$. Moreover, since the sets $\llbracket \mathscr{e}_i \rrbracket_X$ are pairwise disjoint, for all $j \neq i$, $f \notin \llbracket \mathscr{e}_j \rrbracket_X$, which implies that $\Pi_f\big(\llbracket \mathscr{b} \rrbracket_{X\cup\{a\}}\big) \subseteq \llbracket \mathscr{d}_i \rrbracket_{\{a\}}$. Hence, $\Pi_f\big(\llbracket \mathscr{b} \rrbracket_{X\cup\{a\}}\big) = \llbracket \mathscr{d}_i \rrbracket_{\{a\}}$, which implies $\mu\big(\llbracket \mathscr{d}_i \rrbracket\big) \ge q$.
- $\supseteq$ If $f \in \llbracket \mathscr{e}_i \rrbracket_X$, where $\mu\big(\llbracket \mathscr{d}_i \rrbracket\big) \ge q$, then, since $\mathscr{d}_i \wedge \mathscr{e}_i \vDash \mathscr{b}$, we have that $\mu\big(\Pi_f(\llbracket \mathscr{b} \rrbracket_{X\cup\{a\}})\big) \ge \mu\big(\Pi_f(\llbracket \mathscr{d}_i \wedge \mathscr{e}_i \rrbracket_{X\cup\{a\}})\big) = \mu\big(\llbracket \mathscr{d}_i \rrbracket\big) \ge q$. □

An important consequence of the Fundamental Lemma is the following result.

Corollary 1. *For any formula A of **CPL** there exists a named Boolean formula $\mathscr{b}_A$ such that $A \equiv \mathscr{b}_A$.*

Proof. We construct $\mathscr{b}_A$ by induction on A. The only non-trivial cases are when $A = \mathbf{C}_a^q B$ and $A = \mathbf{D}_a^q B$. For the first case, by IH $B \equiv \mathscr{b}_B$, for some named Boolean formula $\mathscr{b}_B$; let $\bigvee_i \mathscr{d}_i \wedge \mathscr{e}_i$ be an a-decomposition of $\mathscr{b}_B$, and let $I = \{i_1, \dots, i_k\}$ be the set of indexes i such that $\mu([\![d_i]\!]) \geq q$. Then, by the Fundamental Lemma, we can take $b_A = e_{i_1} \vee \dots \vee e_{i_k}$. For the second case we can argue in a similar way. $\square$

The result above is analogous to the case of **CPL**$_0$ - cf. Proposition 2. However, contrarily to that case, there is no obvious way to compute b_A from A in polynomial time (even with an access to an oracle for $\sharp$`SAT`), as the computation of a-decompositions for all quantified sub-formulas of A is also required.

Finally, Corollary 1 immediately yields a direct argument showing that the sets interpreting **CPL**-formulas are Borel sets, as they correspond to the interpretation of named Boolean formulas (hence they are all obtained by applying finite unions and intersections to cylinders).

6. Multivariate counting logic: correspondence with CH

We have already seen that the problem `MajMajSAT` can be "captured" using formulas of the form $\mathbf{C}^q_a\mathbf{C}^r_b A$, where A is quantifier-free. We now extend this result to all levels of CH by considering formulas in which an arbitrary number of counting quantifiers may occur. We proceed in three steps. First, we show that any formula of **CPL** can be put in *prenex normal form*, that is, that all counting quantifiers can be moved at top-level. Next, we prove that the quantifier **D**, which has no counterpart in Wagner's problems, can be eliminated. Finally, exploiting a result from [56], we show that prenex formulas with k nested **C**-quantifiers characterize the level k of CH.

6.1. Prenex normal forms

We show that any formula of **CPL** can be converted into prenex normal form. So, let us start by introducing the notion of prenex normal form in the language of **CPL**:

Definition 18 *(PNF).* A formula of **CPL** is an *n-ary prenex normal form* (or simply a *prenex normal form*, PNF for short) if it can be written as $\triangle_1 \dots \triangle_n A$, where, for every $i \in \{1, \dots, n\}$, $\triangle_i$ is either $\mathbf{C}^q_a$ or $\mathbf{D}^q_a$ (for arbitrary a and q), and A is quantifier-free.

To convert a formula of **CPL** into an equivalent PNF, some intermediate lemmas are needed. As for **QPL**, conversion into PNF of **CPL**-formulas can have high complexity. Preliminarily, notice that for every **CPL**-formula A, name a, and finite set X such that $\mathrm{FN}(A) \subseteq X$ and $a \notin X$, if $q = 0$, then $[\![\mathbf{C}^q_a A]\!]_X = (2^{\mathbb{N}})^X$ and $[\![\mathbf{D}^q_a A]\!]_X = \emptyset$.

We will show that the connectives occurring *outside* the scope of a counting quantifier can be permuted with it. The lemma below considers the case of conjunction and disjunction.

Lemma 8. *For all $q > 0$, the following equivalences hold:*

$$A \wedge \mathbf{C}^q_a B \equiv \mathbf{C}^q_a (A \wedge B) \qquad A \vee \mathbf{C}^q_a B \equiv \mathbf{C}^q_a (A \vee B)$$
$$A \wedge \mathbf{D}^q_a B \equiv \mathbf{D}^q_a (\neg A \vee B) \qquad A \vee \mathbf{D}^q_a B \equiv \mathbf{D}^q_a (\neg A \wedge B).$$

To prove Lemma 8 we need a few preliminary results. Let us first define the following operation on sets:

Definition 19. Let X, Y be two disjoint sets of names. For any $\mathcal{X} \subseteq (2^{\mathbb{N}})^X$, the set $X^{\Uparrow Y} \subseteq (2^{\mathbb{N}})^{X \cup Y}$ is defined by

$$\mathcal{X}^{\Uparrow Y} = \{f + g \in (2^{\mathbb{N}})^{X \cup Y} \mid f \in \mathcal{X}, g \in (2^{\mathbb{N}})^Y\}.$$

The following useful properties are easily checked:

Lemma 9.

i. $\Pi_f(\mathcal{X} \cap \mathcal{Y}) = \Pi_f(\mathcal{X}) \cap \Pi_f(\mathcal{Y})$;
ii. $\Pi_f(\mathcal{X} \cup \mathcal{Y}) = \Pi_f(\mathcal{X}) \cup \Pi_f(\mathcal{Y})$;
iii. $\Pi_f(\overline{\mathcal{X}}) = \overline{\Pi_f(\mathcal{X})}$.

We will need two more technical lemmas.

