Annals of Pure and Applied Logic 176 (2025) 103538

Contents lists available at ScienceDirect

Annals of Pure and Applied Logic

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

Full Length Article

Strength and limitations of Sherali-Adams and Nullstellensatz proof systems ☆

Ilario Bonacina *, Maria Luisa Bonet

UPC Universitat Politecnica de Catalunya, Jordi Girona Salgado 31, Barcelona, 08034, Catalunya, Spain

ARTICLE INFO

Article history:
Received 18 November 2022
Received in revised form 31 October 2024
Accepted 24 November 2024
Available online 28 November 2024

MSC:
03F20
68T15
03B05
03B70

Keywords:
Bounded-depth Frege
Nullstellensatz
Sherali-Adams
Pigeonhole principle

ABSTRACT

We compare the strength of the algebraic proof systems Sherali-Adams (SA) and Nullstellensatz (NS) with Frege-style proof systems. Unlike bounded-depth Frege, SA has polynomial-size proofs of the pigeonhole principle (PHP). A natural question is whether adding PHP to bounded-depth Frege is enough to simulate SA. We show that SA, with unary integer coefficients, lies strictly between tree-like depth-1 Frege + PHP and tree-like Resolution. We introduce a *levelled* version of PHP (LPHP) and we show that SA with integer coefficients lies strictly between tree-like depth-1 Frege + LPHP and Resolution. Analogous results are shown for NS using the bijective (i.e. onto and functional) pigeonhole principle and a leveled version of it.

1. Introduction

This paper connects logic based proof systems with algebraic ones. While logic based proof systems work directly with propositional formulas, the algebraic ones work with polynomials, including polynomial translations of Boolean formulas.

For instance, in the *Nullstellensatz proof system* (NS) [5], a CNF formula is shown unsatisfiable by first translating it into a set of polynomial equations, and a proof of the unsatisfiability is a sum of multiples of those equations that, after simplifications, reduces to the trivial contradiction $1 = 0$ (Definition 2.5). NS with coefficients over $\mathbb{Z}_2$ was first studied in connection with a major (and yet open) problem in proof complexity: the problem of proving super-polynomial size lower bounds for bounded-depth Frege systems

☆ A preliminary version of this work appeared in the proceedings of LICS'22 [9].

* Corresponding author.
E-mail addresses: bonacina@cs.upc.edu (I. Bonacina), bonet@cs.upc.edu (M.L. Bonet).

https://doi.org/10.1016/j.apal.2024.103538

with parity gates ([13,6,4] among others). Moreover, lower bounds in NS can be lifted to lower bounds for stronger proof systems [30,27,28].

Sherali-Adams (SA) [32] is similar to NS but instead of equations we first produce polynomial inequalities and a proof of unsatisfiability is a sum of positive multiples of the inequalities together with sums of positive monomials. In this case the trivial contradiction is $-1 \geq 0$ (Definition 2.6). The interest in studying SA relies primarily on its connections to approximation algorithms for important **NP**-hard optimization problems, see for instance the survey [18].

Frege is the standard textbook logic proof system. Restricting the depth of the formulas in Frege, we obtain proof systems like Resolution or *bounded-depth Frege* (Frege_d). SA is known to simulate Resolution, and it is stronger, since SA can prove the pigeonhole principle efficiently, unlike Resolution or even bounded-depth Frege [29,24]. Hence, natural questions are the following.

"*Which axioms do we need to add to constant-depth Frege to simulate* SA *or* NS*?*"

"*What is the minimal depth of constant-depth Frege (plus the extra axiom) needed to simulate* SA *or* NS*?*"

The axioms we want to add should be "natural", in the sense that they should have some clear combinatorial meaning. For instance, constant-depth Frege with *counting* MOD_2 axioms simulates NS with coefficients over $\mathbb{Z}_2$ [21].

The *pigeonhole principle* (PHP, Definition 4.1) is a natural combinatorial principle, which informally says that $n+1$ pigeons cannot all fly to n holes without any two of them sharing a hole. The *bijective* pigeonhole principle, i.e. onto and functional, is denoted by ofPHP (Definition 4.1). In this work we use propositional encodings of these principles.

We use principles generalizing PHP and ofPHP. The *leveled pigeonhole principles* LPHP and ofLPHP (Definition 5.1) informally capture similar combinatorial principles, where the pigeons and the holes are organized into "layers".[1] The pigeons have some "mass" and the holes have some "capacity" depending on the level they belong. The intended mass and capacity of pigeons and holes at the kth level is 2^k. The mass of the ith pigeon is the same as the capacity of the ith hole, but there is an extra pigeon with positive mass. Each pigeon can fly once with the whole mass or twice with half mass. Each hole can accept either one pigeon filling the full capacity or two pigeons filling half capacity each. SA efficiently proves LPHP but the proof seems to require coefficients encoded in binary (Theorem 5.9).

In this article we answer the questions above for NS and SA with coefficients in $\mathbb{Z}$. A bit unexpectedly, their strength seems to depend on whether the coefficients of the polynomials are encoded in unary or binary. Unary NS and unary SA refer to having coefficients encoded in unary.

Before we answer the questions, let us mention that, informally, *bounded-depth Frege + principle* means that the principle is given as an extra tautology. Also, a *tree-like* proof system means that each Boolean formula can only be used once.

We visually summarize our results, although the formal statements of the cited theorems are slightly stronger than what is shown in the figures, since they also take into account the degree of the polynomials.

As we can see in Fig. 1, tree-like Frege_1 + LPHP is strictly stronger than SA and SA is strictly stronger than Resolution. On the other hand, tree-like Frege_1 + PHP is strictly stronger than unary SA and unary SA is strictly stronger than tree-like Resolution.

Prior to our work, it was not clear at all if SA was able to prove efficiently any combinatorial principle significantly different from PHP (in addition to what Resolution can prove). This work shows this is not the

[1] In [9], the preliminary version of his work, the *leveled* pigeonhole principle was called *weighted* pigeonhole principle, wPHP. We use the term *leveled* instead of *weighted*, to avoid confusion with the *weak* pigeonhole principle.

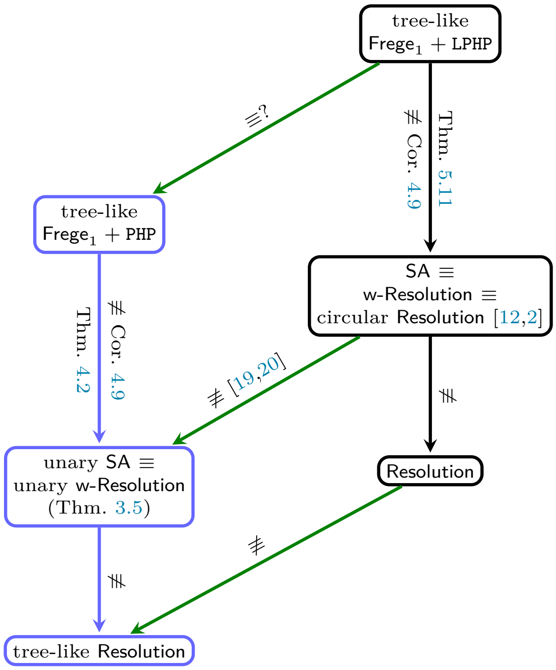

Fig. 1. The p-simulations for SA. The notation $P \rightarrow Q$ means that the proof system P p-simulates the proof system Q. The p-simulations are annotated with "≢" if the p-simulation is known to be strict, or with "≡?" whenever it is an open question if the p-simulation is strict or not. A green arrow → means the p-simulation is trivial. The color • is used to visually differentiate the results for the proof systems with unary weights/coefficients. See [19,20] for the separation between SA and unary SA.

case. At best, SA can prove just principles easily reducible to LPHP (in addition to what Resolution can prove).

Fig. 1 also states some equivalences between SA and unary SA and other proof systems based on Boolean formulas, in particular w-Resolution ([25,26,12], Definition 3.6) and circular Resolution [2].

Informally, *weighted* Resolution (w-Resolution) is a proof system where clauses have weights that can be positive or negative. The positive weight of a clause is the number of times we are allowed to use it as a premise of some inference, while the negative weight is the number of times we used it as an assumption and hence are required to justify it by deriving it. Clauses with positive weights might appear out of nothing as long as the same clauses appear also with negative weights. A proof starts with the initial clauses with some chosen positive weights and produces, using a small modification of the rules of Resolution, an empty clause with positive weight and all the clauses with negative weights have been justified.

As you can see in Fig. 2, tree-like Frege_1 + ofLPHP is stronger than NS. On the other hand, tree-like Frege_1 + ofPHP is stronger than unary NS and unary NS is strictly stronger than tree-like Resolution. We also show that NS and unary NS are p-equivalent to other proof systems based on Boolean formulas (Theorem 3.5).

The notion of weighted Resolution can be extended naturally to formulas of higher depth producing the system w-Frege_d (see Definition 3.6). As w-Resolution corresponds to SA, w-Frege_d can be thought as corresponding to generalization of SA handling algebraic expressions of higher depth. Fig. 3 shows the results we have for w-Frege_d. Basically the same results as in Fig. 1 but lifted from formulas of depth 0, i.e. clauses, to formulas of depth d. Tree-like Frege_{d+1} + LPHP is strictly stronger than w-Frege_d and w-Frege_d is strictly stronger than Frege_d. On the other hand, tree-like Frege_{d+1} + PHP is strictly stronger than unary w-Frege_d, and unary w-Frege_d is strictly stronger than tree-like Frege_d.

The PHP is the most studied principle in proof complexity and, for instance, we know that Frege_d + PHP is strictly weaker than Frege, at least for $d = o\left(\frac{\log\log n}{\log\log\log n}\right)$ [6], hence unary w-Frege_d is also *strictly* weaker than Frege for the same d (Corollary 4.7). To the best of our knowledge, the leveled pigeonhole principle LPHP

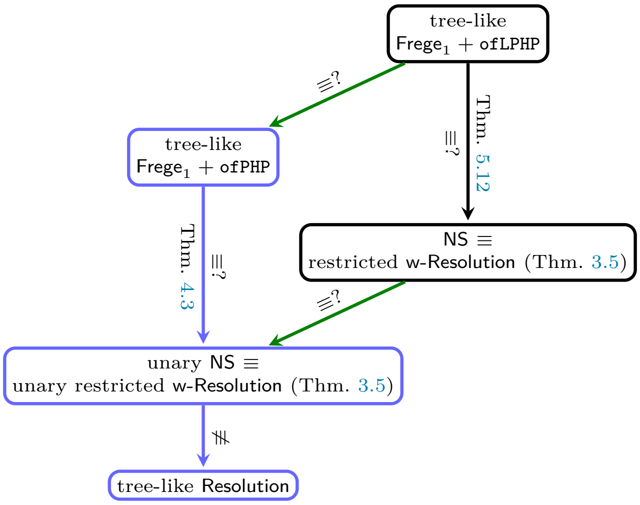

Fig. 2. The p-simulations for NS. The notation is the same of Fig. 1.

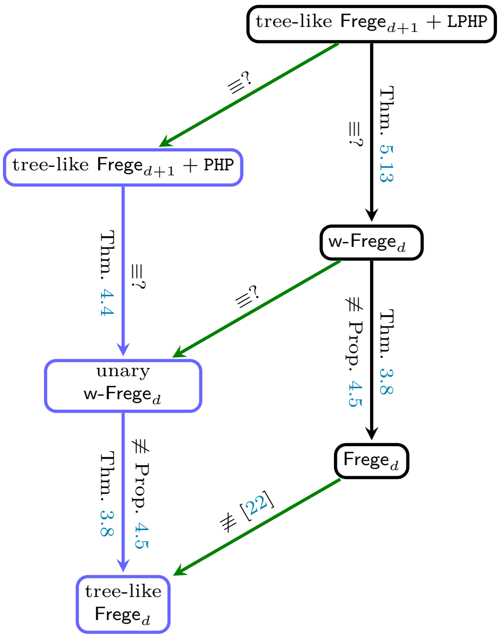

Fig. 3. The p-simulations for $\mathsf{w\text{-}Frege}_d$. The notation is the same of Fig. 1. See [22] for the separation between tree-like Frege_d and Frege_d.

is a completely new generalization of PHP. This naturally leaves several open questions about it. Including the obvious one of proving that depth-d Frege + LPHP is strictly weaker than Frege (see Section 6 for a list of open problems).

1.1. Connections with previous work

This article originated in the context of proof systems for MaxSAT extending MaxSAT Resolution, such as, for instance, DRMaxSAT [10]. Such systems, when seen as usual propositional proof systems, are stronger than Resolution, since they are able to prove some versions of PHP. We were interested to see if they could also prove some different natural combinatorial principles. These proof systems are simulated by w-Resolution (previously called *MaxSAT* Resolution *with Extension* in [25,26]).

Since SA and w-Resolution are equivalent, the question about MaxSAT proof systems morphed into asking whether SA was actually able to prove something significantly different from PHP (in addition to what Resolution can prove).

In this article, we give a first answer to the questions of the strength of SA and NS. SA can, at best, prove principles easily reducible to LPHP (in addition to what Resolution can prove). Similarly, NS can, at best, prove principles easily reducible to ofLPHP, in addition to what Resolution can prove (with a small increase in depth).

The starting point of our work is [10], where the authors prove that DRMaxSAT is simulated by bounded-depth Frege + PHP. The simulations upper-bounding the strength of SA, NS and $\mathsf{w\text{-}Frege}_d$ in this article widely generalize the simulation in [10]. This was possible via the language of w-Resolution and $\mathsf{w\text{-}Frege}_d$, a new way of looking at SA and NS (and other semi-algebraic proof systems).

1.2. Organization of the paper

Section 2 contains all the basic definitions: the notion of Frege_d, $\mathsf{Frege}_d + \phi$, and the semi-algebraic proof systems NS and SA. **Section 3** introduces the proof systems w-Resolution, restricted w-Resolution, and $\mathsf{w\text{-}Frege}_d$, and proves some basic facts about them. **Section 4** contains the definition of the pigeonhole principle PHP and the simulation of unary SA (resp. unary NS) by Frege_1 + PHP (resp. Frege_1 + ofPHP). **Section 5**, builds on the previous section and introduces a *leveled* version of the pigeonhole principle LPHP. We show how to refute it in SA and how to simulate SA by Frege_1 + LPHP. **Section 6** briefly recaps some aspects of this article and suggests some open problems.

2. Preliminaries

For $n \in \mathbb{N}$, let $[n] = \{1, \ldots, n\}$. A *propositional proof system* is a polynomial time function $P\colon \{0,1\}^* \to \{0,1\}^*$ whose range is exactly the set TAUT of propositional tautologies in the DeMorgan language [14]. The notion we use to compare the strength of two propositional proof systems is the notion of *p-simulation*. Given two propositional proof systems P, Q we say that P *p-simulates* Q if there exists a polynomial time function $f\colon \{0,1\}^* \to \{0,1\}^*$ such that for all strings x, $Q(x) = P(f(x))$. If P p-simulates Q and Q p-simulates P we say that P and Q are *p-equivalent*. If P p-simulates Q and they are not p-equivalent we say that the p-simulation is *strict*.

2.1. Constant-depth Frege systems

We follow the notation and definitions of [7] with minor changes. Propositional formulas are constructed from *literals*, i.e. Boolean variables x_i or negated variables $\neg x_i$, and unbounded fan-in conjunctions $\bigwedge$ and disjunctions $\bigvee$.

All formulas are either literals, $\bigvee$-*formulas* or $\bigwedge$-*formulas*. They are defined inductively:

- If Φ is a finite set of literals and $\bigvee$-formulas, then $\bigwedge \Phi$ is a $\bigwedge$-formula.
- If Φ is a finite set of literals and $\bigwedge$-formulas, then $\bigvee \Phi$ is a $\bigvee$-formula.

The point of this definition is that an $\bigwedge$-formula cannot be the argument of an $\bigwedge$, hence intuitively, adjacent $\bigwedge$ (resp. $\bigvee$) must be collapsed.

Definition 2.1 *(depth-d formulas)*. Let $d \in \mathbb{N}$. The classes of formulas Θ_d over a set of variables X are defined inductively as follows:

1. $\phi \in \Theta_0$ iff ϕ is a *literal*, i.e. either x or the negation $\neg x$ of some variable $x \in X$.
2. $\phi \in \Theta_{d+1}$ iff $\phi \in \Theta_d$ or $\phi = \bigwedge \Psi$ or $\phi = \bigvee \Psi$, where Ψ is a finite subset of Θ_d.

