Electronic Colloquium on Computational Complexity, Report No. 182 (2021)

ECCC

On the strength of Sherali-Adams and Nullstellensatz as propositional proof systems

Ilario Bonacina[1] and Maria Luisa Bonet[2]

[1,2]Universitat Politècnica de Catalunya
{bonacina, bonet}@cs.upc.edu

2021-12-30

Abstract

The propositional proof system Sherali-Adams (SA) has polynomial-size proofs of the pigeonhole principle (PHP). Similarly, the Nullstellensatz (NS) proof system has polynomial size proofs of the bijective (i.e. both functional and onto) pigeonhole principle (ofPHP). We characterize the strength of these algebraic proof systems in terms of Boolean proof systems the following way. We show that SA (resp. NS over $\mathbb{Z}$) with unary coefficients lies strictly between tree-like resolution and tree-like depth-1 Frege + PHP (resp. ofPHP). We introduce weighted versions of PHP and ofPHP, resp. wtPHP and of-wtPHP and we show that SA (resp. NS over $\mathbb{Z}$) lies strictly between resolution and tree-like depth-1 Frege + wtPHP (resp. of-wtPHP). We also show analogue results for "depth-d" versions of SA and NS.

1 Introduction

The Nullstellensatz proof system (NS) was introduced in [BIKPP94] due to a connection with a big (and yet open) problem in proof complexity: the problem of proving super-polynomial size lower bunds for bounded-depth Frege systems with modular gates. NS received some attention during the late '90s [BCEIP98; BP96; Bus98] (among others), then, the interest gradually shifted towards stronger algebraic proof systems such as Polynomial Calculus. More recently, Nullstellensatz has gained some renewed interest mostly due to the fact that lower bounds in this system can be lifted to lower bounds for stronger proof systems via composition with gadgets [DMNR21; PR17; PR18; RLNS21; RPRC16].

Sherali-Adams (SA) was introduced in [SA90] as a method to give a hierarchy of linear programming relaxations of any 0-1 integer program. The interest in studying this system relies primarily in its connections to approximation algorithms for important NP-hard optimisation problems, see for instance the excellent survey [FKP19].

Sherali-Adams, as a propositional proof system, also showed-up naturally in the context of proof systems for MaxSAT extending MaxSAT resolution [BBIMM18; BL20; FMSV20; LR20a; LR20b], and in the context of enriching the DAG structure of resolution refutations, as in circular resolution [AL19].

In this paper we want to compare the strength of these semi-algebraic proof systems with the standard restrictions of Frege systems, like resolution and bounded-depth Frege. That is, we look at such propositional proof systems SA (NS) from the point of view of the questions:

"*What axioms do we need to add to constant-depth Frege to p-simulate* SA *(*NS*)?*"

ISSN 1433-8092

“*What is the minimal depth of constant-depth Frege (plus the extra axioms) needed to p-simulate* SA *(*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 (Definition 4.9) p-simulates NS with coefficients over $\mathbb{Z}_2$ [IS06]. For the formal definition of propositional proof system and p-simulation see the beginning of Section 2.

We use the pigeonhole principle (`PHP`, see Definition 4.1), which informally says that $n+1$ pigeons cannot all fly to n pigeonholes without any two of them sharing a pigeonhole. We use the *bijective* (i.e. *onto-functional*) pigeonhole principle (`ofPHP`, see Definition 4.1). We also use a generalization of `PHP`, the *weighted* pigeonhole principle (`wtPHP`, see Definition 5.1). The `wtPHP` informally captures a similar combinatorial principle, where the pigeons have some “mass” and the holes have some “capacity”. 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. Now, each pigeon can fly once with the whole mass or twice with half mass. Each hole can accept either 1 pigeon filling the full capacity or 2 pigeons filling half capacity each. This principle is provable in SA *but* it seems to require binary coefficients (Theorem 5.4).

In this paper we answer the questions above for NS and SA with coefficients over $\mathbb{Z}$ (Definition 2.7 and Definition 2.8). A bit unexpectedly, their strength seems to depend on whether the coefficients of the polynomials are encoded in unary or binary. We summarise visually our results although the formal statements of the cited theorems are slightly stronger than what is shown in the figures.