Lemma 10. *Let X, Y be disjoint sets,* $\mathrm{FN}(A) \subseteq X$ *and* $f \in (2^{\mathbb{N}})^X$. *Then:*

i. if $f \in [\![A]\!]_X$, then $\Pi_f\big([\![A]\!]_{X \cup Y}\big) = (2^{\mathbb{N}})^Y$;
ii. if $f \notin [\![A]\!]_X$, then $\Pi_f\big([\![A]\!]_{X \cup Y}\big) = \emptyset$.

Equivalently, $[\![A]\!]_{X \cup Y} = ([\![A]\!]_X)^{\Uparrow Y}$.

Proof. We argue by induction on A. The base case is easily checked, and all propositional cases can be handled using the induction hypothesis together with the suitable clause from Lemma 9. Suppose now $A = \mathbf{C}^q_a B$. If $f \in [\![A]\!]_X$, then $\mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q$; let $g \in (2^{\mathbb{N}})^Y$; by induction hypothesis $[\![B]\!]_{X\cup Y\cup\{a\}} = ([\![B]\!]_{X\cup\{a\}})^{\Uparrow Y}$, and we have that $\Pi_{f+g}([\![B]\!]_{X\cup\{a\}\cup Y}) = \{h \mid f+h+g \in [\![B]\!]_{X\cup\{a\}\cup Y}\} = \{h \mid f+h+g \in ([\![B]\!]_{X\cup\{a\}})^{\Uparrow Y}\} = \{h \mid f+h \in [\![B]\!]_{X\cup\{a\}}\} = \Pi_f([\![B]\!]_{X\cup\{a\}})$. We deduce then $\mu(\Pi_{f+g}([\![B]\!]_{X\cup Y\cup\{a\}})) \geq q$, and thus $f+g \in [\![A]\!]_{X\cup Y}$. Since g was chosen arbitrary, this shows that $\Pi_f([\![A]\!]_{X\cup Y}) = (2^{\mathbb{N}})^Y$. If $f \notin [\![A]\!]_X$, then $\mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) < q$, and by a similar argument we deduce that, for all $f \in (2^{\mathbb{N}})^Y$, $\mu(\Pi_{f+g}([\![B]\!]_{X\cup Y\cup\{a\}})) < q$ holds, whence $\Pi_f([\![A]\!]_{X\cup Y}) = \emptyset$. The case of $A = \mathbf{D}^q_a B$ can be treated in a similar way. □

Lemma 11. *Assume $a \notin X$, $\mathrm{FN}(A) \subseteq X$, $f \in (2^{\mathbb{N}})^X$, and $q > 0$. Then,*

$$\mu(\Pi_f([\![A]\!]_{X\cup\{a\}})) \geq q \text{ iff } f \in [\![A]\!]_X$$

$$\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q \text{ iff } f \in [\![A]\!]_X \wedge \mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q$$

$$\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cup \Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q \text{ iff } f \in [\![A]\!]_X \vee \mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q$$

$$\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cup \Pi_f([\![B]\!]_{X\cup\{a\}})) < q \text{ iff } f \notin [\![A]\!]_X \wedge \mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) < q$$

$$\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) < q \text{ iff } f \notin [\![A]\!]_X \vee \mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) < q.$$

Proof sketch. Let us consider the first two cases only. For the first one, we have:

- $\Rightarrow$ The proof is by contraposition. Assume $f \notin [\![A]\!]_X$. Since $a \notin X$ and $f \in (2^{\mathbb{N}})^X$ by hypothesis, we apply Lemma 10.ii, obtaining $\Pi_f([\![A]\!]_{X\cup\{a\}}) = \emptyset$. Then, $\mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) = \mu(\emptyset) = 0 < q$.
- $\Leftarrow$ Assume $f \in [\![A]\!]_X$. Since $a \notin X$ and $f \in (2^{\mathbb{N}})^X$ by hypothesis, we apply Lemma 10.i, obtaining $\Pi_f([\![A]\!]_{X\cup\{a\}}) = (2^{\mathbb{N}})^{\{a\}}$. Then, $\mu(\Pi_f([\![A]\!]_{X\cup\{a\}})) = \mu((2^{\mathbb{N}})^{\{a\}}) = 1 \geq q$.

For the second one, preliminarily notice that:

- If $f \in [\![A]\!]_X$, then, since $a \notin X$ and $f \in (2^{\mathbb{N}})^X$, by Lemma 10.i, $\Pi_f([\![A]\!]_{X\cup\{a\}}) = (2^{\mathbb{N}})^{\{a\}}$. So $\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) = \mu((2^{\mathbb{N}})^{\{a\}} \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) = \mu(\Pi_f([\![B]\!]_{X\cup\{a\}}))$.
- If $f \notin [\![A]\!]_X$, then, since $a \notin X$ and $f \in (2^{\mathbb{N}})^X$, by Lemma 10.ii, $\Pi_f([\![A]\!]_{X\cup\{a\}}) = \emptyset$. So $\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) = \mu(\emptyset \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) = \mu(\emptyset) = 0$.

Then, we conclude the proof as follows:

- $\Leftarrow$ By hypothesis $f \in [\![A]\!]_X$ and $\mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q$. Then, by the first clause above together with the first hypothesis, $\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) = \mu(\Pi_f([\![B]\!]_{X\cup\{a\}}))$ and clearly, by the second hypothesis, $\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q$.
- $\Rightarrow$ The proof is by contraposition. If $f \notin [\![A]\!]_X$, then, by the second clause above, $\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) = 0$. So, trivially, for any $q \in \mathbb{Q}_{[0,1]}$, we conclude $\mu(\Pi_f([\![A]\!]_X) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) < q$. Otherwise, $f \in [\![A]\!]_X$ and $\mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) < q$. Observe that for the first clause above, $\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) = \mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) < q$. We can then conclude $\mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) < q$. □

We can now prove Lemma 8.

Proof of Lemma 8. We only illustrate the case of $A \wedge \mathbf{C}^q_a B \equiv \mathbf{C}^q_a (A \wedge B)$. We will show that for any set of names X such that $a \notin X$, $\mathrm{FN}(A) \subseteq X$ and $\mathrm{FN}(B) \subseteq X \cup \{a\}$, $[\![A \wedge \mathbf{C}^q_a B]\!]_X = [\![\mathbf{C}^q_a (A \wedge B)]\!]_X$. Using Lemma 9 i. as well as the second clause from Lemma 11, we have that, for all $f \in (2^{\mathbb{N}})^X$,

$$\begin{array}{lll} f \in [\![\mathbf{C}^q_a (A \wedge B)]\!]_X & \text{iff} & \mu(\Pi_f([\![A]\!]_{X\cup\{a\}} \cap [\![B]\!]_{X\cup\{a\}})) \geq q \\ & \text{iff} & \mu(\Pi_f([\![A]\!]_{X\cup\{a\}}) \cap \Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q \\ & \text{iff} & f \in [\![A]\!]_X \text{ and } \mu(\Pi_f([\![B]\!]_{X\cup\{a\}})) \geq q \\ & \text{iff} & f \in [\![A \wedge \mathbf{C}^q_a B]\!]_X. \end{array}$$

All other cases can be proved in a similar way, exploiting the appropriate clauses from Lemma 9 and Lemma 11. □

Remarkably, a corresponding lemma does *not* hold for $\mathbf{CPL}_0$, due to the impossibility of *renaming* variables (on which Lemma 8 relies).