We refer to $\phi \in \Theta_d$ as ϕ *being of depth* d.

For $\phi \in \Theta_d$ we denote by $\neg\phi$ the formula in Θ_d obtained from ϕ by interchanging $\bigvee$ and $\bigwedge$ and interchanging variables and their negations.

A Θ_d*-cedent* is a finite multiset of formulas of depth d. A Θ_0-cedent is a *clause*. The intended meaning of a cedent Γ is $\bigvee \Gamma$. A *CNF formula* F is a set of clauses. The intended meaning of F is the conjunction of its members. We sometimes abuse notation by writing a cedent $\Gamma \cup \Phi$ simply as Γ, Φ.

Definition 2.2 *(*Frege_d*).* Let $\mathcal{F}$ be a set Θ_d-cedents. A Frege_d *derivation* of a Θ_d-cedent Γ is a tree T in which each node is labeled with a Θ_d-cedent, the root has label Γ, each leaf has label either the empty cedent or a cedent from $\mathcal{F}$, and for each node in the tree the label it gets is a consequence of the labels of its parents via one of the following inference rules:

$$\frac{\Delta, \phi, \phi}{\Delta, \phi}\ (\textsc{contraction}) \qquad \frac{}{\phi, \neg\phi}\ (\textsc{excluded middle})$$

$$\frac{\Delta, \phi \quad \text{for } \phi \in \Phi}{\Delta, \bigwedge \Phi}\ (\textstyle\bigwedge\text{-}\textsc{introduction}) \qquad \frac{\Delta, \Phi}{\Delta, \bigvee \Phi}\ (\textstyle\bigvee\text{-}\textsc{introduction})$$

$$\frac{\Delta, \neg\phi \qquad \Delta, \phi}{\Delta}\ (\textsc{symmetric cut}) \qquad \frac{\Delta}{\Delta,\ \Delta'}\ (\textsc{weakening})$$

where the cedents Δ, Δ', Φ are Θ_d-cedents and $\bigvee \Phi, \bigwedge \Phi, \phi, \neg\phi$ are formulas of depth d. The *size* of T is the number of symbols of distinct cedents in the derivation. The *height* of T is its height as a tree rooted in Γ, the length of its longest path from the root to a leaf. If we count the number of symbols in *all* occurrences of cedents we use the adjective *tree-like*. A Frege_d *refutation* of $\mathcal{F}$ is a derivation of the empty cedent.

The definition of Frege_d in [7] is essentially the one given above with the CONTRACTION rule given implicitly, since their cedents are sets. For us, it is more convenient to consider multisets and to have the rule given explicitly. The propositional proof system $\mathsf{Resolution}$ is Frege_0. In this system, the $\bigwedge$ and $\bigvee$ rules cannot be applied.

Given $\boldsymbol{\phi} = (\phi_n)_{n\in\mathbb{N}}$ a family of unsatisfiable cedents, for instance ϕ_n being the pigeonhole principle $\mathtt{PHP}^{n+1}_n$ (see Section 4 for the definition of $\mathtt{PHP}^{n+1}_n$), the notion of $\mathsf{Frege}_d + \boldsymbol{\phi}$ has been considered in very similar terms, for instance in [1,6], and it is also very common in the context of bounded arithmetic (see for instance [23]).

Informally, to refute a formula F in $\mathsf{Frege}_d + \boldsymbol{\phi}$ we can either (1) refute F in Frege_d, or (2) derive a substitution instance of ϕ_n, for some n. This is a refutation of F, since $\boldsymbol{\phi}$ is a family of unsatisfiable formulas. In the system $\mathsf{Frege}_d + \boldsymbol{\phi}$ we allow the formulas ϕ_n to be used only once. Formally, the definition is the following.

Definition 2.3 *(*$\mathsf{Frege}_d + \boldsymbol{\phi}$*).* Let $\boldsymbol{\phi} = (\phi_n)_{n\in\mathbb{N}}$, where ϕ_n is an unsatisfiable set of s-many Θ_d-cedents in n variables. A refutation of a set of Θ_d-cedents F in $\mathsf{Frege}_d + \boldsymbol{\phi}$ is a set of s Frege_d derivations $\Gamma_1, \ldots, \Gamma_s$ of $G_1, \ldots, G_s$ such that: either (1) $s = 1$ and $G_1 = \emptyset$, i.e. Γ_1 is a refutation of F in Frege_d, or (2) there is a $n \in \mathbb{N}$ such that the set of cedents $\{G_1, \ldots, G_s\}$ is a substitution instance of ϕ_n.[2] The *height* of the refutation is the maximum height of $\Gamma_1, \ldots, \Gamma_s$. The *size* of the refutation is the sum of the sizes of $\Gamma_1, \ldots, \Gamma_s$.

[2] Let ψ_n be in the variables $x_1, \ldots, x_n$. The cedent $\{G_1, \ldots, G_s\}$ is a substitution instance of ϕ_n if there are depth-d formulas $\psi_1, \ldots, \psi_n$ s.t. once we substitute in ϕ_n all the x_is with the ψ_is we get exactly $\{G_1, \ldots, G_s\}$.

Even though we know that tree-like Frege_{d+1} is equivalent to Frege_d [23], tree-like $\mathsf{Frege}_{d+1} + \phi$ is not the same as $\mathsf{Frege}_d + \phi$, since in the first system we allow to derive substitution instances of ϕ that could be formulas of depth $d + 1$.

Definition 2.4 *(*$\mathsf{Res}(k)$*).* Let $d, k \in \mathbb{N}$. The system $\mathsf{Res}(k)$ is the restriction of Frege_1 where the $\bigwedge$-INTRODUCTION rule in Definition 2.2 is limited to Θ_0-cedents (i.e. sets of clauses) Φ of size at most k. Let $\boldsymbol{\phi} = (\phi_n)_{n\in\mathbb{N}}$, where ϕ_n is a set of s many Θ_0-cedents in n variables. $\mathsf{Res}(k) + \boldsymbol{\phi}$ is then defined in an analogous way as $\mathsf{Frege}_d + \boldsymbol{\phi}$ in Definition 2.3.

2.2. Algebraic and semi-algebraic proof systems

In this section, we define formally the proof systems Nullstellensatz [5] and Sherali-Adams [32]. Let X be the set of variables $x_1, \dots, x_n, \bar{x}_1, \dots, \bar{x}_n$. Given the ordered ring of the integers $\mathbb{Z}$, by $\mathbb{Z}[X]$ we denote the set of polynomials in the variables X and coefficients in $\mathbb{Z}$.

Definition 2.5 *(Nullstellensatz,* NS*).* Given polynomials $p_0, \dots, p_\ell \in \mathbb{Z}[X]$, a *Nullstellensatz proof over* $\mathbb{Z}$ ($\mathsf{NS}_\mathbb{Z}$) of the equality $p_0 = 0$ from the equalities $p_1 = 0, \dots, p_\ell = 0$ is a polynomial identity of the form

$$p_0 = \sum_{i=1}^{\ell} q_i p_i + \sum_{j=1}^{n} r_j (x_j^2 - x_j) + \sum_{j=1}^{n} r'_j (x_j + \bar{x}_j - 1)\,, \tag{1}$$

where q_i, r_j, r'_j are polynomials in $\mathbb{Z}[X]$. A *refutation* of $p_1 = 0, \dots, p_\ell = 0$ is a derivation of the equality $c = 0$ where $c \in \mathbb{Z} \setminus \{0\}$. The *size* of the polynomial identity in (1) is the length of a bit-string representing the polynomials q_i, r_j, r'_j, including the coefficients. The *degree* of the polynomial identity in (1) is the maximum degree of the polynomials q_i, r_j, r'_j.

Definition 2.6 *(Sherali-Adams,* SA*).* Given a set of polynomials $p_0, \dots, p_\ell \in \mathbb{Z}[X]$, a *Sherali-Adams proof over* $\mathbb{Z}$ ($\mathsf{SA}_\mathbb{Z}$) *of* $p_0 \geq 0$ *from* $p_1 \geq 0, \dots, p_\ell \geq 0$ is a polynomial identity of the form

$$p_0 = \sum_{i=1}^{\ell} q_i p_i + \sum_{j=1}^{n} r_j (x_j^2 - x_j) + \sum_{j=1}^{n} r'_j (x_j + \bar{x}_j - 1) + q_0\,, \tag{2}$$

where r_j, r'_j are polynomials in $\mathbb{Z}[X]$ and the q_is are polynomials with positive coefficients. A *refutation* of a set of polynomial inequalities $p_1 \geq 0, \dots, p_\ell \geq 0$ is a derivation of $c \geq 0$ where $c \in \mathbb{Z}$ and negative. The *size* of the polynomial identity in (2) is the length of a bit-string representing the polynomials q_i, r_j, r'_j, including the coefficients. The *degree* of the polynomial identity in (2) is the maximum degree of the polynomials q_i, r_j, r'_j.

In Definitions 2.5 and 2.6, substituting the variables $\bar{x}_1, \dots, \bar{x}_n$ with $1 - x_1, \dots, 1 - x_n$ (and the terms involving r'_j will be identical to 0) results in versions of NS and SA exponentially weaker [16] with respect to size. The degree of the two versions of the systems is obviously the same.

Nullstellenstatz and Sherali-Adams could be defined on generic rings (ordered in the case of SA). In this paper we consider only NS and SA over the ring $\mathbb{Z}$, $\mathsf{NS}_\mathbb{Z}$ and $\mathsf{SA}_\mathbb{Z}$, resp., hence from now on we refer to them as NS and SA omitting the reference to $\mathbb{Z}$. When we restrict all the polynomials appearing in NS and SA derivations to have coefficients ± 1, we refer to those systems as *unary* NS and *unary* SA. Equivalently, we could think of unary NS and unary SA as the versions of NS and SA where the coefficients, instead of being encoded in binary, are encoded in unary.

$$\frac{[C,\ell,\ell;w]}{[C,\ell;w]}\ (\textsc{contraction}) \qquad \frac{}{[\ell,\neg\ell;w]}\ (\textsc{excluded middle})$$

$$\frac{[C,\neg\ell;w],\ [C,\ell;w]}{[C;w]}\ (\textsc{symmetric cut}) \qquad \frac{[C;w]}{[C,\neg\ell;w],\ [C,\ell;w]}\ (\textsc{split})$$

$$\frac{[C;u],\ [C;w]}{[C;u+w]}\ (\textsc{fold}) \qquad \frac{[C;u+w]}{[C;u],\ [C;w]}\ (\textsc{unfold})$$

$$\frac{[C;u],\ [C;-u]}{}\ (\textsc{removal}) \qquad \frac{}{[C;u],\ [C;-u]}\ (\textsc{introduction})$$

Fig. 4. Inference rules of w-Resolution. C, D are clauses, ℓ is a literal, $u, w \in \mathbb{Z}$.

The natural encoding of sets of clauses in the context of (semi-)algebraic proof systems is the following. A clause $C = \{x_i : i \in I\} \cup \{\neg x_j : j \in J\}$ is represented as the monomial $-\prod_{i\in I} \bar{x}_i \prod_{j\in J} x_j$, intended to be $= 0$ in NS, and ≥ 0 in SA. In the algebraic context, we follow the common convention that a variable being 0 means it is true. In the propositional context it is the opposite, 0 means false and 1 means true. A set of clauses is then represented by the set of the (in)equalities encoding its clauses.

Under this natural representation, it is well-known that SA p-simulates Resolution (see for instance [3, Lemma 3.5]) and NS with unary coefficients p-simulates tree-like Resolution. Moreover, both p-simulations are known to be strict.

3. Weighted resolution and weighted depth-d Frege

A *weighted clause over* $\mathbb{Z}$ is a pair $[C; w]$ where C is a clause and $w \in \mathbb{Z}$. First we define the system *weighted* Resolution (w-Resolution). This system comes essentially from [12,26].

Definition 3.1 *(w-Resolution).* A w-Resolution derivation (over $\mathbb{Z}$) of a clause C from a set of clauses $\{C_1, \dots, C_m\}$ is a sequence $\mathcal{L}_1, \dots, \mathcal{L}_s$ of multisets of weighted clauses over $\mathbb{Z}$ such that:

1. $\mathcal{L}_1 = \{[C_1; w], \dots [C_m; w]\}$ where $w \in \mathbb{N}$,
2. $[C; z] \in \mathcal{L}_s$ for some $z > 0$,
3. all clauses in $\mathcal{L}_s \setminus \{[C; z]\}$ have positive weights,
4. each $\mathcal{L}_i$ is obtained from $\mathcal{L}_{i-1}$ by applying one of the inference rules in Fig. 4 as substitution rules, i.e. removing the premises from $\mathcal{L}_{i-1}$ and adding the conclusions.

A w-Resolution *refutation* of $\mathcal{F}$ is a w-Resolution derivation of the empty clause. The *size* of a w-Resolution derivation $\mathcal{L}_1, \dots, \mathcal{L}_s$ is the total number of occurrences of symbols in $\mathcal{L}_1, \dots, \mathcal{L}_s$ including the weights. Unless explicitly stated, the weights are assumed to be encoded in binary. If the weights are restricted to $-1, 1$ then we call the system unary w-Resolution. In the system with weights in unary there are no applications of the FOLD/UNFOLD rules and the weighted clauses in $\mathcal{L}_1$ are given as a multiset, instead of $[C_i; w]$ we have a multiset consisting in w many copies of $[C_i; 1]$.

The intuition, behind the definition of weighted proof systems, is that we are allowed to make assumptions (via the INTRODUCTION rule) and the negative weights are a way to have some control over them. If we need to use an assumption k times, we also need to justify it with weight k. At some point, the assumptions must end-up being justified, via the REMOVAL rule. The system then needs to keep track of the weights in a consistent way, and this is done using inference rules as substitution rules (i.e. removing the premises of the rule and adding its conclusions).

A natural subsystem of w-Resolution is *restricted* w-Resolution. In Theorem 3.5 we will see that restricted w-Resolution is equivalent to NS, as w-Resolution is equivalent to SA.

Definition 3.2 *(restricted* w-Resolution*).* A *restricted* w-Resolution derivation (over $\mathbb{Z}$) of a clause C from a set of clauses $\{C_1, \dots, C_m\}$ is a sequence $\mathcal{L}_1, \dots, \mathcal{L}_s$ of multisets of weighted clauses over $\mathbb{Z}$ with the same requirement as in Definition 3.1 with the condition (3) substituted by the condition

(3′) all clauses in $\mathcal{L}_s \setminus \{[C; z]\}$ have positive weights and, moreover, they are also weakenings of clauses in $\{C_1, \dots, C_m\}$.

Notice that the rules in Fig. 4 are weighted versions of the inference rules of Frege_0 (see Definition 2.2) with one exception: the SPLIT rule instead of the WEAKENING rule, although both rules result in equivalent systems.

Using the SPLIT rule instead of the WEAKENING has several advantages. First it allows us to work with a w-Resolution system where the negative weights are allowed only in the INTRODUCTION/REMOVAL rules and not in the other rules (see Remark 3.3). The second advantage is that it simplifies the proof of the equivalence with SA/NS and w-Resolution/restricted w-Resolution (see Theorem 3.5). Finally, with the SPLIT rule, every rule in Fig. 4, has the property that, for any assignment α, the total weight of the falsified premises equals the total weight of the falsified conclusions. This means that the set of rules of w-Resolution will be usable also in the context of weighted maxSAT, with the only difference that the weights of the initial clauses are part of the input.

Remark 3.3. In w-Resolution (and restricted w-Resolution) allowing to have negative weights only in the INTRODUCTION/REMOVAL rules results in a p-equivalent system. To see this we show how to simulate a SYMMETRIC CUT on clauses with negative weight $-w$:

$$\dfrac{\dfrac{\dfrac{[C, x; -w] \quad [C, \neg x; -w]}{[C, x; -w] \quad [C, \neg x; -w] \quad [C; -w] \quad [C; w]}\ \text{INTRODUCTION}}{[C, x; -w] \quad [C, \neg x; -w] \quad [C; -w] \quad [C, x; w] \quad [C, \neg x; w]}\ \text{SPLIT}}{[C; -w]}\ \text{REMOVAL}$$

For the other rules the argument is equally simple.

Moreover, w-Resolution is also p-equivalent to w-Resolution with all the weights only allowed to be powers of 2. The same is true for restricted w-Resolution. To see this, notice that in w-Resolution, we can substitute each weighted clause $[C; w]$ where $w = \sum_{j \in J} 2^j$, by the set of clauses $\{[C; 2^j] \mid j \in J\}$ and perform the rule used on the clause $[C; w]$ to the clauses $\{[C; 2^j] \mid j \in J\}$.