As you can see in Fig. 1.1, tree-like depth-1 Frege + `wtPHP` is strictly stronger than SA and SA is strictly stronger than resolution. On the other hand tree-like depth-1 Frege + `PHP` is strictly stronger than unary SA and unary SA is is strictly stronger than tree-like resolution. We also show that SA and unary SA are p-equivalent to other Boolean proof systems, in particular weighted Resolution (see Definition 3.1).

As you can see in Fig. 1.2, tree-like depth-1 Frege + `of-wtPHP` is stronger than NS. On the other hand tree-like depth-1 Frege + `ofPHP` is stronger than unary NS and unary NS is is strictly stronger than tree-like resolution. We also show that NS and unary NS are p-equivalent to other Boolean proof systems.

SA is p-equivalent to weighted Resolution and we can generalize weighted Resolution to weighted depth-d Frege (see Definition 3.1). Informally, one could think of weighted depth-d Frege as SA over depth-d formulas. Fig. 1.3 shows the generalisation of the results in Fig. 1.1 from “depth-0” to “depth-d”, i.e. the results for “SA over depth-d formulas” aka weighted depth-d Frege.

Another contribution is Theorem 3.9, a way of looking at semi-algebraic proof systems such as NS/SA as weighted resolution, i.e. a proof systems handling weighted clauses where we have two distinct soundness conditions, one characterizing NS and the other characterizing SA. The interest in this result is that it allows to think at derivations of semi-algebraic proof systems in a natural and visual way. It is a natural language in which to describe the p-simulations in Fig. 1.1 and 1.2. For instance, using this characterization of NS/SA, it is very clear why some p-simulations will only work for SA/NS with coefficients in unary. Moreover, for instance, with the language of weighted resolution, it is very easy to show that tree-like depth-1 Frege + $\mathtt{MOD}_2$ p-simulates $\mathsf{NS}_{\mathbb{Z}_2}$. Thus, partially, reproving [IS06, Theorem 4.4], but with a tighter bound on the depth of formulas used in the p-simulation.

From the p-simulations we prove (and known results from the literature) it is also immediate to infer that $\mathtt{MOD}_2$ does not have polyonmial-size refutations in unary SA over depth-d formulas, at least for $d = o(\frac{\log\log n}{\log\log\log n})$ (Corollary 4.10).