The lemma below considers the case of negation:

Lemma 12. *For every $q \in \mathbb{Q}_{[0,1]}$,*

$$\neg\mathbf{D}^q_a A \equiv \mathbf{C}^q_a A \qquad \neg\mathbf{C}^q_a A \equiv \mathbf{D}^q_a A.$$

Using Lemma 8 and Lemma 12, we can conclude that every formula of **CPL** can be put in PNF, as desired.

Proposition 6. *For every formula A of **CPL** there is a PNF B, such that $A \equiv B$ holds. Moreover, B can be computed in polynomial time from A.*

6.2. Positive prenex normal forms

Reducing formulas to PNF is close to what we need, but there is one last step to make, namely getting rid of the quantifier **D**, which does not have any counterpart in Wagner's construction. In other words, we need to reduce **CPL**-formulas to prenex normal forms *of a special kind*:

Definition 20 *(PPNF).* A formula of **CPL** is said to be a *positive prenex normal form* (PPNF, for short) when it is both PNF and **D**-free.

The gist to convert formulas into (equivalent) PPNF, consists in two main steps: (i) converting each instance of **D** into one of **C**, using Lemma 12, and (ii) applying the lemma below which states that **C** enjoys a specific, weak form of self duality, to push the negation inside the matrix. In order to prove (ii) we need to introduce some auxiliary definition and to establish some results, in particular the so-called Epsilon Lemma, which in its turn relies on some preliminary lemmas.

First of all, for all $k \in \mathbb{N}$, let $[0,1]_k$ indicate the set of dyadic rationals of the form $q = \sum_{i=1}^k b_i \cdot 2^{-i}$, where $b_i \in \{0,1\}$. Notice that, for all $p \leq 2^k$, $\frac{p}{2^k} \in [0,1]_k$.

Lemma 13. *For any Boolean formula $\mathscr{b}$ with $\mathrm{FN}(\mathscr{b}) \subseteq \{a\}$, $\mu\big(\llbracket \mathscr{b} \rrbracket\big) \in [0,1]_k$, where k is the maximum natural number such that x^a_{k-1} occurs in $\mathscr{b}$.*

Proof. Let $p \leq 2^k$ indicate the number of valuations $m : \{x^a_0, \dots, x^a_{k-1}\} \to \{0,1\}$ that make $\mathscr{b}$ true. Then, by Lemma 1, $\mu(\llbracket \mathscr{b} \rrbracket) = p \cdot 2^{-k} \in [0,1]_k$. $\square$

Lemma 14. *For all $\mathcal{X} \in \mathscr{B}\big((2^{\mathbb{N}})^X\big)$ and $r \in [0,1]$, $\mu(\mathcal{X}) \leq r$ holds iff $\mu(\overline{\mathcal{X}}) \geq 1 - r$.*

Proof. The claim follows from $1 = \mu\big((2^{\mathbb{N}})^X\big) = \mu(\mathcal{X} \cup \overline{\mathcal{X}}) = \mu(\mathcal{X}) + \mu(\overline{\mathcal{X}})$. $\square$

We now have all ingredients to prove the Epsilon Lemma:

Lemma 15 *(Epsilon Lemma). For every formula A of **CPL**, $a \in \mathrm{FN}(A)$ and $q \in \mathbb{Q}_{[0,1]}$, there is $\epsilon \in \mathbb{Q}_{[0,1]}$ such that $\neg\mathbf{C}^q_a A \equiv \mathbf{C}^{1-(q+\epsilon)}_a \neg A$. Moreover, ϵ can be computed from q in polynomial time.*

Proof. Let $\mathscr{b}_A$ (Remark 1) be a-decomposable as $\bigvee_i^n \mathscr{d}_i \wedge \mathscr{e}_i$ and let k be maximum such that x^a_k occurs in $\mathscr{b}_A$. By Lemma 13, for all $i = 0, \dots, n$, $\mu(\llbracket \mathscr{d}_i \rrbracket_{\{a\}}) \in [0,1]_k$. This implies in particular that for every $f : X \to 2^{\mathbb{N}}$, $\mu\big(\Pi_f\big(\llbracket A \rrbracket_{X \cup \{a\}}\big)\big) \in [0,1]_k$, since $\Pi_f\big(\llbracket A \rrbracket_{X\cup\{a\}}\big)$ coincides with the unique $\llbracket \mathscr{d}_i \rrbracket_{\{a\}}$ such that $f \in \llbracket \mathscr{e}_i \rrbracket_X$, by Lemma 7. Now, if $q \notin [0,1]_k$, let $\epsilon = 0$ and if $q \in [0,1]_k$, then: if $q = 1$, let $\epsilon = -2^{-(k+1)}$, and if $q \neq 1$, let $\epsilon = 2^{-(k+1)}$. In all cases $q + \epsilon \notin [0,1]_k$. So, for all $X \supseteq \mathrm{FN}(A) - \{a\}$, we deduce:

$$\begin{aligned}
\llbracket \neg\mathbf{C}^q_a A \rrbracket_X &= \{f : X \to 2^{\mathbb{N}} \mid \mu\big(\Pi_f\big(\llbracket A \rrbracket_{X\cup\{a\}}\big)\big) < q\} \\
&= \{f : X \to 2^{\mathbb{N}} \mid \mu\big(\Pi_f\big(\llbracket A \rrbracket_{X\cup\{a\}}\big)\big) \leq q + \epsilon\} \\
&\overset{L14}{=} \{f : X \to 2^{\mathbb{N}} \mid \mu\big(\overline{\Pi_f\big(\llbracket A \rrbracket_{X\cup\{a\}}\big)}\big) \geq 1 - (q+\epsilon)\} \\
&\overset{L9}{=} \{f : X \to 2^{\mathbb{N}} \mid \mu\big(\Pi_f\big(\llbracket \neg A \rrbracket_{X\cup\{a\}}\big)\big) \geq 1 - (q+\epsilon)\} \\
&= \llbracket \mathbf{C}^{1-(q+\epsilon)}_a \neg A \rrbracket_X. \quad \square
\end{aligned}$$