Lemma 3.4. *The proof systems* w-Resolution *and restricted* w-Resolution *are sound.*

Proof. Given a truth assignment $\alpha \colon \{x_1, \dots, x_n\} \to \{\top, \bot\}$ and a multiset of weighted clauses $\mathcal{L}$, let

$$W(\mathcal{L}, \alpha) = \sum_{\substack{[C;w] \in \mathcal{L} \\ \alpha(C) = \bot}} w\,.$$

Let $\mathcal{F}$ be a set of clauses and suppose $\mathcal{F}$ has a w-Resolution refutation $(\mathcal{L}_1, \dots, \mathcal{L}_s)$. If $\mathcal{F}$ was satisfiable, then there would exist an assignment α satisfying all clauses in $\mathcal{F}$, hence $W(\mathcal{L}_1, \alpha) = 0$. Since $[\bot; w] \in \mathcal{L}_s$ for some $w > 0$ and $\mathcal{L}_s$ has all positive weights, then

$$W(\mathcal{L}_s, \alpha) > 0\,.$$

$$\frac{[\Gamma,\phi,\phi;w]}{[\Gamma,\phi;w]}\ (\textsc{contraction}) \qquad \frac{}{[\phi,\neg\phi;w]}\ (\textsc{excluded middle})$$

$$\frac{[\Gamma,\phi_1;w]\quad[\Gamma,\phi_2;w]}{[\Gamma,\phi_1\wedge\phi_2;w],\quad[\Gamma,\phi_1,\phi_2;w]}\ (\textstyle\bigwedge\text{-}\textsc{introd.}) \qquad \frac{[\Gamma,\Phi;w]}{[\Gamma,\bigvee\Phi;w]}\ (\textstyle\bigvee\text{-}\textsc{introd.})$$

$$\frac{[\Gamma,\neg\phi;w],\ [\Gamma,\phi;w]}{[\Gamma;w]}\ (\textsc{symmetric cut}) \qquad \frac{[\Gamma;w]}{[\Gamma,\neg\phi;w],\ [\Gamma,\phi;w]}\ (\textsc{split})$$

$$\frac{[\Gamma;u],\ [\Gamma;w]}{[\Gamma;u+w]}\ (\textsc{fold}) \qquad \frac{[\Gamma;u+w]}{[\Gamma;u],\ [\Gamma;w]}\ (\textsc{unfold})$$

$$\frac{[\Gamma;u],\ [\Gamma;-u]}{}\ (\textsc{removal}) \qquad \frac{}{[\Gamma;u],\ [\Gamma;-u]}\ (\textsc{introduction})$$

Fig. 5. Inference rules of w-Frege$_d$. The cedents $\Gamma, \Phi, \bigvee\Phi, \bigwedge\Phi, \phi, \neg\phi$ all are Θ_d-cedents, $u, w \in \mathbb{Z}$.

On the other hand, the inference rules of Fig. 4 guarantee that in the derivation $(\mathcal{L}_1, \dots, \mathcal{L}_s)$

$$W(\mathcal{L}_1, \alpha) = W(\mathcal{L}_2, \alpha) = \cdots = W(\mathcal{L}_s, \alpha) = 0\,.$$

This means that $\mathcal{F}$ must be unsatisfiable. □

One of the reasons we introduced w-Resolution and its restricted version is that they are a characterization of SA and NS in a language similar to the one used for Frege$_d$.

Theorem 3.5. *(Unary)* SA *is p-equivalent to (unary) w-*Resolution*.*
(Unary) NS *is p-equivalent to (unary) restricted w-*Resolution*.*
Moreover, degree-d proofs in SA/NS *correspond to width-d weighted proofs, where the width of a proof is the maximum number of literals in a clause of the proof.*

Proof. (sketch) The part of this theorem for SA with binary coefficients follows via p-equivalence with *circular* Resolution: w-Resolution is p-equivalent to circular Resolution [12,31] and circular Resolution is p-equivalent to SA [2].

It is not hard to see that refutations in the systems NS/SA and refutations in (restricted) w-Resolution are two different ways of looking at the same thing. The multisets $\mathcal{L}_1, \dots, \mathcal{L}_s$ in a w-Resolution refutation are in a correspondence with partial sums of SA/NS refutations. The binomials $m(x_j^2 - x_j)$ correspond to applications of the CONTRACTION rule, and the trinomials $m(x_j + \bar{x}_j - 1)$ correspond to applications of the SPLIT/SYMM. CUT rules. The argument that uses these intuitions is in [8]. □

We conclude this section showing a natural generalization of weighted clauses and w-Resolution to Θ_d-cedents and *weighted* Frege$_d$ (w-Frege$_d$).

A *weighted* Θ_d*-cedent over* $\mathbb{Z}$ is a pair $[\Gamma; w]$ where Γ is a Θ_d-cedent and $w \in \mathbb{Z}$.

Definition 3.6 *(weighted* Frege$_d$*, w-*Frege$_d$*).* A *weighted* Frege$_d$ (w-Frege$_d$) derivation (over $\mathbb{Z}$) of a Θ_d-cedent Γ from a set of Θ_d-cedents $\mathcal{F} = \{\Gamma_1, \dots, \Gamma_m\}$ is a sequence $\mathcal{L}_1, \dots, \mathcal{L}_s$ of multisets of weighted Θ_d-cedents over $\mathbb{Z}$ such that:

1. $\mathcal{L}_1 = \{[\Gamma_1; w], \dots [\Gamma_m; w]\}$ where $w \in \mathbb{N}$,
2. $[\Gamma; z] \in \mathcal{L}_s$ for some $z > 0$,
3. all cedents in $\mathcal{L}_s \setminus \{[\Gamma; z]\}$ have positive weights,
4. each $\mathcal{L}_i$ is obtained from $\mathcal{L}_{i-1}$ by applying one of the inference rules in Fig. 5 as substitution rules, i.e. removing the premises from $\mathcal{L}_{i-1}$ and adding the conclusions.

A $\mathsf{w\text{-}Frege}_d$ *refutation* of $\mathcal{F}$ is a $\mathsf{w\text{-}Frege}_d$ derivation of the empty cedent. The *size* of a $\mathsf{w\text{-}Frege}_d$ derivation $\mathcal{L}_1, \dots, \mathcal{L}_s$ is the total number of occurrences of symbols in $\mathcal{L}_1, \dots, \mathcal{L}_s$ including the weights. Unless explicitly stated, the weights are assumed to be encoded in binary. If the weights are restricted to $-1, 1$ then we call the system unary $\mathsf{w\text{-}Frege}_d$. In the system with weights in unary there are no applications of the FOLD/UNFOLD rules and the weighted cedents in $\mathcal{L}_1$ are given as a multiset, instead of $[\Gamma_i; w]$ we have a multiset consisting in w many copies of $[\Gamma_i; 1]$, if $w > 0$, or a multiset consisting in $-w$ many copies of $[\Gamma_i; -1]$, if $w < 0$.

Notice that the rules in Fig. 5 are weighted versions of the inference rules of Frege_d (see Definition 2.2) with two exceptions: the SPLIT and the $\bigwedge$-INTROD. The use of the SPLIT rule instead of the WEAKING rule implies that it is enough to consider negative weights only in the INTRODUCTION/REMOVAL rules, that is Remark 3.3 generalizes from $\mathsf{w\text{-}Resolution}$ to $\mathsf{w\text{-}Frege}_d$.

The version of the $\bigwedge$-INTROD. used in Fig. 5 has the property that, for any assignment α, the total weight of the falsified premises equals the total weight of the falsified conclusions. That is the rules $\mathsf{w\text{-}Frege}_d$ would be also appropriate for weighted maxSAT, thus generalizing $\mathsf{w\text{-}Resolution}$ also to the context of maxSAT.

Lemma 3.7. *For every $d \in \mathbb{N}$, the proof system $\mathsf{w\text{-}Frege}_d$ is sound.*

Proof. The proof is identical to the proof of Lemma 3.4, just changing “clauses” with “cedents”. □

We conclude this section proving that $\mathsf{w\text{-}Frege}_d$ p-simulates Frege_d. This is a generalization of the known fact that SA p-simulates $\mathsf{Resolution}$.

Theorem 3.8. *For every $d \in \mathbb{N}$, $\mathsf{w\text{-}Frege}_d$ p-simulates Frege_d.*

Proof. To simulate Frege_d with $\mathsf{w\text{-}Frege}_d$ we need to argue two things: first how to simulate each inference rule of Frege_d with rules of $\mathsf{w\text{-}Frege}_d$ and, second, how to set the weights of a $\mathsf{w\text{-}Frege}_d$ derivation to simulate a Frege_d derivation.

The inference rules of $\mathsf{w\text{-}Frege}_d$ are the same of the rules of Frege_d (setting aside the weights for the moment) with the exception of the WEAKENING and the $\bigwedge$-INTRODUCTION. By repeated applications of the SPLIT and $\bigwedge$-INTRODUCTION rules of $\mathsf{w\text{-}Frege}_d$ it is immediate to see that is possible to produce the same consequences of the WEAKENING and $\bigwedge$-INTRODUCTION rules of Frege_d (having derived possibly some extra cedents).

Now, since the rules of $\mathsf{w\text{-}Frege}_d$ are substitution rules, given a Frege_d derivation π of some cedent Γ, to simulate π in $\mathsf{w\text{-}Frege}_d$ we need to take into account, for each cedent of π, the number of times it is used as a premise of some rule in π.

To assign weights to Θ_d-cedents, the idea is to set $[\Gamma; 1]$ and then proceed bottom-up in π setting the weight of any Θ_d-cedent looking at all the times it is used and summing the weights of those weighted cedents. It is immediate to see that the weights might be as big as 2^S, where S is the size of π, in the worst case. Since the weights are encoded in binary this gives a p-simulation. Notice that, since the inference rules of $\mathsf{w\text{-}Frege}_d$ are binary, it is possible to get a slight better bound on the weights, in the form $\phi^S/\sqrt{5}$ where ϕ is the golden ratio [11, Lemma 31]. □

4. The pigeonhole principle and unary NS/SA

In this section we prove the p-simulations relative to the *unary* parts of Fig. 1, 2 and 3.

Definition 4.1 *(Pigeonhole principle).* Let $m, n \in \mathbb{N}$ with $m > n$ and let $p_{i,j}$ be Boolean variables with $i \in [m]$ and $j \in [n]$. The *pigeonhole principle* is the set of clauses

$$\begin{aligned}\mathtt{PHP}_n^m &= \{\{p_{i,1},\dots,p_{i,n}\} : i \in [m]\} \\ &\quad \cup \{\{\neg p_{i,j}, \neg p_{i',j}\} : i,i' \in [m] \text{ distinct}, j \in [n]\}\,.\end{aligned}$$

The *onto-functional* pigeonhole principle $\mathtt{ofPHP}_n^m$ is the formula $\mathtt{PHP}_n^m$ together with the set of cedents

$$\{\{\neg p_{i,j}, \neg p_{i,j'}\} : i \in [m] \quad j,j' \in [n] \text{ distinct}\}\,, \tag{3}$$

the *functionality* axioms, and the set

$$\{\{p_{1,j},\dots,p_{m,j}\} : j \in [n]\}\,, \tag{4}$$

the *onto* axioms. Given a bipartite graph $G = (P \cup H, E)$ with $|P| = m$ and $|H| = n$, the *graph* pigeonhole principle $\mathtt{PHP}_n^m(G)$ is the formula $\mathtt{PHP}_n^m$ restricted by a partial assignment mapping $p_{i,j} = \bot$ for all $(i,j) \notin E$, i.e. we remove the literal $p_{i,j}$ from every clause of $\mathtt{PHP}_n^m$ where it appears and remove all clauses of $\mathtt{PHP}_n^m$ containing $\neg p_{i,j}$. The onto-functional *graph* pigeonhole principle $\mathtt{ofPHP}_n^m(G)$ is defined in the same way.

It is well-known that $\mathtt{PHP}_n^{n+1}$ has polynomial size unary SA refutations and $\mathtt{ofPHP}_n^{n+1}$ has polynomial size unary NS refutations. Let's recall briefly the argument. To refute $\mathtt{PHP}_n^{n+1}$ in SA first derive

$$\sum_{j \in [n+1]} \sum_{i \in [n]} p_{i,j} - (n+1) \geq 0 \tag{5}$$

$$n - \sum_{i \in [n]} \sum_{j \in [n+1]} p_{i,j} \geq 0\,. \tag{6}$$

Then, sum the two inequalities to get $-1 \geq 0$. The same argument can be easily adapted to show the results for unary NS. Moreover, for a bipartite graph G with maximum degree d, $\mathtt{PHP}_n^{n+1}(G)$ has degree-d unary SA refutations and $\mathtt{ofPHP}_n^{n+1}(G)$ has degree-d unary NS refutations.

We now show some sort of converse of the previous results: $\mathsf{Frege}_1 + \mathtt{PHP}_n^{n+1}(G)$ p-simulates unary SA and $\mathsf{Frege}_1 + \mathtt{ofPHP}_n^m(G)$ p-simulates unary NS.

Theorem 4.2. *For every d, tree-like $\mathsf{Res}(d) + \mathtt{PHP}_n^{n+1}(G)$ p-simulates degree-d unary SA, where G is restricted to bipartite graphs of degree at most 3 and the height of the tree-like $\mathsf{Res}(d) + \mathtt{PHP}_n^{n+1}(G)$ derivations is 5.*

Notice that tree-like $\mathsf{Res}(n)$ is tree-like Frege_1. The proof of this result is loosely inspired by the proof of [10, Theorem 4].

Proof. We use the characterization of SA given by Theorem 3.5. Let $\mathcal{L} = (\mathcal{L}_1,\dots,\mathcal{L}_s)$ be a weighted $\mathsf{Resolution}$ refutation of a set of clauses $F = \{C_1,\dots,C_m\}$. The strategy for the simulation is to obtain a tree-like $\mathsf{Res}(d)$ derivation of a substitution instance of the cedents of $\mathtt{PHP}_n^{n+1}$ for some n. To describe this derivation it is useful to reason semantically, arguing that from an assignment satisfying all the clauses in F we obtain a one-to-one mapping from $n+1$ to n using the weighted $\mathsf{Resolution}$ derivation $\mathcal{L}$.

Since the weights in $\mathcal{L}$ are in unary, all the weights in π are just ± 1. Moreover, in $\mathcal{L}$, there will be no application of the FOLD/UNFOLD rules. Without loss of generality, we can assume the weights in the CONTRACTION/SYMM.CUT/SPLIT/EXCL. MIDDLE rules are $+1$ (see Remark 3.3).

Let $\mathcal{L}_{s+1} = \{[\emptyset; 1]\}$ and let P be the multiset given by the disjoint union of the multisets $\mathcal{L}_1,\dots,\mathcal{L}_{s+1}$ and H be the multiset given by the disjoint union of the multisets $\mathcal{L}_1,\dots,\mathcal{L}_s$. In particular, $|P| = |H| + 1$. The multiset P will represent the pigeons and H the holes.

Now for each $\alpha \in P$ and each $\beta \in H$ we want to define $p_{\alpha,\beta}$ as conjunctions of a set of at most d literals, such that we have small tree-like $\mathrm{Res}(d)$ derivations of the cedents $\{p_{\alpha,\beta} : \beta \in H\}$ for all $\alpha \in P$, and

$\{\neg p_{\alpha,\beta}, \neg p_{\alpha',\beta}\}$ for all $\beta \in H$, and distinct $\alpha, \alpha' \in P$. We also want that $p_{\alpha,\beta} \neq \bot$ for at most 3 values of β and $p_{\alpha,\beta} \neq \bot$ for at most 3 values of α.