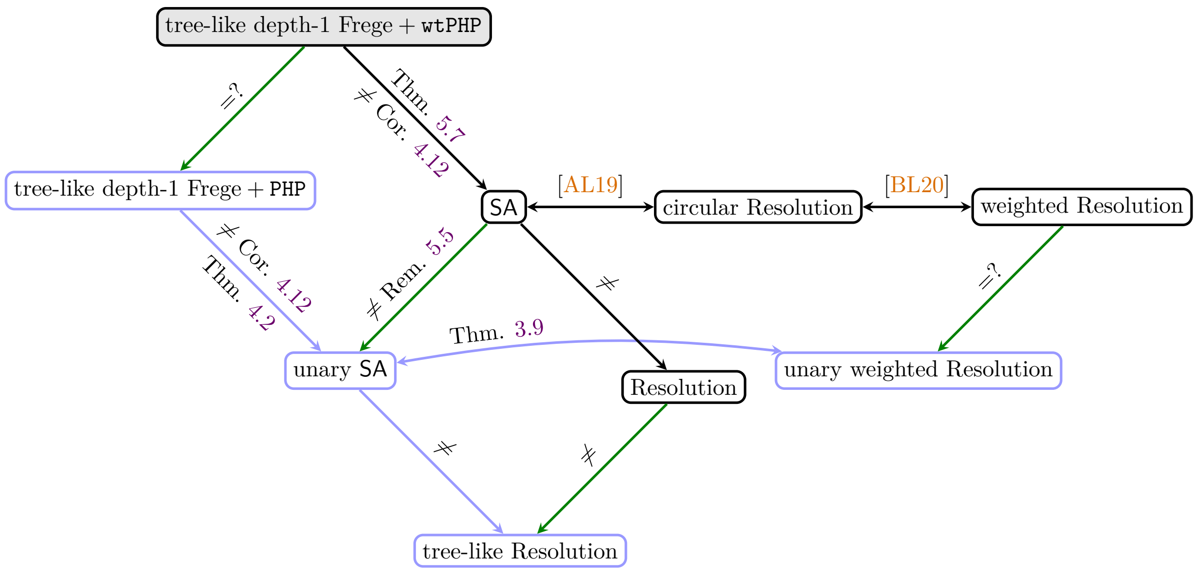

Figure 1.1: The p-simulations for SA. The notation $P \rightarrow Q$ means that the proof system P p-simulates the proof system Q. The p-simulations are annotated with "$\neq$" if the p-simulation is known to be strict, or with "=?" if it is an open question if the p-simulation is strict or not. An arrow $\rightarrow$ means the p-simulation is trivial. The color • is used to visually differentiate the results for the proof systems with unary weights/coefficients. The new proof systems introduced in this paper are highlighted .

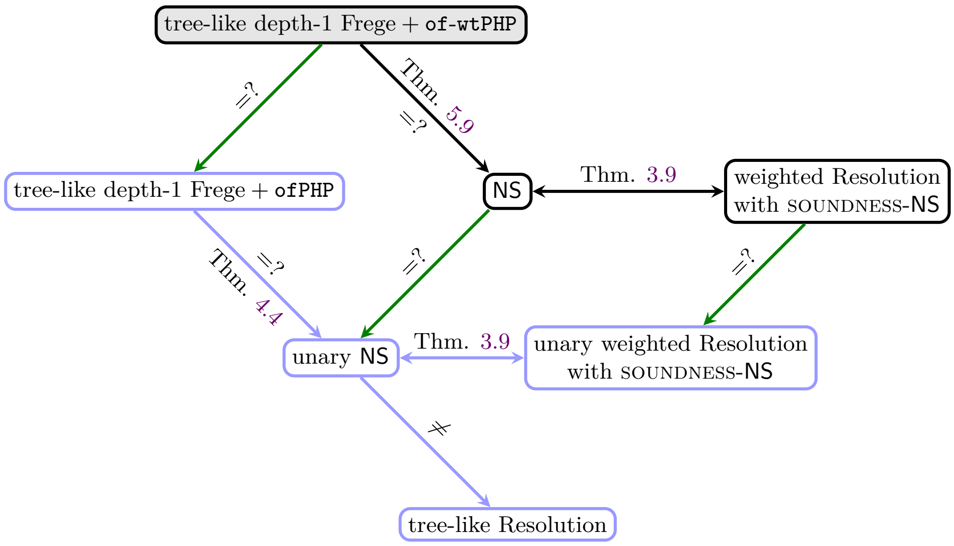

Figure 1.2: The p-simulations for NS.

1.1 Organisation of the paper

Section 2 contains all the basic definitions we need: the notion of depth-d Frege, depth-d Frege + φ, circular depth-d Frege and the semi-algebraic proof systems NS and SA.

Section 3 introduces the proof system weighted depth-d Frege with two soundness conditions, proves some basic facts about them, and the connection to semi-algebraic proof systems.

Section 4 contains the definition of the pigeonhole principle PHP and the simulation of unary SA (resp. unary NS) by depth-1 Frege + PHP (resp. depth-1 Frege + ofPHP).

Section 5, builds on the previous section and introduces a *weighted* version of the pigeonhole