From Lemma 15 it follows that any PNF can be transformed into a PPNF, so we deduce the following:

Proposition 7. *For every formula A of **CPL** there is a PPNF B such that $A \equiv B$ holds. Moreover, B can be computed from A in polynomial time.*

6.3. **CPL** and the counting hierarchy

We now have all ingredients to establish our main result, that is, the correspondence between **CPL** and the counting hierarchy. The hierarchy CH, introduced in [56], consists in the sequence of complexity classes CH_n defined by

$$\mathsf{CH}_0 = \mathsf{P} \qquad \mathsf{CH}_{n+1} = \mathsf{PP}^{\mathsf{CH}_n}.$$

It is well-known that `MajSat`, which asks to decide whether a Boolean formula is satisfied by *the majority* of its models, is a complete problem for $\mathsf{PP} = \mathsf{CH}_1$. Moreover, the already mentioned problem `MajMajSat` is complete for $\mathsf{PP}^{\mathsf{PP}} = \mathsf{CH}_2$. More generally, in the same work Wagner defined complete problems for *each level* of CH, which can be seen as generalizations of the two problems just mentioned. Below, we present a slightly weaker version of Wagner's Theorem [56, pp. 338-339], which perfectly fits our needs.

Let S be a set and suppose $\mathcal{L}$ is any subset of S^n, with $1 \leq m < n$, and let $b \in \mathbb{N}$. We define $\mathbf{C}^b_m\mathcal{L}$ as the following subset of S^{n-m}:

$$\mathbf{C}^b_m\mathcal{L} = \{(a_n, \ldots, a_{m+1}) \mid \#(\{(a_m, \ldots, a_1) \mid (a_n, \ldots, a_1) \in \mathcal{L}\}) \geq b\}.$$

For any natural number $n \in \mathbb{N}$, let $\mathcal{TF}^n$ be the set of all tuples of the form $(t_1, \ldots, t_n, A)$, where A is a propositional formula in CNF with at most n free variables $x_1, \ldots, x_n$, and $t_1, \ldots, t_n \in \{\mathbf{T}, \mathbf{F}\}$ render A true. Finally, for every $k \in \mathbb{N}$, let W^k be the language consisting of all (binary encodings of) tuples of the form $(m_1, \ldots, m_k, b_1, \ldots, b_k, A)$ such that $A \in \mathbf{C}^{b_1}_{m_1} \cdots \mathbf{C}^{b_k}_{m_k} \mathcal{TF}^{\sum m_i}$.

Theorem 1 *([56], Th. 7). For every k, the language W^k is complete for CH_k.*

Observe that elements of W^k can be seen as alternative representations for PPNF formulas of **CPL**, once any m_i is replaced by $\min\{1, \frac{m_i}{2^{b_i}}\}$. As a consequence, we finally obtain the following claim, which shows, as desired, that the formulas of **CPL** in PPNF provide complete problems for all levels of the counting hierarchy.

Corollary 2. *The closed and valid k-ary PPNFs, whose matrix is in CNF, define a complete set for CH_k.*

7. Multivariate counting logic: proof theory

In this section we define a sound and complete proof system for **CPL**. As most arguments are straightforward variants of those developed in Section 7 for $\mathbf{CPL}_0$, we omit most proofs. We introduce *named* variants of sequents and external hypotheses.

Definition 21 *(Named external hypothesis).* A *named external hypothesis* is an expression of one of the following forms:

- $a \in X$,
- $\mu(\llbracket \mathscr{b} \rrbracket) = 0$ or $\mu(\llbracket \mathscr{b} \rrbracket) = 1$,
- $\mathscr{b} \vDash^X \mathscr{c}$.

Moreover, with a slight abuse of notation, given an a-decomposition formula $\bigvee_i \mathscr{c}_i \wedge \mathscr{d}_i$ and $q \in \mathbb{Q}_{[0,1]}$, we indicate as $\bigvee_i \{\mathscr{c}_i \mid \mu(\llbracket \mathscr{d}_i \rrbracket) \triangleright q\}$ (with $\triangleright \in \{\geq, \leq, >, <, =\}$) the Boolean formula $\mathscr{c}_{i_1} \vee \cdots \vee \mathscr{c}_{i_n}$, where $\{i_1, \ldots, i_n\}$ is the set of all indexes i such that the condition $\mu(\llbracket \mathscr{d}_i \rrbracket) \triangleright q$ holds.

Analogously to the case of $\mathbf{CLK}_0$, a labelled formula is an expression of one of the forms $\mathscr{b} \rightarrowtail A$ or $\mathscr{b} \leftarrowtail A$, where $\mathscr{b}$ is a named Boolean formula and A is a **CPL**-formula. A labelled formula $\mathscr{b} \rightarrowtail A$ (resp. $\mathscr{b} \leftarrowtail A$) is *valid* when, letting $X = \mathrm{FN}(\mathscr{b}) \cup \mathrm{FN}(A)$, $\llbracket \mathscr{b} \rrbracket_X \subseteq \llbracket A \rrbracket_X$ (resp. $\llbracket \mathscr{b} \rrbracket_X \supseteq \llbracket A \rrbracket_X$). A labelled sequent is an expression of the form $\vdash L$, where L is a labelled formula. $\vdash L$ is *valid* (noted $\vDash L$) when L is.

We define a one-sided, single-succedent and labelled sequent calculus, called **CLK**, in analogy with the system $\mathbf{CLK}_0$. Most rules are straightforward extensions of those of $\mathbf{CPL}_0$. The counting rules rely on the Fundamental Lemma 7. The rules of **CLK** are displayed in Fig. 4 As before, let us call $\mathsf{R}^{\leftarrowtail}_{\mu}$ and $\mathsf{R}^{\rightarrowtail}_{\mu}$ μ-rules. The notion of derivation height is defined as for $\mathbf{CPL}_0$.

The proofs of soundness and completeness for **CLK** are structurally very similar to those for $\mathbf{CLK}_0$. For this reason, we just sum up their structure and briefly explain a few discrepancies with $\mathbf{CPL}_0$.

Soundness is established as for $\mathbf{CLK}_0$, by standard induction on the height of derivations. Notice that, in this case, the argument for the μ-rules relies on Lemma 7.