Given $\alpha \in P$, let $\alpha = [C_\alpha; w_\alpha]$ and let i_α be the index of the level to which α belongs, i.e. the unique i_α such that $\alpha \in \mathcal{L}_{i_\alpha}$; similarly for $\beta \in H$. Given α, β as above, we say that β is a *CONTRACTION/SYMM.CUT/SPLIT-premise* of α if $i_\alpha = i_\beta + 1$ and between the layers $\mathcal{L}_{i_\beta}$ and $\mathcal{L}_{i_\alpha}$ there is an application of the CONTRACTION/SYMM.CUT/SPLIT rule of weighted Resolution with β one of the premises and α one of the conclusions. There are no applications of the FOLD/UNFOLD rules, so the only rule having two premises is the SYMMETRIC CUT. We say that α is a *copy* of β if $i_\alpha = i_\beta + 1$ and between the layers $\mathcal{L}_{i_\alpha}$ and $\mathcal{L}_{i_\beta}$, the inference rule applied does not involve α or β. In particular, $[\emptyset; 1]$ in $\mathcal{L}_{s+1}$ is a copy of some element in $\mathcal{L}_s$. Moreover, if α is a copy of β, then $C_\alpha = C_\beta$ and $w_\alpha = w_\beta$. If $w_\alpha = 1$ we say that α is a *positive-copy* of β, if $w_\alpha = -1$ we say that α is a *negative-copy* of β. Finally, we say that α, β are *appearing (resp. disappearing) siblings* if $i_\alpha = i_\beta$ and α and β are the result of an INTRODUCTION rule on the layer $\mathcal{L}_{i_\alpha}$ (resp. α and β are used as premises of a REMOVAL rule on the layer $\mathcal{L}_{i_\alpha+1}$).

We now describe the candidate injective mapping from P to H. The pigeons $\alpha = [C_\alpha; 1]$ are mapped to holes β with $i_\beta = i_\alpha - 1$, or with $i_\beta = i_\alpha$, while the pigeons $\alpha = [C_\alpha; -1]$ are mapped to holes β with $i_\beta = i_\alpha + 1$, or with $i_\beta = i_\alpha$. The precise hole a pigeon α flies to depends on the truth value of C_α and related clauses. The notion of a clause being true or false is under a hypothetical assignment satisfying all the initial clauses in F.

- If C_α is true, then the pigeon α flies to the hole α. That is, we set $p_{\alpha,\alpha}$ to be the formula $\bigvee C_\alpha$ (see (8) below).
- If C_α is an initial clause, the pigeon α always flies to the hole α. In other words α as a member of P is mapped to α as a member of H. So we set $p_{\alpha,\alpha}$ to the tautology $x \vee \neg x$ (see (7)).
- If C_α is false and its weight is $+1$, the pigeon α flies either to the hole β corresponding to the false premise C_β used to derive it, or to the hole γ corresponding to the appearing sibling of C_α, that is $\gamma = [C_\alpha; -1]$. The way to say that C_α and C_β are false is to use the formula $\neg \bigvee C_\alpha \wedge \neg \bigvee C_\beta$, but this is redundant, since it is always the case that either C_α contains C_β (see (11)) or the opposite (see (9)).
- If C_α is false and the weight of C_α is -1 then α flies to its copy $[C_\beta, -1]$ in the direction of the proof ($i_\beta = i_\alpha + 1$), or to its disappearing sibling (see (10)). The way to define $p_{\alpha,\beta}$ is analogous as before.

Formally, $p_{\alpha,\beta}$ is the formula

$$x \vee \neg x \quad \text{if } \alpha = \beta \text{ and } \alpha \in \mathcal{L}_1 \,, \tag{7}$$

$$\bigvee C_\alpha \quad \text{if } \alpha = \beta \text{ and } \alpha \notin \mathcal{L}_1 \,, \tag{8}$$

$$\neg \bigvee C_\beta \quad \text{if } \begin{cases} \beta \text{ is a SYMM.CUT-premise of } \alpha \\ \beta \text{ is a CONTRACTION-premise of } \alpha \\ \alpha, \beta \text{ appearing siblings}, w_\alpha = 1 \\ \alpha \text{ is a positive-copy of } \beta \end{cases} \tag{9}$$

$$\neg \bigvee C_\beta \quad \text{if } \begin{cases} \alpha, \beta \text{ disappearing siblings}, w_\alpha = -1 \\ \beta \text{ is a negative-copy of } \alpha \end{cases} \tag{10}$$

$$\neg \bigvee C_\alpha \quad \text{if } \beta \text{ is a SPLIT-premise of } \alpha \,, \tag{11}$$

$$\bot \quad \text{otherwise} \,.$$

The totality axioms $\{p_{\alpha,\beta} : \beta \in H\}$ are easily derivable in tree-like Res(d) from the initial clauses $C_1, \ldots, C_m$. We need to check several cases.

If C_α is one of the initial clauses $C_1, \ldots, C_m$ or an instance of the EXCLUDED MIDDLE rule, in both cases $\{p_{\alpha,\beta} : \beta \in H\} = \{p_{\alpha,\alpha}\}$. The cedent $\{p_{\alpha,\alpha}\}$ can be obtained by the EXCLUDED MIDDLE rule and $\bigvee$-INTRODUCTION RULE.

If C_α is the result of the application of a CONTRACTION rule on C_β

$$\{p_{\alpha,\gamma} : \gamma \in H\} = \{\bigvee C_\alpha, \neg \bigvee C_\beta\}\,.$$

If C_α is the result of the application of a SPLIT rule on C_β or α is a copy of β or α, β are appearing/disappearing siblings then

$$\{p_{\alpha,\gamma} : \gamma \in H\} = \{\bigvee C_\alpha,\ \neg \bigvee C_\alpha\}$$

is an instance of the EXCLUDED MIDDLE rule of $\mathrm{Res}(d)$, the height to derive it is 1.

The only remaining case is when α is the conclusion of a SYMMETRIC CUT with premises β, β'. Then, $\bigvee C_\beta = \bigvee C_\alpha \vee x$ and $\bigvee C_{\beta'} = \bigvee C_\alpha \vee \neg x$, and the totality axiom for the pigeon α is

$$\{p_{\alpha,\gamma} : \gamma \in H\} = \{\bigvee C_\alpha,\ \neg \bigvee C_\alpha \wedge \neg x,\ \neg \bigvee C_\alpha \wedge x\}\,.$$

This formula can be derived by first deriving by EXCLUDED MIDDLE

$$\{\bigvee C_\alpha \vee x,\ \neg \bigvee C_\alpha \wedge \neg x\} \quad \text{and} \quad \{\bigvee C_\alpha \vee \neg x,\ \neg \bigvee C_\alpha \wedge x\}\,,$$

then by SYMMETRIC CUT on weakening of the previous two cedents we derive

$$\{\bigvee C_\alpha,\ \neg \bigvee C_\alpha \wedge \neg x,\ \neg \bigvee C_\alpha \wedge x\}\,.$$

This derivation has height 5.

The injectivity axioms $\{\neg p_{\alpha,\beta},\ \neg p_{\alpha',\beta}\}$ have $\alpha \neq \alpha'$ and they are also easily derivable from the initial clauses $C_1, \ldots, C_m$. As before, we have several cases.

Case $\alpha' = \beta$.

- If $\beta \notin \mathcal{L}_1$, then $\{\neg p_{\alpha,\beta},\ \neg p_{\beta,\beta}\}$ is either $\{\bigvee C_\beta, \neg \bigvee C_\beta\}$ or $\{\bigvee C_\alpha, \neg \bigvee C_\beta\}$ if β is a SPLIT-premise of α. In both cases, these are easy tautologies derivable in small height.
- If $\beta \in \mathcal{L}_1$, then $\{\neg p_{\alpha,\beta},\ \neg p_{\beta,\beta}\}$ is either $\{\bigvee C_\beta,\ \neg(x \vee \neg x)\}$ or $\{\bigvee C_\alpha,\ \neg(x \vee \neg x)\}$ if β is a SPLIT-premise of α. In both cases they are derivable from C_β, a clause that is a weakening of an initial clause from $C_1, \ldots, C_m$, in small height.

Case $\alpha, \alpha' \neq \beta$.

- If $w_\beta = -1$, then $p_{\gamma,\beta} = \bot$ for all but at most two γs. This is so because when we have negative weights, either β is a copy from the previous layer or β part of an appearing sibling. In both cases one of the γs is β so in each axiom of the form $\{\neg p_{\alpha,\beta},\ \neg p_{\alpha',\beta}\}$ either $p_{\alpha,\beta} = \bot$ or $p_{\alpha',\beta} = \bot$ since $\alpha, \alpha' \neq \beta$.
- If $w_\beta = 1$, and β is a disappearing sibling, as in the previous case we have that either $p_{\alpha,\beta} = \bot$ or $p_{\alpha',\beta} = \bot$, and the axiom $\{\neg p_{\alpha,\beta},\ \neg p_{\alpha',\beta}\}$ is trivially derivable. Suppose then both $p_{\alpha,\beta}$ and $p_{\alpha',\beta}$ are distinct from $\bot$. This means in particular that $i_\alpha = i_{\alpha'} = i_\beta + 1$ and β is a premise of both α and α'. That is, at level $\mathcal{L}_{i_\beta}$ we applied a SPLIT rule on β obtaining α, α'. I.e. $\bigvee C_\alpha = \bigvee C_\beta \vee x$ and $\bigvee C_{\alpha'} = \bigvee C_\beta \vee \neg x$ for some variable x. Hence,

$$\{\neg p_{\alpha,\beta},\ \neg p_{\alpha',\beta}\} = \{\neg(\neg \bigvee C_\beta \wedge \neg x),\ \neg(\neg \bigvee C_\beta \wedge x)\}$$
$$= \{\bigvee C_\beta \vee x, \bigvee C_\beta \vee \neg x\}\,,$$

which is a tautology derivable in small height in $\mathsf{Res}(d)$ using two weakenings of $\{x, \neg x\}$ and two $\bigvee$-introductions.

We showed that from the clauses $C_1, \dots, C_m$ in tree-like $\mathsf{Res}(d)$ it is possible to derive all the clauses of the formula $\mathtt{PHP}^{n+1}_n(p_{\alpha,\beta})$, which is a $\mathtt{PHP}^{n+1}_n(G)$ for some graph G of degree at most 3. This concludes the refutation in tree-like $\mathsf{Res}(d) + \mathtt{PHP}^{n+1}_n(G)$. It is a refutation of height 5. □

The construction of the formulas $p_{\alpha,\beta}$ in the previous proof does not satisfy the *onto* axioms but it clearly satisfies the *functionality axioms* of $\mathtt{ofPHP}^{n+1}_n(G)$, which means that the substitution instance of the functionality axioms is a tautology easily derivable. The reason the construction does not satisfy the onto axioms is the following. The last layer $\mathcal{L}_s$ might contain arbitrary weighted clauses $[C_\beta; 1]$ that, if true, are mapped to themselves. Therefore, they receive a pigeon. If they are false, they are mapped to some hole in $\mathcal{L}_{s-1}$, and hence they, as a hole, don't receive a pigeon. Therefore, we have no guarantee that the holes in $\mathcal{L}_s$ receive some pigeon. If $\mathcal{L}_s$ satisfies the condition (3') in the definition of restricted $\mathsf{w\text{-}Resolution}$ (see Definition 3.2), then we can adapt the definition of $p_{\alpha,\beta}$ in the proof of Theorem 4.2 to satisfy the *onto* axioms of the pigeonhole principle.

Theorem 4.3. *For every d, tree-like $\mathsf{Res}(d)+\mathtt{ofPHP}^m_n(G)$ p-simulates degree-d unary NS, where G is restricted to bipartite graphs of degree at most 3 and the height of the tree-like $\mathsf{Res}(d) + \mathtt{ofPHP}^m_n(G)$ derivations is 5. Moreover, for the p-simulation it is enough to have $\mathtt{ofPHP}^m_n(G)$ for $m \leq 2n$.*

Proof. We use the characterization of unary NS given by Theorem 3.5, and we argue basically as in Theorem 4.2. We know that the problematic clauses in $\mathcal{L}_s$ are weakening of initial axioms or several copies of $[\emptyset; 1]$. We can define the formula $p_{\alpha,\beta}$ as in Theorem 4.2 setting $p_{\alpha,\alpha} = x \vee \neg x$ whenever the clause associated to α is a weakening of an initial axiom regardless of the location of α in the proof.

All the copies of $[\emptyset; 1]$ in $\mathcal{L}_s$ are copied to $\mathcal{L}_{s+1}$, as in the case of unary SA. For the argument in SA we only needed to copy one of the $[\emptyset; 1]$, here we need to copy all of them. Hence instead of $\mathtt{PHP}^{n+1}_n(G)$ we use $\mathtt{ofPHP}^m_n(G)$. Of course the number of $[\emptyset; 1]$ we copy in the layer $\mathcal{L}_{s+1}$ cannot be larger than the size of the original proof, therefore $m \leq 2n$ is enough.

A simple case analysis shows that the functionality axioms $\{\neg p_{\alpha,\beta}, \neg p_{\alpha',\beta}\}$ are easily derivable from the initial axioms.

Similarly, the onto axioms $\{p_{\gamma,\beta} : \gamma \in P\}$ are also immediate to derive. For sake of clarity we highlight the less trivial case: when β is a SPLIT-premise and one of its conclusions is a weakening of an initial axiom. That is suppose the clause associated with β is C_β, the conclusions are α and α', with $C_\alpha = C_\beta \vee x$ and $C_{\alpha'} = C_\beta \vee \neg x$ and α is the one which is a weakening of an initial axiom. That is

$$\{p_{\gamma,\beta} : \gamma \in P\} = \{p_{\alpha,\beta}, p_{\alpha',\beta}, p_{\beta,\beta}\} = \{\bot, \neg \bigvee C_\beta \wedge x, \bigvee C_\beta = \{\neg \bigvee C_\beta \wedge x, \bigvee C_\beta\}\,.$$

We assumed that $\{C_\beta, x\}$ is a WEAKENING of an initial axiom, then from it we can derive by $\vee$-INTRODUCTION, $\{\bigvee C_\beta, x\}$, by EXCLUDED MIDDLE, $\{\bigvee C_\beta, \neg \bigvee C_\beta\}$ and finally by $\wedge$-INTRODUCTION $\{\bigvee C_\beta, x \wedge \neg \bigvee C_\beta\}$. □

The proof of Theorem 4.2 will generalize, almost without changes, if instead of clauses we consider Θ_d-cedents.

Theorem 4.4. *For every $d \in \mathbb{N}$, tree-like $\mathsf{Frege}_{d+1} + \mathtt{PHP}^{n+1}_n(G)$ p-simulates unary $\mathsf{w}\text{-}\mathsf{Frege}_d$, where G is restricted to bipartite graphs of degree at most 3.*

Proof. The rules that could be applied in $\mathsf{w}\text{-}\mathsf{Frege}_d$ that could have not been applied in w-Resolution are the $\bigwedge$-INTRODUCTION and the $\bigvee$-INTRODUCTION. This last rule is not problematic, it has only one premise and one conclusion. As in the proof of Theorem 4.2, if the conclusion is true, the pigeon corresponding to it is mapped to the hole corresponding to the conclusion, if it is false, it is mapped to the hole corresponding to the premise.

The rule that might create problems is the $\bigwedge$-INTRODUCTION rule, i.e.

$$\frac{[\Gamma, \phi_1; w] \quad [\Gamma, \phi_2; w]}{[\Gamma, \phi_1 \wedge \phi_2; w] \quad [\Gamma, \phi_1, \phi_2; w]}\,. \tag{12}$$

Then, for every instance of the binary $\bigwedge$-INTRODUCTION if the conclusion of the inference is true, the pigeon corresponding to it is mapped to the hole corresponding to the conclusion. If one of the conclusions of the inference is false, the pigeon corresponding to it is mapped to a false premise. More precisely, fix an instance of the rule in (12) and let α_1 be the pigeon corresponding to the left conclusion, α_2 be the pigeon corresponding to the right conclusion, β_1 the hole corresponding to the left premise and β_2 the hole corresponding to the right premise. Then the formula p_{α_1,β_1} is

$$\neg\Gamma \wedge (\neg\phi_1 \vee \neg\phi_2) \wedge \neg\phi_1\,,$$

and the formula p_{α_1,β_2} is

$$\neg\Gamma \wedge (\neg\phi_1 \vee \neg\phi_2) \wedge \phi_1 \wedge \neg\phi_2\,.$$

The formula $p_{\alpha_2,\beta_1} = \bot$ and the formula p_{α_2,β_2} is

$$\neg \bigvee \Gamma \wedge \neg\phi_1 \wedge \neg\phi_2\,.$$

This is the only simple non-trivial change needed on top of the trivial generalization of the definition of the formula $p_{\alpha,\beta}$ from clauses to Θ_d-cedents. The fact that in the rule (12) the injectivity of the mapping is not violated follows from the soundness of the rule, that the number of false premises equals the number of false conclusions. □

Notice that, by the form of the $\bigwedge$-INTROD. rule, the functionality axioms of the $\mathtt{ofPHP}^{n+1}_n(G)$ would also be satisfied. This would not be the case for the standard generalization of the $\bigwedge$-INTROD. to the context of weighted cedents. We conclude this section with a couple of separations and lower-bounds.