Initial Sequents

$$\dfrac{\mathscr{b} \vDash x_i^a}{\vdash \mathscr{b} \rightarrowtail \mathbf{i}_a}\ \mathsf{Ax1} \qquad\qquad \dfrac{x_i^a \vDash \mathscr{b}}{\vdash \mathscr{b} \leftarrowtail \mathbf{i}_a}\ \mathsf{Ax2}$$

Union Rule

$$\dfrac{\vdash \mathscr{c} \rightarrowtail A \qquad \vdash \mathscr{d} \rightarrowtail A \qquad \mathscr{b} \vDash \mathscr{c} \vee \mathscr{d}}{\vdash \mathscr{b} \rightarrowtail A}\ \mathsf{R}_{\cup}^{\rightarrowtail}$$

Intersection Rule

$$\dfrac{\vdash \mathscr{c} \leftarrowtail A \qquad \vdash \mathscr{d} \leftarrowtail A \qquad \mathscr{c} \wedge \mathscr{d} \vDash \mathscr{b}}{\vdash \mathscr{b} \leftarrowtail A}\ \mathsf{R}_{\cap}^{\leftarrowtail}$$

Logical Rules

$$\dfrac{\vdash \mathscr{c} \leftarrowtail A \qquad \mathscr{b} \vDash \neg \mathscr{c}}{\vdash \mathscr{b} \rightarrowtail \neg A}\ \mathsf{R}_{\neg}^{\rightarrowtail} \qquad\qquad \dfrac{\vdash \mathscr{c} \rightarrowtail A \qquad \neg \mathscr{c} \vDash \mathscr{b}}{\vdash \mathscr{b} \leftarrowtail \neg A}\ \mathsf{R}_{\neg}^{\leftarrowtail}$$

$$\dfrac{\vdash \mathscr{b} \rightarrowtail A}{\vdash \mathscr{b} \rightarrowtail A \vee B}\ \mathsf{R1}_{\vee}^{\rightarrowtail} \qquad\qquad \dfrac{\vdash \mathscr{b} \rightarrowtail B}{\vdash \mathscr{b} \rightarrowtail A \vee B}\ \mathsf{R2}_{\vee}^{\rightarrowtail}$$

$$\dfrac{\vdash \mathscr{b} \leftarrowtail A \qquad \vdash \mathscr{b} \leftarrowtail B}{\vdash \mathscr{b} \leftarrowtail A \vee B}\ \mathsf{R}_{\vee}^{\leftarrowtail} \qquad\qquad \dfrac{\vdash \mathscr{b} \rightarrowtail A \qquad \vdash \mathscr{b} \rightarrowtail B}{\vdash \mathscr{b} \rightarrowtail A \wedge B}\ \mathsf{R}_{\wedge}^{\rightarrowtail}$$

$$\dfrac{\vdash \mathscr{b} \leftarrowtail A}{\vdash \mathscr{b} \leftarrowtail A \wedge B}\ \mathsf{R1}_{\wedge}^{\leftarrowtail} \qquad\qquad \dfrac{\vdash \mathscr{b} \leftarrowtail B}{\vdash \mathscr{b} \leftarrowtail A \wedge B}\ \mathsf{R2}_{\wedge}^{\leftarrowtail}$$

Counting Rules

$$\dfrac{\mu(\llbracket \mathscr{b} \rrbracket) = 0}{\vdash \mathscr{b} \rightarrowtail A}\ \mathsf{R}_{\mu}^{\rightarrowtail} \qquad\qquad \dfrac{\mu(\llbracket \mathscr{b} \rrbracket) = 1}{\vdash \mathscr{b} \leftarrowtail A}\ \mathsf{R}_{\mu}^{\leftarrowtail}$$

$$\dfrac{\vdash \mathscr{c} \rightarrowtail A \qquad \mathscr{b} \vDash \bigvee_i \{\mathscr{e}_i \mid \mu(\llbracket \mathscr{d}_i \rrbracket) \geq q\}}{\vdash \mathscr{b} \rightarrowtail \mathbf{C}_a^q A}\ \mathsf{R}_{\mathbf{C}}^{\rightarrowtail}$$

$$\dfrac{\vdash \mathscr{c} \leftarrowtail A \qquad \bigvee_i \{\mathscr{e}_i \mid \mu(\llbracket \mathscr{d}_i \rrbracket) \geq q\} \vDash \mathscr{b}}{\vdash \mathscr{b} \leftarrowtail \mathbf{C}_a^q A}\ \mathsf{R}_{\mathbf{C}}^{\leftarrowtail}$$

$$\dfrac{\vdash \mathscr{c} \leftarrowtail A \qquad \bigvee_i \{\mathscr{e}_i \mid \mu(\llbracket \mathscr{d}_i \rrbracket) \geq q\} \vDash \neg \mathscr{b}}{\vdash \mathscr{b} \rightarrowtail \mathbf{D}_a^q A}\ \mathsf{R}_{\mathbf{D}}^{\rightarrowtail}$$

$$\dfrac{\vdash \mathscr{c} \rightarrowtail A \qquad \neg \mathscr{b} \vDash \bigvee_i \{\mathscr{e}_i \mid \mu(\llbracket \mathscr{d}_i \rrbracket) \geq q\}}{\vdash \mathscr{b} \leftarrowtail \mathbf{D}_a^q A}\ \mathsf{R}_{\mathbf{D}}^{\leftarrowtail}$$

where $\bigvee_i \mathscr{e}_i \wedge \mathscr{d}_i$ is an a-decomposition of $\mathscr{c}$.

Fig. 4. Proof System for **CLK**.

Proposition 8 *(Soundness of* **CLK***). If* $\vdash_{\mathbf{CLK}} L$ *then* $\vDash L$.

Proof sketch. We show that, if $\vdash_{\mathbf{CLK}} \mathscr{b} \rightarrowtail A$ (resp. $\vdash_{\mathbf{CLK}} \mathscr{b} \leftarrowtail A$) holds, then for any $X \supseteq \mathrm{FN}(\mathscr{b}) \cup \mathrm{FN}(A)$, $\llbracket \mathscr{b} \rrbracket_X \subseteq \llbracket A \rrbracket_X$ (resp. $\llbracket A \rrbracket_X \subseteq \llbracket \mathscr{b} \rrbracket_X$). The proof is by induction on the height of the derivation of $\vdash L$.

- *Base case.* The sequent is either initial or derived by a μ-rule. Let us just consider Ax1 as an example. Assume that the derivation is as follows:

$$\dfrac{\mathscr{b} \vDash x_i^a}{\vdash \mathscr{b} \rightarrowtail \mathbf{i}_a}\ \mathsf{Ax1}$$

  Let $X \supseteq \mathrm{FN}(\mathscr{b}) \cup \{a\}$. From $\mathscr{b} \vDash x_i^a$, we deduce that $\llbracket \mathscr{b} \rrbracket_X \subseteq \llbracket x_i^a \rrbracket_X = \{f \in (2^{\mathbb{N}})^X \mid f(a)(i) = 1\} = \llbracket \mathbf{i}_a \rrbracket_X$. Hence $\vdash \mathscr{b} \rightarrowtail \mathbf{i}_a$ is valid.
- *Inductive case.* Let us consider the counting rule $\mathsf{R}_{\mathbf{C}}^{\rightarrowtail}$ as an example. Assume that $\bigvee_i \mathscr{e}_i \wedge \mathscr{d}_i$ is an a-decomposition of $\mathscr{c}$ and that the derivation is as follows:

$$\dfrac{\dfrac{\vdots}{\vdash \mathscr{c} \rightarrowtail A} \qquad \mathscr{b} \vDash \bigvee_i \{\mathscr{e}_i \mid \mu([\![\mathscr{d}_i]\!]) \geq q\}}{\vdash \mathscr{b} \rightarrowtail \mathbf{C}^q_a A}\ \mathsf{R}^{\rightarrowtail}_{\mathbf{C}}$$

Let $X \supseteq \mathrm{FN}(\mathbf{C}^q_a A) \cup \mathrm{FN}(\mathscr{b})$ and let Y be such that $Y \cup \{a\} \supseteq X \cup \mathrm{FN}(\mathscr{c})$. We can suppose w.l.o.g. that $a \notin X, Y$. From $\mathscr{b} \vDash \bigvee_i \{\mathscr{e}_i \mid \mu([\![\mathscr{d}_i]\!]) \geq q\}$, via Lemma 7, we deduce that $[\![\mathscr{b}]\!]_Y \subseteq \{f \in (2^{\mathbb{N}})^Y \mid \mu\big(\Pi_f([\![\mathscr{c}]\!]_{Y\cup\{a\}})\big) \geq q\}$. From $\vdash \mathscr{c} \rightarrowtail A$, by IH, we furthermore deduce $[\![\mathscr{c}]\!]_{Y\cup\{a\}} \subseteq [\![A]\!]_{Y\cup\{a\}}$, which implies $[\![\mathscr{b}]\!]_Y \subseteq \{f \in (2^{\mathbb{N}})^Y \mid \mu\big(\Pi_f([\![A]\!]_{Y\cup\{a\}})\big) \geq q\}$. Now, using Lemma 10, we have that, letting $Y = X \cup X'$, with X, X' disjoint, $[\![\mathscr{b}]\!]_Y = ([\![\mathscr{b}]\!]_X)^{\Uparrow X'}$ and $[\![A]\!]_{Y\cup\{a\}} = ([\![A]\!]_{X\cup\{a\}})^{\Uparrow X'}$; using these facts, we can deduce that $[\![\mathscr{b}]\!]_X \subseteq \{f \in (2^{\mathbb{N}})^X \mid \mu\big(\Pi_f([\![A]\!]_{X\cup\{a\}})\big) \geq q\}$, that is $[\![\mathscr{b}]\!]_X \subseteq [\![\mathbf{C}^q_a A]\!]_X$, as desired. □

Again, the completeness argument is similar to the one given for $\mathbf{CLK}_0$. Also in this case, it is based on a decomposition relation $\rightsquigarrow$ between sets of sequents. Just to give an idea of how $\rightsquigarrow_0$ is defined, we show a few examples of reduction:

$$\begin{aligned}
\text{if } \mathscr{b} \vDash \neg\mathscr{c},\ \vdash \mathscr{b} \rightarrowtail \neg A \ &\rightsquigarrow_0\ \{\vdash \mathscr{c} \leftarrowtail A\}\\
\text{if } \mathscr{b} \vDash \mathscr{c} \vee \mathscr{d},\ \vdash \mathscr{b} \rightarrowtail A \vee B \ &\rightsquigarrow_0\ \{\vdash \mathscr{c} \rightarrowtail A, \vdash \mathscr{d} \rightarrowtail B\}\\
\text{if } \mathscr{b} \vDash \bigvee_i \{\mathscr{e}_i \mid \mu([\![\mathscr{d}_i]\!]_{\{a\}}) \geq q\},\ \vdash \mathscr{b} \rightarrowtail \mathbf{C}^q_a A \ &\rightsquigarrow_0\ \{\vdash^{X\cup\{a\}} \mathscr{c} \rightarrowtail A\}.
\end{aligned}$$

where in the last case it is assumed that $\mathscr{c} = \bigvee_i \mathscr{e}_i \wedge \mathscr{d}_i$ is an a-decomposition of $\mathscr{c}$.

Again, as for $\mathbf{CLK}_0$, $\rightsquigarrow$ is the natural lifting of $\rightsquigarrow_0$ to a relation between sets of sequents and predicates about sequents can be generalized to predicates about sets. "Multivariate" $\rightsquigarrow$ is still strongly normalizing, as can be easily established by exhaustive analysis of all possible (named) $\rightsquigarrow_0$.

The notions of $\rightsquigarrow$-normal forms, normalization and basic sequents are straightforward generalizations to named sequents of the corresponding definitions, as presented in Section 4. Also the proofs of the properties below can be obtained as for the univariate language of $\mathbf{CLK}_0$, by simply switching to the context of named sequents and of the corresponding (named) $\rightsquigarrow_0$ (and using the Lemma 7 and Lemma 15).

Proposition 9.

i. *If a sequent is $\rightsquigarrow_0$-normal, then it is valid if and only if derivable in* **CLK**.
ii. *Each valid sequent of* **CLK** *has a valid $\rightsquigarrow$-normal form.*[1]
iii. *Given a set of (named) sequents Φ and Ψ, if $\Phi \rightsquigarrow \Psi$ and Ψ is derivable in* **CLK**, *then Φ is derivable in* **CLK**.

As in Section 4, completeness easily follows by combining these properties.

Proposition 10. *$\vDash L$ implies $\vdash_{\mathbf{CLK}} L$.*

8. Related works

Probability logics In the last decades, several probabilistic logical systems have been developed in the realm of modal logic, starting from the pioneering works by [39,40]. In particular, in the 1990s, some noteworthy probability logics were (independently) introduced both by [7,5,6] and by [17,22,16,23]. In particular, Bacchus defined probability terms in a way which is not too different from how we define formulas in $\mathbf{CPL}_0$ (even though he considers terms rather than formulas).

Another class of probabilistic modal logics have been designed to model Markov chains and similar structures, see for instance [24,30,32]. Some of these logics are probabilistic extensions of **CTL**, the standard logic for model-checking. Differently from **CPL**, in these systems, modal operators have a dynamical meaning, as they describe transitions in Markov decision processes.

While these approaches focus on semantics, several calculi for probabilistic logics have also been studied. On the one hand, complete axiomatic systems have been provided for the two probability logics mentioned above [7,17]. The probabilistic logic in [33], which admits a sound and complete axiomatic system, is somehow reminiscent of our system $\mathbf{CPL}_0$. Another sequent calculus is provided in [11] for a logic describing Carnap-Popper-type probability models. However, our calculi are actually inspired by labelled systems, e.g. **G3K*** and **G3P***, as presented for example in [38,21].