Proposition 4.5. *For every $d = o\left(\frac{\log\log n}{\log\log\log n}\right)$, Frege_d does not p-simulate unary $\mathsf{w}\text{-}\mathsf{Frege}_d$.*

Proof. Any refutation of $\mathtt{PHP}^{n+1}_n$ in Frege_d must have size at least $2^{n^{(1/6)^d}}$ (see for instance [33]). $\mathtt{PHP}^{n+1}_n$ has polynomial size unary SA refutations, and hence it has polynomial size refutations in unary $\mathsf{w}\text{-}\mathsf{Frege}_d$. □

Definition 4.6 *(MOD_2 principle).* Given $n \in \mathbb{N}$, the *MOD_2-principle* is the set of cedents in the variables $x_{i,j}$ for $i \neq j \in S$

$$\begin{aligned}\mathtt{MOD}^n_2 &= \{\{x_{i,1}, \ldots, x_{i,i-1}, x_{i,i+1}, \ldots, x_{i,2n+1}\} : i \in [2n+1]\}\\ &\cup \{\{\neg x_{i,j}, \neg x_{i',j}\} : i, i' \in [2n+1] \text{ distinct}, j \in [2n+1]\}\,.\end{aligned}$$

Corollary 4.7. *Given* $n \in \mathbb{N}$ *and* $d = o\left(\frac{\log\log n}{\log\log\log n}\right)$, $\mathtt{MOD}_2^n$ *has no polynomial-size unary* w-Frege$_d$ *refutations.*

Proof. Any refutation of the $\mathtt{MOD}_2^n$ principle in $\mathsf{Frege}_d + \mathtt{PHP}$ must require size at least $\exp(n^{\Omega\left(1/d4^d\right)})$ [6, Theorem 4]. By Theorem 4.4, $\mathsf{Frege}_{d+1} + \mathtt{PHP}$ p-simulates unary w-Frege$_d$. The lower bound follows: the formula $\mathtt{MOD}_2^n$ requires unary w-Frege$_d$ refutations of size $\exp(n^{\Omega\left(1/(d+1)4^d\right)})$. □

Definition 4.8 *(*bit-*pigeonhole principle).* Let $n = 2^k$. The formula $\mathtt{bit\text{-}PHP}_n$ has variables $b_{i\ell}$ for each $i \in [n+1]$ and $\ell \in [k]$. The variables $b_{i1}, \dots, b_{ik}$ represent the binary expansion of a hole, the hole i is mapped to. Then $\mathtt{bit\text{-}PHP}_n$ only needs to enforce injectivity. The formula $\mathtt{bit\text{-}PHP}_n$ is

$$\left\{ \{b_{i1}^{1-h_1}, \dots, b_{ik}^{1-h_k}, b_{i'1}^{1-h_1}, \dots, b_{i'k}^{1-h_k}\} : \begin{array}{c} i \neq i' \in [n+1] \\ h \in [n] \end{array} \right\},$$

where $h_1, \dots, h_k$ is the binary representation of the hole h and $b_{ij}^{h_j} = b_{ij}$ if $h_j = 1$ and $b_{ij}^{h_j} = \neg b_{ij}$ if $h_j = 0$.

Theorem 4.9. SA *does not p-simulate tree-like* $\mathsf{Frege}_1 + \mathtt{PHP}_n^{n+1}$.

Proof. $\mathtt{bit\text{-}PHP}_n$ does not have polynomial-size SA refutations [15]. On the other hand, $\mathtt{bit\text{-}PHP}_n$ has polynomial size tree-like $\mathsf{Frege}_1 + \mathtt{PHP}_n^{n+1}$ refutations. To see this we use the substitution $p_{ij} = b_{i1}^{j_1} \wedge \dots \wedge b_{ik}^{j_k}$ where $j_1, \dots, j_k$ is the binary representation of j. For $i \neq i' \in [n+1]$ and $j \in [n]$, the cedent $\{\neg p_{ij}, \neg p_{i'j}\}$ is immediately derivable from the axioms of $\mathtt{bit\text{-}PHP}_n$ by $\bigvee$-INTRODUCTION. For every $i \in [n+1]$, the cedent $\{p_{i1}, \dots, p_{in}\}$ is tautological, and it has $k = \log n$ variables. By EXCLUDED MIDDLE, derive all the $\{p_{ij}, \neg p_{ij}\}$ and then with WEAKENING and 2^k applications of SYMM. CUT it is easy to obtain $\{p_{i1}, \dots, p_{in}\}$. □

5. The *leveled* pigeonhole principle and Sherali-Adams

In this section, we generalize the constructions given for unary SA/NS and unary w-Frege$_d$ to systems with binary weights/coefficients. We prove all remaining p-simulations in Fig. 1, 2 and 3.

The starting point of this section is that, it is not clear at all whether it is possible to adapt Theorem 4.4 to show that tree-like $\mathsf{Frege}_1 + \mathtt{PHP}_n^{n+1}(G)$ p-simulates SA. It seems we need a stronger version of the pigeonhole principle. For this reason, we introduce a new combinatorial principle, the *leveled* PHP.

The leveled pigeonhole principle maps $n^2 + 1$ pigeons into n^2 holes. First, we partition both sets of pigeons and holes into n parts. The partition of the holes consists of n sets $H_1, \dots, H_n$ given by $H_\ell = \{(\ell-1)n + 1, \dots, \ell n\}$. Let $H_0 = H_{n+1} = \emptyset$. For the partition of the pigeons we set, for some $j \in [n]$, $P_j = H_j \cup \{n^2 + 1\}$ and for the remaining $\ell \in [n] \setminus \{j\}$, $P_\ell = H_\ell$. Let $P_0 = P_{n+1} = \emptyset$.

Definition 5.1 *(Leveled pigeonhole principle,* LPHP*).* The *leveled* pigeonhole principle[3] has variables x_{ij} for each $i \in [n^2+1]$ and each $j \in [n^2]$. The formula $\mathtt{LPHP}_{n^2}^{n^2+1}$ has the following clauses. For every $\ell \in [n]$, every pigeon $p \in P_\ell$ we have clauses

$$\{x_{p1}, \dots, x_{pn^2}\}, \tag{13}$$

$$\{\neg x_{pj}\} \quad \text{for all } j \notin H_{\ell-1} \cup H_\ell \cup H_{\ell+1}, \tag{14}$$

$$\{\neg x_{pj},\ x_{pj'} : j' \in H_{\ell-1} \setminus \{j\}\} \ \text{ for all } \ j \in H_{\ell-1}, \tag{15}$$

$$\{\neg x_{pj_1},\ \neg x_{pj_2},\ \neg x_{pj_3}\} \ \text{ for all } \ \text{distinct } j_1, j_2, j_3 \in H_{\ell-1} \tag{16}$$

[3] In [9], the preliminary version of his work, the *leveled* pigeonhole principle was called *weighted* pigeonhole principle, in short wPHP. To avoid confusion with the *weak* pigeonhole principle, we use the term *leveled*.

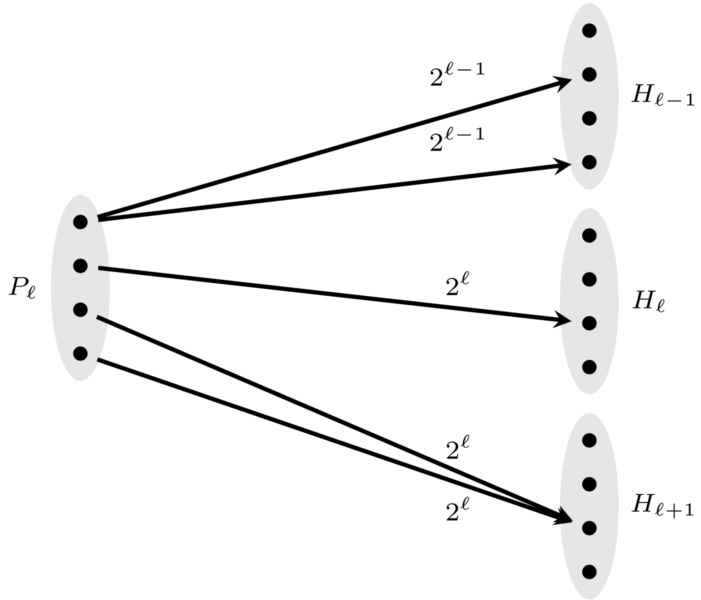

Fig. 6. Possible ways pigeons in P_ℓ can fly.

and every hole $h \in H_\ell$, we have clauses

$$\{\neg x_{ih},\ \neg x_{i'h}\} \quad \text{for all distinct } i \in P_\ell \cup P_{\ell+1} \text{ and } i' \in [n^2+1]\,, \tag{17}$$

$$\{\neg x_{i_1h},\ \neg x_{i_2h},\ \neg x_{i_3h}\} \quad \text{for all distinct } i_1, i_2, i_3 \in P_{\ell-1}\,. \tag{18}$$

Intuitively, $p \in P_\ell$ means that p has mass 2^ℓ, and $h \in H_\ell$ means h has capacity 2^ℓ, see Fig. 6. The pigeon $p \in P_\ell$ has to fly somewhere (eq. (13)) and moreover, it can only fly to holes in $H_{\ell-1}$ or H_ℓ or $H_{\ell+1}$ (eq. (14)). The pigeon has to fly with either full or half-mass. If $p \in P_\ell$ flies to $H_{\ell-1}$, it flies with half-mass and hence it should fly to two distinct holes in $H_{\ell-1}$ (eq. (15)) but not to three holes in $H_{\ell-1}$ (eq. (16)). If $p \in P_\ell$ flies to H_ℓ, we assume it flies with full mass, hence completely filling the capacity of a hole in H_ℓ (eq. (17)). If $p \in P_\ell$ flies to $H_{\ell+1}$, we also assume it flies with full mass, but now it only fills half of the capacity of a hole in $H_{\ell+1}$. Therefore, to fill the capacity of a hole $h \in H_{\ell+1}$ we will need another pigeon from P_ℓ flying to h but not two more (eq. (18)).

The intended meaning of the variable x_{ij} for $i \in P_\ell$ is: for $j \in H_\ell \cup H_{\ell+1}$, $x_{ij} = 1$ means "*the pigeon i flies to j with mass 2^ℓ*"; for $j \in H_{\ell-1}$, $x_{ij} = 1$ means "*i flies to j with mass $2^{\ell-1}$*". If $j \notin H_{\ell-1} \cup H_\ell \cup H_{\ell+1}$ then $x_{ij} = \bot$.

Definition 5.2 *(*Onto-functional `LPHP`, `ofLPHP`*).* Let $m \le n$ and allow $P_j \neq H_j$ for m indices $j_1, \dots, j_m$ where we have $P_{j_\ell} = H_{j_\ell} \cup \{n^2 + \ell\}$. If we add to the axioms of $\mathtt{LPHP}^{n^2+1}_{n^2}$ the following *onto-functional* axioms, we obtain the formula $\mathtt{ofLPHP}^{n^2+m}_{n^2}$. The axioms we add are: for every $\ell \in [n]$ and every pigeon $p \in P_\ell$ the clauses

$$\{\neg x_{pj},\ \neg x_{pj'}\} \quad \text{for all distinct } j \in H_\ell \cup H_{\ell+1} \text{ and } j' \in [n^2]\,,$$

and every hole $h \in H_\ell$ the clauses

$$\{x_{1h}, \ldots, x_{n^2+m,h}\},$$
$$\{\neg x_{ih},\ x_{i'h} : i' \in P_{\ell-1} \setminus \{i\}\} \text{ for all } i \in P_{\ell-1}.$$

Definition 5.3 *(*graph LPHP *and* ofLPHP*).* Similar to the PHP case, given a bipartite graph $G = (P \sqcup H, E)$ with vertices the disjoint union of P with size n^2+1 and H with size n^2, the *graph* leveled pigeonhole principle $\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ is the formula $\mathtt{LPHP}^{n^2+1}_{n^2}$ restricted by the partial truth assignment mapping all the variables $x_{i,j}$ for $(i,j) \notin E$ to false ($\bot$).

Given a bipartite graph $G = (P \sqcup H, E)$ with $|P| = n^2+m$ and $|H| = n^2$, the *graph* onto-functional leveled pigeonhole principle $\mathtt{ofLPHP}^{n^2+m}_{n^2}(G)$ is the formula $\mathtt{ofLPHP}^{n^2+m}_{n^2}$ restricted by the partial truth assignment mapping all the variables $x_{i,j}$ for $(i,j) \notin E$ to false ($\bot$).

Remark 5.4. The clauses in eq. (16) are not needed to have an unsatisfiable formula but they are useful to have a short proof in SA. When considering $\mathtt{LPHP}^{n^2+1}_{n^2}(G)$, the graphs G we need to consider, turn out to always have at most 2 edges of the form $(p,j),(p,j')$ with $p \in P_\ell$ and $j,j' \in H_{\ell-1}$. Hence, for those graphs G, the axioms in eq. (16) are always satisfied: one of the variables $x_{pj_1},\ x_{pj_2},\ x_{pj_3}$ is always set to $\bot$.

Remark 5.5. We defined $\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ for specific fixed partitions $H_1,\ldots H_n$, and $P_1,\ldots,P_n$, all of size n except for one P_j of size $n+1$. We could also allow $P_1,P_2,\ldots,P_n$ to be disjoint sets of size possibly smaller than n (at most $n+1$ for one P_j). This would not give a more general definition of $\mathtt{LPHP}^{n^2+1}_{n^2}$, as long as for every $\ell \in [n]$, $H_\ell = P_\ell \setminus \{n^2+1\}$. Basically, we could add some padding to all P_js and H_js, until they have size n and change G to a graph that forces the new vertices in each part P_j to be mapped to the corresponding new vertex in H_j. In Theorem 5.11 we will use the $\mathtt{LPHP}^{n^2+1}_{n^2}$ with partition sets possibly smaller than n and we will not use the padding.

It may be not immediately clear why $\mathtt{LPHP}^{n^2+1}_{n^2}$ is unsatisfiable. Informally, a way to see this is to notice that for every pigeon p (say $p \in P_\ell$) the axioms of $\mathtt{LPHP}^{n^2+1}_{n^2}$ can be interpreted to state that the weight flying away from p is at least 2^ℓ and, for every hole h (say $h \in H_\ell$), the weight it can accommodate is at most 2^ℓ. So the holes can, in total, accommodate a total weight of at most $\sum_{\ell\in[n]} n2^\ell$ which is strictly smaller than the total weight of the pigeons flying, that is $2^j + \sum_{\ell\in[n]} n2^\ell$ for some $j \in [n]$.

5.1. Upper bounds of $\mathtt{LPHP}^{n^2+1}_{n^2}$/$\mathtt{ofLPHP}^{n^2+m}_{n^2}$ *in Sherali-Adams/Nullstellensatz*

In this section we show how to prove in SA the unsatisfiability of $\mathtt{LPHP}^{n^2+1}_{n^2}$ and the unsatisfiability of $\mathtt{ofLPHP}^{n^2+1}_{n^2}$ in NS.

We use the following notation. Given two polynomials p and q we write $p \equiv q$ if $p - q$ is a polynomial in the ideal generated by the Boolean axioms and it has polynomial size. The *Boolean axioms* in the ring of polynomials $\mathbb{Z}[x_1,\ldots,x_n,\bar{x}_1,\ldots,\bar{x}_n]$ are the polynomials $x_i^2 - x_i$ and $x_i + \bar{x}_i - 1$ for $i \in [n]$, and the ideal generated by them is the set of all polynomials $p \in \mathbb{Z}[x_1,\ldots,x_n,\bar{x}_1,\ldots,\bar{x}_n]$ of the form $p = \sum_{i=1}^n q_i(x_i^2 - x_i) + q_i'(x_i + \bar{x}_i - 1)$ for some polynomials $q_i, q_i' \in \mathbb{Z}[x_1,\ldots,x_n,\bar{x}_1,\ldots,\bar{x}_n]$.

It is immediate to see that $\equiv$ is an equivalence relation. Moreover, given polynomials p,q,p',q',

- if M is a monomial, and $p \equiv q$ then $Mp \equiv Mq$, and
- if $p \equiv q$ and $p' \equiv q'$, then $p + p' \equiv q + q'$.