As we said, our notion of counting quantifiers was mostly inspired by Wagner's counting operator over languages, [54,56,55]. Other proof-theoretic approaches to probabilistic logics have been studied in connection with computational aspects.

[1] As for $\mathbf{CLK}_0$ the proof is by exhaustive inspection of all possible forms of $\rightsquigarrow_0$. Notice that, when dealing with named sequents, the proof for the counting cases relies on Lemma 7 and Lemma 15.

For instance, *Riesz modal logic* [18], which admits a sound and complete sequent calculus, was developed to describe properties of probabilistic Markov processes. [25] describes an extension of linear logic with two probabilistic connectives and a deep inference calculus, also with the goal of describing probabilistic processes. Finally, several probabilistic extensions of relational logics (like *probabilistic relational Hoare logic* [8]) have been developed for probabilistic verification (e.g. to formally prove properties like differential privacy or the security of cryptographic protocols).

Models of probabilistic computation Probabilistic models and randomized computation are pervasive in many areas of computer science and programming. From the 1950s on, the interest for probabilistic algorithms and models started spreading, see [31,15,13,43–45,48], and probabilistic computation has been intensively studied by the TCS community, see [43,46,31,47]. Nowadays, well-defined computational models, such as randomized variations on probabilistic automata, (both Markovian and oracle) Turing machines [46,47,19], and λ-calculi, are available.

Probabilistic complexity classes and the counting hierarchy The counting hierarchy was (independently) defined in the 1980s by [54–56] and, by [42]. It was conceived as an extension of [36,37]'s PH aiming at characterizing natural problems in which counting is involved. As proved in [52], there are two main, equivalent characterization of CH: the original characterization in terms of *alternating quantifiers*, [56], and the one based on *oracles*, [51].

Notably, Wagner's operator was not the only "probabilistic" (class) quantifier introduced in the 1980s. For example, [41] characterizes PPSPACE by alternating standard and *probabilistic* quantifiers, the latter ones expressing that more than the half of the strings of a certain length satisfy the underlying predicate. [59] characterizes BPP by means of a *random* quantifier. [58] also considered the relationship between classical and probabilistic classes introducing other quantifiers, such as the *overwhelming* and *majority* ones. However, to the best of the authors' knowledge, all these operators are counting quantifiers on (classes of) languages, rather than *stricto sensu* logical ones. One remarkable exception is represented by [29,28]'s work, in which second-order quantifiers are defined in the style of descriptive complexity.

9. Conclusion

To the best of our knowledge, **CPL** is the first logical system extending propositional logic with counting quantifiers. Our main source of inspiration comes from computational complexity, namely from Wagner's counting operator on classes. By the way, we believe that the main contribution of the paper is not the introduction of counting logics *per se*, but the investigation of its connections with counting classes. Indeed, we have shown that counting quantifiers play nicely with propositional logic in characterizing CH, and thus relate nicely with some old and recent results in complexity theory. In our opinion, **CPL** naturally appears as the probabilistic counterpart of **QPL**.

Due to space reasons, we left out some important applications of counting propositional logic to other branches of computer science, such as the theory of programming languages. In particular, it is possible to design type systems for the randomized λ-calculus by extending simple types with counting quantifiers, and to define a probabilistic counterpart of the *Curry-Howard correspondence* (see, e.g. [20,49]) relating typing derivations with derivations in an intuitionistic variant of **CLK**, see [3]. Moreover, the proof theory of **CPL** has just been briefly delineated and the dynamics (i.e. the cut-elimination procedure) of the introduced formal systems deserves further investigation.

Promising results also concern the possibility to inject counting quantifiers into the language of arithmetic. In particular, in [2] we have investigated an extension of standard Peano Arithmetics with *measure quantifiers*, which can be seen as a natural generalization of the quantifiers of $\mathbf{CPL}_0$ to the language of arithmetic. The extension of counting quantifiers to arithmetic looks particularly promising, as it suggests ways of characterizing in a somehow logical way explicit lower bounds for counting problems [35], as well as the possibility of defining new logical systems capturing probabilistic complexity classes like e.g. BPP (see for instance [26]).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] M. Antonelli, U. Dal Lago, P. Pistone, On counting propositional logic and Wagner's hierarchy, in: C. Sacerdoti Coen, I. Salvo (Eds.), Proceedings of the Italian Conference on Theoretical Computer Science 2021 (ICTCS '21), 2021, pp. 107–121.
[2] M. Antonelli, U. Dal Lago, P. Pistone, On measure quantifiers in first-order arithmetic, in: Proceedings of Computability in Europe 2021 (CiE2021), 2021, pp. 12–24.
[3] M. Antonelli, U. Dal Lago, P. Pistone, Curry and Howard meet Borel, in: Proceedings of the 37th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS '22), Association for Computing Machinery, New York, NY, USA, 2022, pp. 1–13.
[4] S. Arora, B. Barak, Computational Complexity: A Modern Approach, Cambridge University Press, 2009.
[5] F. Bacchus, Lp, a logic for representing and reasoning with statistical knowledge, Comput. Intell. 6 (1990) 209–231.
[6] F. Bacchus, On probability distributions over possible worlds, in: Machine Intelligence and Pattern Recognition, vol. 9, 1990, pp. 217–226.
[7] F. Bacchus, Representing and Reasoning with Probabilistic Knowledge, MIT Press, 1990.