Lemma 5.6. *Let $N \in \mathbb{N}$ and $x_1,\ldots,x_N,\bar{x}_1,\ldots,\bar{x}_N$ be some generic variables. Let $X_0 = 1$, $X_i = \prod_{\ell\in[i]} \bar{x}_i$ for $i \in [N]$ and let $\widetilde{x}_i$ be the polynomial $x_i + \bar{x}_i - 1$. By a telescopic sum we have the following algebraic equality*

$$X_i = 1 + \sum_{j\in[i]} (X_{j-1}\widetilde{x_j} - X_{j-1}x_j)\,, \tag{19}$$

that is

$$X_i \equiv 1 - \sum_{j\in[i]} X_{j-1}x_j\,, \tag{20}$$

$$X_N \equiv 1 - \sum_{\ell\in[N]} x_\ell + \sum_{\ell\in[N]}\sum_{k\in[\ell-1]} x_\ell x_k - \sum_{\ell\in[N]}\sum_{k\in[\ell-1]}\sum_{z\in[k-1]} x_\ell x_k x_z X_{z-1} \tag{21}$$

$$\sum_{i\in[N]} x_i \equiv 1 - X_N + \sum_{i\in[N]}\sum_{j\in[i-1]} X_{j-1}x_j x_i\,. \tag{22}$$

Proof. Eq. (19) follows from the fact the RHS is just a telescopic sum. Eq. (20) follows from eq. (19) and the definition of $\equiv$. To obtain eq. (21) we use eq. (19) in the following chain of equalities:

$$\begin{aligned}
X_N &= 1 + \sum_{\ell\in[N]} (X_{\ell-1}\widetilde{x_\ell} - x_\ell X_{\ell-1}) \\
&\overset{\text{eq. (19)}}{=} 1 + \sum_{\ell\in[N]} \Big(X_{\ell-1}\widetilde{x_\ell} - x_\ell\Big(1 + \sum_{k\in[\ell-1]} (X_{k-1}\widetilde{x_k} - X_{k-1}x_k)\Big)\Big) \\
&= 1 - \sum_{\ell\in[N]} x_\ell + \sum_{\ell\in[N]}\sum_{k\in[\ell-1]} x_\ell x_k X_{k-1} + \sum_{\ell\in[N]} \Big(X_{\ell-1}\widetilde{x_\ell} - \sum_{k\in[\ell-1]} x_\ell X_{k-1}\widetilde{x_k}\Big) \\
&\overset{\text{eq. (19)}}{=} 1 - \sum_{\ell\in[N]} x_\ell + \sum_{\ell\in[N]}\sum_{k\in[\ell-1]} x_\ell x_k \Big(1 + \sum_{z\in[k-1]} (X_{z-1}\widetilde{x_z} - x_z X_{z-1})\Big) \\
&\qquad + \sum_{\ell\in[N]} \Big(X_{\ell-1}\widetilde{x_\ell} - \sum_{k\in[\ell-1]} x_\ell X_{k-1}\widetilde{x_k}\Big) \\
&= 1 - \sum_{\ell\in[m]} x_\ell + \sum_{\ell\in[m]}\sum_{k\in[\ell-1]} x_\ell x_k - \sum_{\ell\in[m]}\sum_{k\in[\ell-1]}\sum_{z\in[k-1]} x_\ell x_k x_z X_{z-1} \\
&\qquad + \sum_{\ell\in[m]} \Big(X_{\ell-1}\widetilde{x_\ell} - \sum_{k\in[\ell-1]} \Big(x_\ell X_{k-1}\widetilde{x_k} - \sum_{z\in[k-1]} x_\ell x_k X_{z-1}\widetilde{x_z}\Big)\Big) \\
&\equiv 1 - \sum_{\ell\in[m]} x_\ell + \sum_{\ell\in[m]}\sum_{k\in[\ell-1]} x_\ell x_k - \sum_{\ell\in[m]}\sum_{k\in[\ell-1]}\sum_{z\in[k-1]} x_\ell x_k x_z X_{z-1}\,.
\end{aligned}$$

To prove eq. (22) notice that eq. (19) immediately implies the algebraic equality

$$\begin{aligned}
\sum_{i\in[N]} x_i &= 1 - X_N + \sum_{i\in[N]} (X_{i-1}\widetilde{x_i} + x_i(1 - X_{i-1})) \\
&= 1 - X_N + \sum_{i\in[N]} (X_{i-1}\widetilde{x_i} + \sum_{j\in[i-1]} (X_{j-1}x_j x_i - X_{j-1}x_i\widetilde{x_j})) \\
&\equiv 1 - X_N + \sum_{i\in[N]}\sum_{j\in[i-1]} X_{j-1}x_j x_i\,. \quad \square
\end{aligned}$$

Lemma 5.7. *Given variables $a_1, \ldots, a_n, \bar{a}_1, \ldots, \bar{a}_n$ and $b_1, \ldots, b_m, \bar{b}_1, \ldots, \bar{b}_n$, there is a polynomial-size* SA *derivation of the inequality*

$$2\sum_{i\in[n]} a_i + \sum_{j\in[m]} b_j - 2 \geq 0 \tag{23}$$

from the Boolean axioms $a_i + \bar{a}_i - 1 = 0$ and $b_i + \bar{b}_i - 1 = 0$ and the axioms

$$\begin{cases} -b_i b_j b_k \geq 0 & \text{for all distinct } i,j,k \in [m]\,, \\ -\prod_{i\in[n]} \bar{a}_i \prod_{j\in[m]} \bar{b}_j \geq 0\,, & \\ -b_\ell \prod_{j\in[m]\setminus\{\ell\}} \bar{b}_j \geq 0 & \text{for all } \ell \in [m]\,. \end{cases}$$

There is a polynomial-size SA *derivation of the inequality*

$$2 - 2\sum_{i\in[n]} a_i - \sum_{j\in[m]} b_j \geq 0\,. \tag{24}$$

from the Boolean axioms, and the axioms

$$\begin{cases} -a_i a_j \geq 0 & \text{for all distinct } i,j \in [n]\,, \\ -a_i b_j \geq 0 & \text{for all } i \in [n], j \in [m]\,, \\ -b_i b_j b_k \geq 0 & \text{for all distinct } i,j,k \in [m]\,. \end{cases}$$

Proof. Let $A_0 = 1$, $B_0 = 1$, for every $j \in [m]$ $B_{j,0} = 1$ and, for every $k \in [n]$, let $A_k = \prod_{\ell\in[k]} \bar{a}_\ell$, for every $i \in [m]$ let $B_i = \prod_{\ell\in[i]} \bar{b}_\ell$ and $B_{j,i} = \prod_{\ell\in[i]\setminus\{j\}} \bar{b}_\ell$. Lemma 5.6, with these notations, gives the following equivalences:

$$B_m \equiv 1 - \sum_{j\in[m]} B_{j-1} b_j\,, \tag{25}$$

$$B_m \equiv 1 - \sum_{\ell\in[m]} b_\ell + \sum_{\substack{\ell,k\in[m]\\ k<\ell}} b_\ell b_k - \sum_{\substack{\ell,k,z\in[m]\\ z<k<\ell}} b_\ell b_k b_z B_{z-1} \tag{26}$$

$$\sum_{i\in[n]} a_i \equiv 1 - A_n + \sum_{\substack{i,j\in[n]\\ j<i}} A_{j-1} a_j a_i\,, \tag{27}$$

$$\sum_{j\in[m]\setminus\{i\}} b_j \equiv 1 - B_{i,m} + \sum_{j\in[m]\setminus\{i\}} \sum_{k\in[j-1]\setminus\{i\}} B_{i,j-1} b_j b_k\,. \tag{28}$$

Multiplying (28) by b_i and summing for every $i \in [m]$, we get

$$2\sum_{\substack{i,j\in[m]\\ i<j}} b_i b_j \equiv \sum_{i\in[m]} b_i - \sum_{i\in[m]} b_i B_{i,m} + \sum_{\substack{i,j,k\in[m]\\ i,j,k \text{distinct}}} B_{i,j-1} b_i b_j b_k\,, \tag{29}$$

or, equivalently,

$$2\sum_{\substack{i,j\in[m]\\ i<j}} b_i b_j - \sum_{i\in[m]} b_i \equiv -\sum_{i\in[m]} b_i B_{i,m} + \sum_{\substack{i,j,k\in[m]\\ i,j,k \text{distinct}}} B_{i,j-1} b_i b_j b_k\,, \tag{30}$$

Multiplying (27) by B_m, we get

$$\sum_{i\in[n]} a_i B_m \equiv B_m - A_n B_m + \sum_{i\in[n]} \sum_{j\in[i-1]} A_{j-1} a_j a_i B_m\,. \tag{31}$$

Now substitute for B_m in the LHS using (25) and for the first B_m on the RHS using (26). This gives,

$$\sum_{i\in[n]} a_i - \sum_{i\in[n] j\in[m]} B_{j-1}a_ib_j \equiv 1 - \sum_{\ell\in[m]} b_\ell + \sum_{\substack{\ell,k\in[m]\\ k<\ell}} b_\ell b_k - \sum_{\substack{\ell,k,z\in[m]\\ z<k<\ell}} b_\ell b_k b_z B_{z-1}$$
$$- A_nB_m + \sum_{i\in[n]}\sum_{j\in[i-1]} A_{j-1}a_ja_iB_m\,.$$

Which, after reordering the terms, is

$$\sum_{i\in[n]} a_i + \sum_{\ell\in[m]} b_\ell - \sum_{\substack{\ell,k\in[m]\\ k<\ell}} b_\ell b_k - 1 \equiv - \sum_{i\in[n] j\in[m]} B_{j-1}a_ib_j - \sum_{\substack{\ell,k,z\in[m]\\ z<k<\ell}} b_\ell b_k b_z B_{z-1}$$
$$- A_nB_m + \sum_{i\in[n]}\sum_{j\in[i-1]} A_{j-1}a_ja_iB_m\,. \tag{32}$$

To conclude it is enough then to sum eq. (30) and twice (32), this gives

$$2\sum_{i\in[n]} a_i + \sum_{\ell\in[m]} b_\ell - 2 \equiv -2 \sum_{i\in[n] j\in[m]} B_{j-1}a_ib_j - 2\sum_{\substack{\ell,k,z\in[m]\\ z<k<\ell}} b_\ell b_k b_z B_{z-1}$$
$$- 2A_nB_m + 2\sum_{\substack{i,j\in[n]\\ j<i}} A_{j-1}a_ja_iB_m$$
$$- \sum_{i\in[m]} b_iB_{i,m} + \sum_{\substack{i,j,k\in[m]\\ i,j,k \text{distinct}}} B_{i,j-1}b_ib_jb_k\,. \tag{33}$$

This last equivalence is essentially a SA derivation (modulo the Boolean axioms) of eq. (23) from the desired axioms. Multiplying eq. (33) by -1 it becomes a SA derivation (modulo the Boolean axioms) of eq. (24) from the desired axioms. □

Lemma 5.8. *Given variables $a_1,\dots,a_n,\bar a_1,\dots,\bar a_n$ and $b_1,\dots,b_m,\bar b_1,\dots,\bar b_n$, there is a polynomial-size* NS *derivation of the equality*

$$2\sum_{i\in[n]} a_i + \sum_{j\in[m]} b_j = 2 \tag{34}$$

from the Boolean axioms $a_i+\bar a_i-1=0$ and $b_i+\bar b_i-1=0$ and the axioms

$$\begin{cases} b_ib_jb_k = 0 & \text{for all distinct } i,j,k\in[m]\,,\\ \prod_{i\in[n]}\bar a_i\prod_{j\in[m]}\bar b_j = 0 & \\ b_\ell\prod_{j\in[m]\setminus\{\ell\}}\bar b_j = 0 & \text{for all } \ell\in[m]\\ a_ia_j = 0 & \text{for all distinct } i,j\in[n]\,,\\ a_ib_j = 0 & \text{for all } i\in[n], j\in[m]\,. \end{cases}$$

Proof. Immediate from eq. (33) in the previous Lemma. □

Theorem 5.9. *The formula $\mathtt{LPHP}^{n^2+1}_{n^2}$ has polynomial-size* SA *refutations. Also, for every bipartite graph $G=(P\sqcup H,E)$ with $|P|=n^2+1$, $|H|=n^2$ and degree d, $\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ has* SA*-refutations of degree d.*

Proof. First use Lemma 5.7. The variables a_i correspond to the variables x_{ij} for $i\in P_\ell$ and $j\in H_\ell\cup H_{\ell+1}$. The variables b_i correspond to the variables x_{ij} with $i\in P_\ell$ and $j\in H_{\ell-1}$.

By Lemma 5.7 we have that the axioms of $\texttt{LPHP}^{n^2+1}_{n^2}$ imply, for every $i \in [n^2+1]$ with $i \in P_\ell$, the inequality

$$2\sum_{j\in H_\ell\cup H_{\ell+1}} x_{ij} \;+ \sum_{j\in H_{\ell-1}} x_{ij} - 2 \geq 0\,, \tag{35}$$

and, for each $j \in [n^2]$ with $j \in H_\ell$, the inequality

$$2 - 2\sum_{i\in P_\ell\cup P_{\ell+1}} x_{ij} \;- \sum_{i\in P_{\ell-1}} x_{ij} \geq 0\,. \tag{36}$$

Eq. (35) says that the pigeon i must fly at least once into the set $H_\ell \cup H_{\ell+1}$ or at least twice into the set $H_{\ell-1}$.

Eq. (36) says that the hole j can receive at most one pigeon from the set $P_\ell \cup P_{\ell+1}$, or at most two pigeons from $P_{\ell-1}$.

To conclude, we want to sum appropriate multiples of eq. (35) and eq. (36), in a way that all variables from (35) cancel with variables in (36), and after all the cancellations we just get some negative constant:

$$\sum_{\substack{\ell\in[n]\\ i\in P_\ell}} 2^\ell \left(2\sum_{j\in H_\ell\cup H_{\ell+1}} x_{ij} \;+ \sum_{j\in P_{\ell-1}} x_{ij} - 2\right) + \sum_{\substack{\ell\in[n]\\ j\in H_\ell}} 2^\ell \left(2 - 2\sum_{i\in P_\ell\cup P_{\ell+1}} x_{ij} \;- \sum_{i\in P_{\ell-1}} x_{ij}\right) \geq 0\,. \tag{37}$$

Consider a variable x_{ij} in (37), with $i \in P_\ell$.

If $j \in H_\ell$, the coefficient of x_{ij} is $2^\ell \cdot 2 - 2^\ell \cdot 2 = 0$.

If $j \in H_{\ell+1}$, the coefficient of x_{ij} is $2^\ell \cdot 2 - 2^{\ell+1} = 0$.

If $j \in H_{\ell-1}$, the coefficient of x_{ij} is $2^\ell - 2 \cdot 2^{\ell-1} = 0$.

That is, all the variables x_{ij} cancel out in (37). The constants in (37) sum to

$$-2\sum_{\substack{\ell\in[n]\\ i\in P_\ell}} 2^\ell + 2\sum_{\substack{\ell\in[n]\\ j\in H_\ell}} 2^\ell = -2^{j+1}\,,$$

if the pigeon n^2+1 was in the set P_j, since $|P_\ell| = |H_\ell|$ for all ℓ except for j where $|P_j| = |H_j| + 1$. That is, the sum in (37), after cancellations, reduces to the trivial contradiction $-2^{j+1} \geq 0$. □

Via a similar argument, it is easy to see that $\texttt{ofLPHP}^{n^2+m}_{n^2}$ has polynomial-size NS refutations.

Theorem 5.10. *The formula* $\texttt{ofLPHP}^{n^2+m}_{n^2}$ *has polynomial-size* NS *refutations. Also, for every bipartite graph* $G = (P \sqcup H, E)$ *with* $|P| = n^2 + m$, $|H| = n^2$ *and degree* d, $\texttt{ofLPHP}^{n^2+m}_{n^2}(G)$ *has* NS*-refutations of degree* d.

Proof. We proceed as in Theorem 5.9 but we use Lemma 5.8 instead of Lemma 5.7. We obtain

$$\sum_{\substack{\ell\in[n]\\ i\in P_\ell}} 2^\ell \left(2\sum_{j\in H_\ell\cup H_{\ell+1}} x_{ij} \;+ \sum_{j\in P_{\ell-1}} x_{ij} - 2\right)$$

$$+\sum_{\substack{\ell\in[n]\\ j\in H_\ell}} 2^\ell \left(2-2\sum_{i\in P_\ell\cup P_{\ell+1}} x_{ij} \;-\; \sum_{i\in P_{\ell-1}} x_{ij}\right)=0\,,$$

which, after the same semplifications in the previous theorem reduces to the trivial contradiction

$$0=-2\sum_{\substack{\ell\in[n]\\ i\in P_\ell}} 2^\ell + 2\sum_{\substack{\ell\in[n]\\ j\in H_\ell}} 2^\ell = -\sum_{\ell\in[m]} 2^{j_\ell+1}\,. \quad \square$$

It is also easy to see that $\mathsf{Frege}_1 + \mathtt{LPHP}$ proves $\mathtt{PHP}$ in polynomial size. We don't know whether the opposite is true, but we suspect it is not (see Section 6), even using higher constant depth. This would imply not only that $\mathtt{LPHP}^{n^2+1}_{n^2}$ is hard to refute in unary SA, via Theorem 4.2, but even in unary $\mathsf{w\text{-}Frege}_d$, via Theorem 4.4.

5.2. p-Simulations

We now prove the remaining p-simulations from Fig. 1, Fig. 2, and Fig. 3.

Theorem 5.11. *For every $d\in\mathbb{N}$, the proof system tree-like $\mathsf{Res}(d)+\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ p-simulates degree-d SA, where G is restricted to bipartite graphs of degree at most 3 and the tree-like $\mathsf{Res}(d)+\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ derivations have height 5.*

Proof. The structure of the proof is similar to the proof of Theorem 4.2. By Theorem 3.5 it is enough to prove the result for $\mathsf{w\text{-}Resolution}$. Let $\pi=\mathcal{L}_1,\dots,\mathcal{L}_s$ be a weighted $\mathsf{Resolution}$ refutation of a set of clauses $\{C_1,\dots,C_m\}$. W.l.o.g. we can assume that no weighted cedent in π has weight 0 and, by Remark 3.3, we can assume that all the weights appearing in π are powers of 2, and all the rules have positive weights, except for INTRODUCTION/REMOVAL. Moreover, since π is a refutation, we can assume $[\emptyset;1]\in\mathcal{L}_s$. If the last layer of π had $[\emptyset;2^z]$ for some $z\geq 0$, we can obtain a new last layer containing $[\emptyset;1]$, using the UNFOLD rule.

We define a substitution instance of $\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ without padding (see Remark 5.5) such that we have small-depth $\mathsf{Res}(d)$ derivations of it.

Let $\mathcal{S}+1$ be the size of π, let $\mathcal{L}_{s+1}=\{[\emptyset;1]\}$ and let $P_1,\dots,P_{\mathcal{S}}$ be a partition of the multiset $\mathcal{L}_1\cup\dots\cup\mathcal{L}_{s+1}$ according to the weights of the weighted clauses, i.e. all the weighted clauses in P_j have weight 2^{j-1} or -2^{j-1}. By assumption, all those multisets have size at most $\mathcal{S}$, except P_1 that has size at most $\mathcal{S}+1$. Let $P_0=P_{\mathcal{S}+1}=\emptyset$. Let $H_1,\dots,H_{\mathcal{S}}$ be defined as $H_1=P_1\setminus\mathcal{L}_{s+1}$, and for all $\ell\in\{2,\dots,\mathcal{S}\}$, $H_\ell=P_\ell$. Let $H_0=H_{\mathcal{S}+1}=\emptyset$.

Let P be the multiset given by the disjoint union of the multisets $P_1,\dots,P_{\mathcal{S}}$ and similarly, let H be the disjoint union of the multisets $H_1,\dots,H_{\mathcal{S}}$. Now, for all $\ell\in[S]$, $\alpha\in P_\ell$, and $\beta\in H_\ell$ we want to define $\bigwedge$-formulas $x_{\alpha,\gamma}$ and $x_{\gamma',\beta}$ such that we can easily derive from $C_1,\dots,C_m$ the cedents

$$\{x_{\alpha\gamma}:\gamma\in H\} \tag{38}$$

$$\{\neg x_{\alpha\beta}\} \quad \text{for all } \beta\notin H_{\ell-1}\cup H_\ell\cup H_{\ell+1} \tag{39}$$

$$\{\neg x_{\alpha\gamma}, x_{\alpha\gamma'}:\gamma'\in H_{\ell-1}\setminus\{\gamma\}\} \quad \text{for all } \; \gamma\in H_{\ell-1} \tag{40}$$

$$\{\neg x_{\alpha\gamma_1},\; \neg x_{\alpha\gamma_2},\; \neg x_{\alpha\gamma_3}\} \quad \text{for all distinct } \gamma_1,\gamma_2,\gamma_3\in H_{\ell-1}\,, \tag{41}$$

$$\{\neg x_{\gamma\beta},\; \neg x_{\gamma'\beta}\} \quad \text{for all distinct } \gamma\in P_\ell\cup P_{\ell+1}\,,\, \gamma'\in P \tag{42}$$

$$\{\neg x_{\gamma_1\beta},\; \neg x_{\gamma_2\beta}, \neg x_{\gamma_3\beta}\} \quad \text{for all distinct } \gamma_1,\gamma_2,\gamma_3\in P_{\ell-1}\,. \tag{43}$$

Informally, the idea is very similar to Theorem 4.2. We want the $\bigwedge$-formulas $x_{\alpha,\beta}$ to express that if the clause C_α is true then α flies to itself (as a hole), and if it is false and its weight is positive, it flies to *all* the false premises used to derive it (i.e. two in the case of the FOLD and one in all remaining cases) or to its appearing sibling. If C_α is a weakening of an initial clause, it flies to itself. If the weight of C_α is negative, then α flies to its copy in the direction of the proof, or to its disappearing sibling. If we define a mapping from pigeons to holes in this way, there might be collisions due to the UNFOLD rules. Those types of collisions are exactly the ones allowed to have in the $\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ principle, since they correspond to mapping two pigeons with mass 2^j to one hole with capacity 2^{j+1}.

Given $\alpha \in \pi \cup \mathcal{L}_{s+1}$, let i_α be the unique index of the level where α belongs, i.e. $\alpha \in \mathcal{L}_{i_\alpha}$, and let w_α be the weight of α. Recall that given α, β in π we say that β is a *premise* of α if $i_\alpha = i_\beta + 1$, and between the layers $\mathcal{L}_{i_\beta}$ and $\mathcal{L}_{i_\alpha}$ we apply one of the inference rules of Fig. 5, with β one of the premises and α one of the conclusions. β is an *UNFOLD-premise* of α if β is a premise of α and the rule applied is the UNFOLD rule. The rest of the terminology is the same as in the proof of Theorem 4.2.

Using the terminology from Theorem 4.2, the definition of $x_{\alpha,\beta}$ is the same as the definition of $p_{\alpha,\beta}$, with just two more cases. For $\alpha \in P$ and $\beta \in H$, the formula $x_{\alpha,\beta}$ is

$$x \vee \neg x \quad \text{if } \alpha = \beta \text{ and } \alpha \in \mathcal{L}_1\,,$$

$$\bigvee C_\alpha \quad \text{if } \alpha = \beta \text{ and } \alpha \notin \mathcal{L}_1\,,$$

$$\neg \bigvee C_\beta \quad \text{if} \begin{cases} \beta \text{ is a SYMM.CUT-premise of } \alpha \\ \beta \text{ is a CONTRACTION-premise of } \alpha \\ \beta \text{ is a FOLD/UNFOLD-premise of } \alpha \\ \alpha, \beta \text{ are appearing siblings and } w_\alpha > 0 \\ \alpha \text{ is a positive-copy of } \beta \end{cases}$$

$$\neg \bigvee C_\beta \quad \text{if} \begin{cases} \alpha, \beta \text{ are disappearing siblings and } w_\alpha < 0 \\ \beta \text{ is a negative-copy of } \alpha \end{cases}$$

$$\neg \bigvee C_\alpha \quad \text{if } \beta \text{ is a SPLIT-premise of } \alpha\,,$$

$$\bot \quad \text{otherwise}\,.$$

Notice that if $\alpha \in P_\ell$, and $x_{\alpha,\beta} \neq \bot$, then $\beta \in H_\ell$ in all cases except for the FOLD/UNFOLD where $\beta \in H_{\ell-1}/H_{\ell+1}$.

The axioms that require a slightly different argument from the proof of Theorem 4.2 are (40)–(43). The axiom (40) is a weakening of $\top$ in all cases, except when α is the conclusion of a FOLD rule and γ is one of its premises. Let 2^ℓ be the weight of α, i.e. both its FOLD premises β, γ have weights $2^{\ell-1}$ and

$$\{\neg x_{\alpha\gamma}, x_{\alpha\gamma'} : \gamma' \in H_{\ell-1} \setminus \{\gamma\}\} = \{\bigvee C_\alpha, \neg \bigvee C_\alpha\}\,.$$

The axiom (41) is always a weakening of $\top$, since all inference rules have at most 2 premises and none of the $\gamma_1, \gamma_2, \gamma_3$ can be α, since $\alpha \in P_\ell$ and the γ_is are in $H_{\ell-1}$. Hence, at least one among the variables $x_{\alpha\gamma_1}, x_{\alpha\gamma_2}, x_{\alpha\gamma_3}$ is $\bot$ and its negation is true, i.e. $\top$. Similarly, the axiom (43) is always a weakening of $\top$, since all the rules have at most two conclusions and the γ_is cannot be β, for the same reason as before. Hence, one among the variables $x_{\gamma_1\beta}, x_{\gamma_2\beta}, x_{\gamma_3\beta}$ is always $\bot$.

To check the axioms in (42) we proceed exactly as in the cases of the injectivity in Theorem 4.2. Notice that, for β an UNFOLD-premise of γ and γ', the cedents $\{\neg x_{\gamma\beta},\ \neg x_{\gamma'\beta}\}$ are <u>not</u> part of the cedents in eq. (42).

We showed that from the clauses $C_1, \ldots, C_m$ in tree-like Res(d) it is possible to derive all the clauses of the formula $\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ in the formulas $x_{\alpha,\beta}$, which is a $\mathtt{LPHP}^{n^2+1}_{n^2}(G)$ for some graph G of degree at most 3. □

The construction of the formulas $x_{\alpha,\beta}$ in the previous proof does not satisfy the *onto/functional* axioms of ofLPHP. The reason is the same we had for PHP and unary SA: the last layer $\mathcal{L}_s$ might contain arbitrary weighted clauses $[C_\beta; w_\beta]$. If they are true, they are mapped to themselves. If they are false, they are mapped to some hole in $\mathcal{L}_{s-1}$. We have no guarantees that the holes in $\mathcal{L}_s$ receive some pigeon. If $\mathcal{L}_s$ satisfies the condition (3′) of the definition of restricted w-Resolution (Definition 3.2) we can adapt the definition of $x_{\alpha,\beta}$ in the proof of Theorem 5.11 to satisfy the *onto/functional* axioms of the leveled pigeonhole principle.

Theorem 5.12. *For every d, tree-like* Res(d)$+\mathtt{ofLPHP}^{n^2+m}_{n^2}(G)$ *p-simulates degree-d* NS*, where G is restricted to bipartite graphs of degree at most* 3 *and the height of the tree-like* Res(d) $+ \mathtt{ofLPHP}^{n^2+m}_{n^2}(G)$ *derivations is* 5*.*

Proof. (sketch) We use the characterization of NS given by Theorem 3.5 and we reason basically as in Theorem 4.3. Let S be the size of the restricted weighted resolution refutation we want to p-simulate.

We know that the problematic clauses in $\mathcal{L}_s$ are weakening of initial axioms or a single instance of $[\emptyset; z]$. Let $z_1, \ldots, z_{\log S}$ be the binary expansion of z.

For each $z_j \neq 0$ we copy $[\emptyset; 2^{z_j}]$ to $\mathcal{L}_{s+1}$, and we define the formula $x_{\alpha,\beta}$ as in Theorem 5.11. Now the onto axioms for the holes in $\mathcal{L}_s$ become weakening of initial clauses except for the holes $[\emptyset; 2^{z_j}]$, which receive pigeons from the layer $\mathcal{L}_{s+1}$. We construct the sets P_j and H_j as in Theorem 5.11. The semantic meaning of the variables $x_{\alpha\beta}$ is the same as in Theorem 5.11 with the adaptation already seen in Theorem 4.3, i.e. that if the clause in α is a weakening of an initial axiom we define $x_{\alpha\alpha} = x \vee \neg x$ and $x_{\alpha\beta} = \bot$ for all $\beta \neq \alpha$. Compared to Theorem 5.11 there are additional substitution instances of the axioms of $\mathtt{ofLPHP}^{n^2+m}_{n^2}(G)$ that we need to show how to derive from the initial axioms. As in Theorem 4.3 this is a simple cases analysis. □

It is immediate to generalize Theorem 5.11 from clauses to Θ_d-cedents. The argument for this generalization is the same as in Theorem 4.4.

Theorem 5.13. *For all $d \in \mathbb{N}$, tree-like* $\mathsf{Frege}_{d+1} + \mathtt{LPHP}^{n^2+1}_{n^2}(G)$ *p-simulates* w-Frege_d*, where G is restricted to bipartite graphs of degree at most* 3*.*

6. Open questions

In addition to the open questions left in Fig. 1, 2 and 3, we conclude this article with a list of open problems.

1. Prove that depth-d Frege + LPHP is strictly weaker than Frege, say for at least d constant.
2. Refining on the problem above, prove that the formula $\mathtt{MOD}_2$ (see Definition 4.6) does not have polynomial size refutations in Frege_d + LPHP, say for at least d constant.
3. Does Frege_d + PHP, say for constant d, have polynomial size refutations of LPHP? A negative answer, together with Theorem 5.13, would imply super-polynomial size lower bounds for w-Frege_d.

7. Proof of Theorem 3.5

Theorem 7.1 *(Normal form for* NS/SA *proofs). Given a (unary)* NS *derivation π of p_0 as in eq. (1), there is a (unary)* NS *derivation of p_0 of the form*

$$p_0 = \sum_{i=1}^{\ell} cp_i + \sum_{j=1}^{n} r_j(x_j^2 - x_j) + \sum_{j=1}^{n} r''_j(x_j + \bar{x}_j - 1) - \sum_{i=1}^{\ell} q'_i p_i \tag{44}$$

with size only polynomially larger than π, a constant $c > 0$ and all polynomials q'_i with positive coefficients. Similarly, given a (unary) SA *derivation π of p_0 as in eq. (2), if all the p_is have negative coefficients, there is a (unary)* SA *derivation of p_0 of the form*

$$p_0 = \sum_{i=1}^{\ell} cp_i + \sum_{j=1}^{n} r_j(x_j^2 - x_j) + \sum_{j=1}^{n} r''_j(x_j + \bar{x}_j - 1) + q_0 - \sum_{i=1}^{\ell} q'_i p_i \tag{45}$$

with size only polynomially larger than π, a constant $c > 0$ and all polynomials q'_i with positive coefficients.

An analogous result appeared independently in [17, Theorem 1.5].

Proof. Let ax_jm be a monomial in q_i. If $a < 0$ consider this monomial to be part of q'_i (this case can only happen in NS). If $a > 0$ then we can rewrite amx_jp_i as

$$amx_jp_i = amp_i(x_j + \bar{x}_j - 1) - am\bar{x}_jp_i + amp_i \,,$$

where the polynomial amp_i is going to be part of r''_j and the polynomial $am\bar{x}_j$ is going to be part of q'_i. We then rewrite amp_i in an analogous way, variable by variable. We repeat this for all the monomials in all the q_is. This way the sum $\sum_{i\in[\ell]} q_ip_i$ is rewritten as $\sum_{i\in[m]} c_ip_i$ for some constants $c_i > 0$ at the cost of adding monomials to the r''_js and q'_is. Let $c = \max_{i\in[\ell]} c_i$. We can then further rewrite $\sum_{i\in[\ell]} c_ip_i$ as

$$\sum_{i\in[\ell]} c_ip_i = \sum_{i\in[\ell]} cp_i - \sum_{i\in[\ell]} (c - c_i)p_i \,.$$

To conclude, we just consider all monomials in $(c - c_i)p_i$ as part of q'_i. □

Notice that, if all the coefficients in $p_1, \ldots, p_\ell$ are negative, then the Normal Form for SA in the theorem above (i.e. eq. (45)) gets further simplified to

$$p_0 = \sum_{i=1}^{m} cp_i + \sum_{j=1}^{n} r_j(x_j^2 - x_j) + \sum_{j=1}^{n} r''_j(x_j + \bar{x}_j - 1) + q'_0 \,,$$

for some polynomial q'_0 with positive coefficients, since all monomials in $-\sum_{i=1}^{\ell} q'_ip_i$ have positive coefficients.