[8] G. Barthe, B. Grégoire, S. Zanella Béguelin, Probabilistic relational Hoare logics for computer-aided security proofs, in: Proceedings of the 11th International Conference on Mathematics of Program Construction (MPC '12), Springer-Verlag, Berlin, Heidelberg, 2012, pp. 1–6.
[9] A. Biere, M. Heule, H. van Maaren, T. Walsh, Handbook of Satisfiability, IOS Press, 2009.
[10] P. Billingsley, Probability and Measure, Wiley, 1995.
[11] M. Boričić, Suppes-style sequent calculus for probability logic, J. Log. Comput. 27 (2015) 1157–1168.
[12] H. Büning, U. Bubeck, Theory of quantified Boolean formulas, in: A. Biere, M. Heule, H. van Maaren, T. Walsh (Eds.), Handbook of Satisfiability, IOS Press, 2009, pp. 735–760.
[13] J. Carlyle, Reduced forms for stochastic sequential machines, J. Math. Anal. Appl. 7 (1963) 167–174.
[14] S. Cook, The complexity of theorem-proving procedures, in: Proceedings of the Third Annual ACM Symposium on Theory of Computing (STOC '71), Association for Computing Machinery, 1971, pp. 151–158.
[15] A. Davis, Markov chains as random input automata, Am. Math. Mon. 68 (1961) 264–267.
[16] R. Fagin, J. Halpern, Reasoning about knowledge and probability, J. ACM 41 (1994) 340–367.
[17] R. Fagin, J. Halpern, N. Megiddo, A logic for reasoning about probabilities, Inf. Comput. 87 (1990) 78–128.
[18] R. Furber, R. Mardare, M. Mio, Probabilistic logics based on Riesz spaces, Log. Methods Comput. Sci. 16 (2020) 6.
[19] J. Gill, Computational complexity of probabilistic Turing machines, in: Proceedings of the Sixth Annual ACM Symposium on Theory of Computing (STOC '74), Association for Computing Machinery, 1974, pp. 91–95.
[20] J.Y. Girard, Proof and Types, Cambridge University Press, 1989.
[21] M. Girlando, S. Negri, G. Sbardolini, Uniform labelled calculi for conditional and counterfactual logics, in: Language, Information, and Computation (WoLLIC '19), Springer Berlin Heidelberg, 2019, pp. 248–263.
[22] J. Halpern, An analysis of first-order logics for probability, Artif. Intell. 46 (1990) 311–350.
[23] J. Halpern, Reasoning About Uncertainty, MIT Press, 2003.
[24] H. Hansson, B. Jonsson, A logic for reasoning about time and reliability, Form. Asp. Comput. 6 (1994) 512–535.
[25] R. Horne, The sub-additives: a proof theory for probabilistic choice extending linear logic, in: H. Geuvers (Ed.), 4th International Conference on Formal Structures for Computation and Deduction (FSCD '19), Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2019, pp. 23:1–23:16.
[26] E. Jerábek, Approximate counting in bounded arithmetic, J. Symb. Log. 72 (2007) 959–993.
[27] A. Kechris, Classical Descriptive Set Theory, Graduate Texts in Mathematics, vol. 156, Springer-Verlag, 1995.
[28] J. Kontinen, H. Niemistö, Extensions of MSO and the monadic counting hierarchy, Inf. Comput. 209 (2011) 1–19.
[29] J. Kontinen, A logical characterization of the counting hierarchy, ACM Trans. Comput. Log. 10 (2009) 1–21.
[30] D. Kozen, Semantics of probabilistic programs, J. Comput. Syst. Sci. 53 (1982) 165–198.
[31] K. de Leeuw, E. Moore, C. Shannon, N. Shapiro, Computability by probabilistic machines, in: C.E. Shannon, J. McCarthy (Eds.), Automata Studies, in: Annals of Mathematics Studies, vol. 34, P.U. Press, 1956, pp. 183–212.
[32] D. Lehmann, S. Shelah, Reasoning with time and chance, Inf. Control 53 (1982) 165–198.
[33] Z. Marković, Z. Ognjanović, M. Rašković, A probabilistic extension of intuitionistic logic, Math. Log. Q. 49 (2003) 415–424.
[34] D. van Melkebeek, A survey on lower bounds for satisfiability and related problems, Found. Trends Theor. Comput. Sci. 2 (2007) 197–303.
[35] D. van Melkebeek, T. Watson, A quantum time-space lower bound for the counting hierarchy, Available at: https://minds.wisconsin.edu/handle/1793/60568, 2007.
[36] A. Meyer, L. Stockmeyer, The equivalence problem for regular expressions with squaring requires exponential space, in: Proceedings of the 13th Annual Symposium on Switching and Automata Theory (SWAT '72), IEEE, 1972, pp. 125–129.
[37] A. Meyer, L. Stockmeyer, Word problems requiring exponential time (preliminary report), in: Proceedings of the Fifth Annual ACM Symposium on Theory of Computing (STOC '73), Association for Computing Machinery, New York, NY, USA, 1973, pp. 1–9.
[38] S. Negri, J. von Plato, Proof Analysis: A Contribution to Hilbert's Last Problem, Cambridge University Press, 2011.
[39] N. Nilsson, Probabilistic logic, Artif. Intell. 28 (1986) 71–87.
[40] N. Nilsson, Probabilistic logic revisited, Artif. Intell. 59 (1993) 39–42.
[41] C. Papadimitriou, Games against nature, J. Comput. Syst. Sci. 31 (1985) 288–301.
[42] I. Parberry, G. Schnitger, Parallel computation with threshold functions, J. Comput. Syst. Sci. 36 (1988) 278–302.
[43] M.O. Rabin, Probabilistic automata, Inf. Comput. 6 (1963) 230–245.
[44] E. Santos, Maximin automata, Inf. Control 13 (1968) 363–377.
[45] E. Santos, Maximin sequential-like machines and chains, Math. Syst. Theory 3 (4) (1969) 300–309.
[46] E. Santos, Probabilistic Turing machines and computability, Proc. Am. Math. Soc. 22 (1969) 704–710.
[47] E. Santos, Computability by probabilistic Turing machines, Trans. Am. Math. Soc. (1971) 159–165.
[48] J. Simon, On tape-bounded probabilistic Turing machine acceptors, Theor. Comput. Sci. 16 (1981) 75–91.
[49] M. Sorensen, P. Urzyczyn, Lectures on the Curry-Howard Isomorphism, vol. 149, Elsevier, 2006.
[50] L. Stockmeyer, The polynomial-time hierarchy, Theor. Comput. Sci. 3 (1977) 1–22.
[51] J. Torán, An oracle characterization of the counting hierarchy, in: Proceedings of the Third Annual Conference on Structure in Complexity Theory, 1988, pp. 213–223.
[52] J. Torán, Complexity classes defined by counting quantifiers, J. ACM 38 (1991) 753–774.
[53] L. Valiant, The complexity of computing the permanent, Theor. Comput. Sci. 8 (1979) 189–201.
[54] K. Wagner, Compact descriptions and the counting polynomial-time hierarchy, in: Frege Conference 1984: Proceedings of the International Conference Held at Schwerin, 1984, pp. 383–392.
[55] K. Wagner, Some observations on the connection between counting and recursion, Theor. Comput. Sci. 47 (1986) 131–147.
[56] K. Wagner, The complexity of combinatorial problems with succinct input representation, Acta Inform. 23 (1986) 325–356.
[57] C. Wrathall, Complete sets and the polynomial-time hierarchy, Theor. Comput. Sci. 3 (1976) 23–33.
[58] S. Zachos, Probabilistic quantifiers and games, J. Comput. Syst. Sci. 36 (1988) 433–451.
[59] S. Zachos, H. Heller, A decisive characterization of BPP, Inf. Control 69 (1–3) (1986) 125–135.