Proof of Theorem 3.5. Given a clause $C = \{x_i : i \in I\} \cup \{\neg x_j : j \in J\}$ let $M(C)$ be the monomial $\prod_{i\in I} \bar{x}_i \prod_{j\in J} x_j$ and vice versa, given a monomial $m = \prod_{i\in I} \bar{x}_i \prod_{j\in J} x_j$ let $C(m)$ be the clause $\{x_i : i \in I\} \cup \{\neg x_j : j \in J\}$.

The argument is essentially the same for all the cases. Let's see it first for NS. Suppose we have some set of clauses $F = \{C_1, \ldots, C_\ell\}$. By Theorem 7.1 a NS refutation of F can be supposed to have the form

$$-z - \sum_{i=1}^{\ell'} w_im_iM(C_i) = -\sum_{i=1}^{\ell} wM(C_i) + \sum_{j=1}^{n} r_j(x_j^2 - x_j) + \sum_{j=1}^{n} r'_j(x_j + \bar{x}_j - 1) \,, \tag{46}$$

for some $z > 0$, $w, w_i \geq 0$, polynomials r_j, r'_j, monomials m_i and $\ell' \geq \ell$. If $j > \ell$, C_j is one among the clauses $C_1, \dots, C_\ell$. Recall that $-M(C_i)$ is how the clause C_i is encoded in (semi)-algebraic proof systems.

The idea now is to interpret each monomial in eq. (46) as a weighted clause: a monomial $-wm$ is interpreted as the weighted clause $[C(m); w]$.

Given two clauses C, D, we slightly abuse notation and denote $C \cup D$ simply as C, D. For instance, $-z$ is the weighted clause $[\emptyset; z]$ and $-w_i m_i M(C_i)$ is $[C_i, C(m_i); w_i]$.

We can then start constructing a weighted Resolution refutation $(\mathcal{L}_1, \dots, \mathcal{L}_s)$ of F.

The multiset $\mathcal{L}_1$ is $\{[C_i; w] : i = 1, \dots, \ell\}$ and corresponds to $-\sum_{i=1}^{\ell} wM(C_i)$. Suppose we already constructed $\mathcal{L}_j$, then pick any binomial of the form $wm(x_j^2 - x_j)$, not already picked from the sum $\sum_{j=1}^{n} r_j(x_j^2 - x_j)$, and let

$$\mathcal{L}_{j+1} = \mathcal{L}_j \cup \{[C(m), \neg x_j, \neg x_j; -w], [C(m), \neg x_j; w]\} .$$

We need to justify how to obtain $\mathcal{L}_{i+1}$ from $\mathcal{L}_i$ applying the rules of Fig. 5. This is immediate. We interleave intermediate multisets between $\mathcal{L}_i$ and $\mathcal{L}_{i+1}$

$$\frac{\dfrac{\mathcal{L}_j}{\mathcal{L}_j, [C(m), \neg x_j, \neg x_j; w],\ [C(m), \neg x_j, \neg x_j; -w]}}{\mathcal{L}_j, [C(m), \neg x_j; w],\ [C(m), \neg x_j, \neg x_j; -w] .}$$

Continue this way till all the binomials from $\sum_{j=1}^{n} r_j(x_j^2 - x_j)$ are picked. Then continue with the trinomials from $\sum_{j=1}^{n} r'_j(x_j + \bar{x}_j - 1)$. Suppose we constructed $\mathcal{L}_k$, then pick any trinomial of the form $wm(x_j + \bar{x}_j - 1)$, not already picked from the sum $\sum_{j=1}^{n} r'_j(x_j + \bar{x}_j - 1)$, and let

$$\begin{aligned} \mathcal{L}_{k+1} = \mathcal{L}_k & \\ & \cup \{[C(m), \neg x_j; -w],\ [C(m), x_j; -w],\ [C(m); w]\} . \end{aligned}$$

Again, we need to justify how to obtain $\mathcal{L}_{k+1}$ from $\mathcal{L}_k$ applying the rules of Fig. 5. Again, this again immediate. We interleave intermediate multisets between $\mathcal{L}_k$ and $\mathcal{L}_{k+1}$

$$\frac{\dfrac{\dfrac{\mathcal{L}_k}{\mathcal{L}_k,\ [C(m), x_j; w],\ [C(m), x_j; -w]}}{\mathcal{L}_k,\ [C(m), x_j; w],\ [C(m), x_j; -w],\ [C(m), \neg x_j; w],\ [C(m), \neg x_j; -w]}}{\mathcal{L}_k,\ [C(m); w],\ [C(m), x_j; -w],\ [C(m), \neg x_j; -w] .}$$

After we finish this process, let $\mathcal{L}_{s'}$ the multiset we got. We exhausted all the terms from the RHS of eq. (46) and all the monomials, except the ones in the LHS of eq. (46), must cancel. This means that from $\mathcal{L}_{s'}$ with some applications of the FOLD/UNFOLD/REMOVAL rules we eventually get to

$$\mathcal{L}_s = \{[\emptyset; z],\ [C_i, C(m'_i); w'_i] : i = 1, \dots, \ell\} .$$

This multiset satisfies the SOUNDNESS-NS condition. This concludes the proof that NS-SOUND weighted Resolution p-simulates NS.

For the case of SA the argument is completely analogous. A SA refutation of F has the form

$$-z - \sum_{i \in J} w'_i m'_i = -\sum_{i=1}^{\ell} wM(C_i) + \sum_{j=1}^{n} r_j(x_j^2 - x_j)$$

$$+\sum_{j=1}^{n} r'_j(x_j + \bar{x}_j - 1)\,, \tag{47}$$

for some $z > 0$, $w_i, w'_i \geq 0$, polynomials r_j, r'_j and monomials m_i. With the same construction as above we arrive to a

$$\mathcal{L}_s = \{[\emptyset; z],\ [C(m'_i); w'_i] : i \in I\}\,,$$

and this multiset clearly satisfies the condition that all the weights are non-negative.

The other direction of the p-simulations is easier. Given a w-Resolution refutation $(\mathcal{L}_1, \ldots,\ \mathcal{L}_s)$ we want to construct an algebraic expression having the form of a NS/SA refutation. Let $S_1 = \sum_{[C;w]\in\mathcal{L}_1} -wM(C)$. By assumption, all the clauses C in S_1 are clauses from F. Then suppose we constructed an algebraic expression $S_i = -\sum_{[C;w]\in\mathcal{L}_i} wM(C)$ having the form of a NS/SA derivation.

We want then to construct S_{i+1}. If from $\mathcal{L}_i$ to $\mathcal{L}_{i+1}$ is applied a SYMMETRIC CUT rule $\frac{[C,x;w] \quad [C,\neg x;w]}{[C;w]}$, then add to the sum the terms

$$-wM(C) + wM(C,x) + wM(C,\neg x) = wM(C)(x + \bar{x} - 1)\,.$$

If from $\mathcal{L}_i$ to $\mathcal{L}_{i+1}$ is applied a SPLIT rule $\frac{[C;w]}{[C,x;w] \quad [C,\neg x;w]}$, then add to S_i the terms

$$wM(C) - wM(C,x) - wM(C,\neg x) = -wM(C)(x + \bar{x} - 1)\,.$$

If from $\mathcal{L}_i$ to $\mathcal{L}_{i+1}$ is applied a CONTRACTION rule $\frac{[C,\neg x,\neg x;w]}{[C,\neg x;w]}$ then add to S_i the terms

$$-wM(C,\neg x) + wM(C,\neg x,\neg x) = wM(C)(x^2 - x)\,.$$

If from $\mathcal{L}_i$ to $\mathcal{L}_{i+1}$ is applied a CONTRACTION rule $\frac{[C,x,x;w]}{[C,x;w]}$ then add to S_i the terms

$$\begin{aligned} &- wM(C,x) + wM(C,x,x) = wM(C)(\bar{x}^2 - \bar{x}) \\ &= wM(C)(x^2 - x) + (wM(C)\bar{x} - wM(C)x)(\bar{x} + x - 1)\,. \end{aligned}$$

If from $\mathcal{L}_i$ to $\mathcal{L}_{i+1}$ is applied an EXCL. MIDDLE rule $\frac{}{[x,\neg x;w]}$ then add to S_i the terms

$$\begin{aligned} -wM(x,\neg x) &= -wx\bar{x} \\ &= -x(x + \bar{x} - 1) - x(x^2 - x)\,. \end{aligned}$$

If $\mathcal{L}_i$ to $\mathcal{L}_{i+1}$ is applied some other rules let $S_{i+1} = S_i$.[4]

It is immediate to see that S_{i+1} constructed following the procedures above is such that $S_{i+1} = -\sum_{[C;w]\in\mathcal{L}_{i+1}} wM(C)$.

Then, the soundness conditions will guarantee that the final sum S_s has the form required to be a NS/SA refutation respectively. □

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.

[4] The other rules are not important since, in NS/SA, the cancellations between monomials are done implicitly by the underlying algebraic structure. That is, for instance, there is no need of a rule saying that $m + m - 2m = 0$. Instead, in w-Resolution, all the cancellations between weighted clauses are done explicitly by applications of some rules.

Acknowledgements

We would like to thank Albert Atserias, Massimo Lauria, Sam Buss, Neil Thapen and Moritz Müller for very fruitful conversations that simplified and improved this paper.

This work was supported by the Ministerio de Ciencia e Innovación/Agencia Estatal de Investigación MCIN/AEI/10.13039/501100011033, Spain [grant numbers PID2019-109137GB-C21, PID2019-109137GB-C22, IJC2018-035334-I].

Data availability

No data was used for the research described in the article.

References

[1] M. Ajtai, Parity and the Pigeonhole Principle, Birkhäuser Boston, Boston, MA, 1990, pp. 1–24.
[2] A. Atserias, M. Lauria, Circular (yet sound) proofs, in: M. Janota, I. Lynce (Eds.), Theory and Applications of Satisfiability Testing - SAT 2019 - 22nd International Conference, SAT 2019, Lisbon, Portugal, July 9-12, 2019, Proceedings, in: Lecture Notes in Computer Science, vol. 11628, Springer, 2019, pp. 1–18.
[3] A. Atserias, J. Ochremiak, Proof complexity meets algebra, ACM Trans. Comput. Log. 20 (1) (2018) 1–46.
[4] P. Beame, S. Cook, J. Edmonds, R. Impagliazzo, T. Pitassi, The relative complexity of NP search problems, J. Comput. Syst. Sci. 57 (1) (1998) 3–19.
[5] P. Beame, R. Impagliazzo, J. Krajicek, T. Pitassi, P. Pudlak, Lower bounds on Hilbert's Nullstellensatz and propositional proofs, in: Proceedings 35th Annual Symposium on Foundations of Computer Science, 1994, pp. 794–806.
[6] P. Beame, T. Pitassi, An exponential separation between the parity principle and the pigeonhole principle, Ann. Pure Appl. Log. 80 (1996) 195–228.
[7] A. Beckmann, S. Buss, On transformations of constant depth propositional proofs, Ann. Pure Appl. Log. 170 (10) (2019) 1176–1187.
[8] I. Bonacina, M.L. Bonet, On the strength of Sherali-Adams and nullstellensatz as propositional proof systems, Electron. Colloq. Comput. Complex. (2021) TR22–182.
[9] I. Bonacina, M.L. Bonet, On the strength of Sherali-Adams and nullstellensatz as propositional proof systems, in: C. Baier, D. Fisman (Eds.), LICS '22: 37th Annual ACM/IEEE Symposium on Logic in Computer Science, Haifa, Israel August 2-5, 2022, ACM, 2022, pp. 25:1–25:12.
[10] M.L. Bonet, S. Buss, A. Ignatiev, J. Marques-Silva, A. Morgado, Maxsat resolution with the dual rail encoding, in: AAAI, 2018, pp. 6565–6572.
[11] M.L. Bonet, S. Buss, A. Ignatiev, A. Morgado, J. Marques-Silva, Propositional proof systems based on maximum satisfiability, Artif. Intell. 300 (2021) 103552.
[12] M.L. Bonet, J. Levy, Equivalence between systems stronger than resolution, in: Theory and Applications of Satisfiability Testing – SAT 2020, Springer International Publishing, Cham, 2020, pp. 166–181.
[13] S.R. Buss, Lower bounds on Nullstellensatz proofs via designs, in: Proof Complexity and Feasible Arithmetics, Rutgers, NJ, 1996, in: DIMACS Ser. Discrete Math. Theoret. Comput. Sci., vol. 39, Amer. Math. Soc., Providence, RI, 1998, pp. 59–71.
[14] S.A. Cook, R.A. Reckhow, The relative efficiency of propositional proof systems, J. Symb. Log. 44 (1) (1979) 36–50.
[15] S.S. Dantchev, A. Ghani, B. Martin, Sherali-Adams and the binary encoding of combinatorial principles, in: Y. Kohayakawa, F.K. Miyazawa (Eds.), LATIN 2020: Theoretical Informatics - 14th Latin American Symposium, São Paulo, Brazil, January 5-8, 2021, Proceedings, in: Lecture Notes in Computer Science, vol. 12118, Springer, 2020, pp. 336–347.
[16] S.F. de Rezende, M. Lauria, J. Nordström, D. Sokolov, The power of negative reasoning, in: V. Kabanets (Ed.), 36th Computational Complexity Conference (CCC 2021), in: Leibniz International Proceedings in Informatics (LIPIcs), vol. 200, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2021, pp. 40:1–40:24.
[17] N. Fleming, S. Grosser, M. Göös, R. Robere, On semi-algebraic proofs and algorithms, Electron. Colloq. Comput. Complex. (2022) TR22–003.
[18] N. Fleming, P. Kothari, T. Pitassi, Semialgebraic proofs and efficient algorithm design, Found. Trends Theor. Comput. Sci. 14 (1–2) (2019) 1–221.
[19] M. Göös, A. Hollender, S. Jain, G. Maystre, W. Pires, R. Robere, R. Tao, Separations in proof complexity and TFNP, in: 63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS, 2022.
[20] M. Göös, A. Hollender, S. Jain, G. Maystre, W. Pires, R. Robere, R. Tao, Separations in proof complexity and TFNP, J. ACM 71 (4) (Aug. 2024), https://doi.org/10.1145/3663758.
[21] R. Impagliazzo, N. Segerlind, Constant-depth Frege systems with counting axioms polynomially simulate nullstellensatz refutations, ACM Trans. Comput. Log. 7 (2) (2006) 199–218.
[22] J. Krajiček, Lower bounds to the size of constant-depth propositional proofs, J. Symb. Log. 59 (1) (1994) 73–86.
[23] J. Krajíček, Proof Complexity, Cambridge University Press, 2019.
[24] J. Krajíček, P. Pudlák, A. Woods, An exponential lower bound to the size of bounded depth Frege proofs of the pigeonhole principle, Random Struct. Algorithms 7 (1) (1995) 15–39.

[25] J. Larrosa, E. Rollon, Augmenting the power of (partial) maxsat resolution with extension, Proc. AAAI Conf. Artif. Intell. 34 (02) (2020) 1561–1568.
[26] J. Larrosa, E. Rollon, Towards a better understanding of (partial weighted) maxsat proof systems, in: L. Pulina, M. Seidl (Eds.), Theory and Applications of Satisfiability Testing – SAT 2020, Springer International Publishing, Cham, 2020, pp. 218–232.
[27] T. Pitassi, R. Robere, Strongly exponential lower bounds for monotone computation, in: Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Association for Computing Machinery, New York, NY, USA, 2017, pp. 1246–1255.
[28] T. Pitassi, R. Robere, Lifting nullstellensatz to monotone span programs over any field, in: Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Association for Computing Machinery, New York, NY, USA, 2018, pp. 1207–1219.
[29] T. Pitassi, P. Beame, R. Impagliazzo, Exponential lower bounds for the pigeonhole principle, Comput. Complex. 3 (2) (1993) 97–140.
[30] R. Robere, T. Pitassi, B. Rossman, S.A. Cook, Exponential lower bounds for monotone span programs, in: 2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS), 2016, pp. 406–415.
[31] E. Rollon, J. Larrosa, Proof complexity for the maximum satisfiability problem and its use in SAT refutations, J. Log. Comput. 03 (2022).
[32] H.D. Sherali, W.P. Adams, A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems, SIAM J. Discrete Math. 3 (3) (1990) 411–430.
[33] A. Urquhart, X. Fu, Simplified lower bounds for propositional proofs, Notre Dame J. Form. Log. 37 (4) (1996) 523–544.