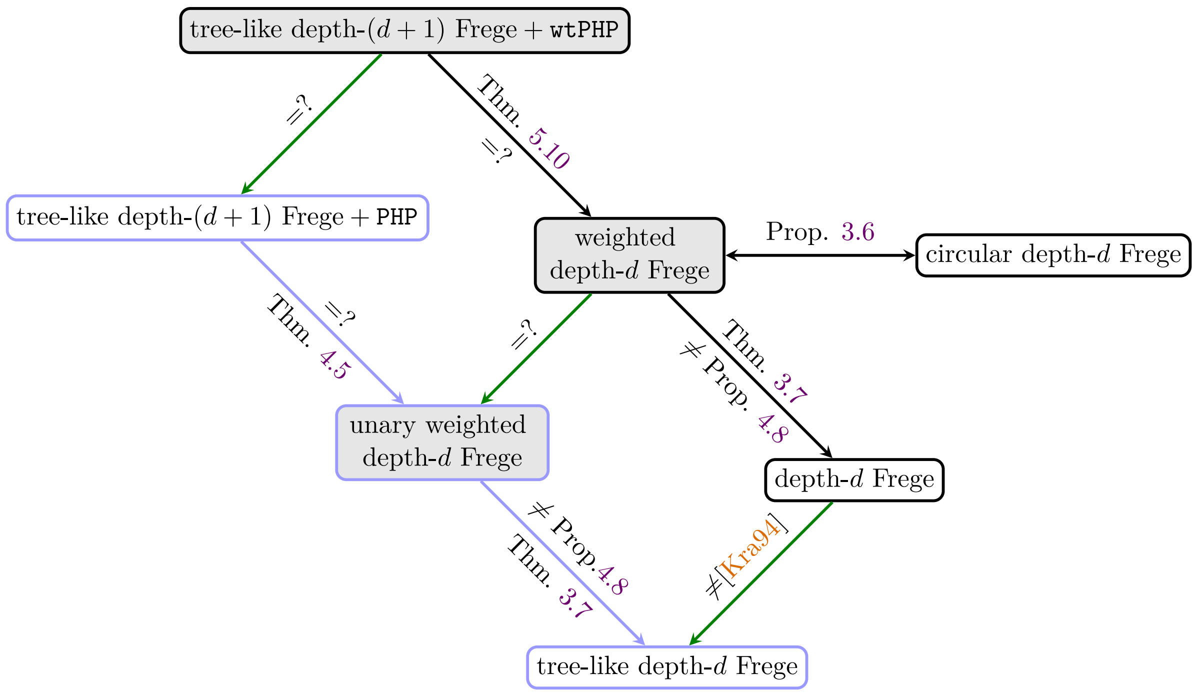

Figure 1.3: The p-simulations for weighted depth-d Frege.

principle wtPHP. we show how to refute it in SA and we show how to simulate SA by depth-1 Frege + wtPHP.

Appendix A recalls a way to refute PHP in SA.

Appendix B contains the proof of a technical lemma needed to show that the weighted pigeonhole principle is provable in SA.

2 Preliminaries

For $n \in \mathbb{N}$ let $[n] = \{1, \ldots, n\}$.

A propositional proof system is as polynomial time function $P\colon \{0,1\}^* \to \{0,1\}^*$ whose range is exactly the set TAUT of propositional tautologies in the DeMorgan language [CR79].

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 exist 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 [BB19] 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.

$$\frac{\Gamma, \varphi, \varphi}{\Gamma, \varphi} \quad (\textsc{contraction}),$$

$$\frac{}{\varphi, \neg\varphi} \quad (\textsc{excluded middle}),$$

$$\frac{\Gamma, \varphi \qquad \text{for } \varphi \in \Phi}{\Gamma, \bigwedge \Phi} \quad (\textstyle\bigwedge\text{-}\textsc{introduction}),$$

$$\frac{\Gamma, \neg\varphi \qquad \Gamma, \varphi}{\Gamma} \quad (\textsc{symmetric cut}),$$

$$\frac{\Gamma}{\Gamma,\ \Gamma'} \quad (\textsc{weakening}),$$

$$\frac{\Gamma, \Phi}{\Gamma, \bigvee \Phi} \quad (\textstyle\bigvee\text{-}\textsc{introduction}),$$

$$\frac{\Gamma, \bigvee \Phi}{\Gamma, \Phi} \quad (\textstyle\bigvee\text{-}\textsc{elimination}),$$

Figure 2.1: Inference rules of depth-d Frege. The cedents $\Gamma, \Gamma', \Phi, \bigvee \Phi, \bigwedge \Phi, \varphi, \neg\varphi$ all are Θ_d-cedents.

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

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

For $\varphi \in \Theta_d$ we denote by $\neg\varphi$ the formula in Θ_d obtained from φ 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 (depth-d Frege)**.** Let $\mathcal{F}$ be a set Θ_d-cedents. A depth-d Frege *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 inference rules in Fig. 2.1. The *size* of T is the number of symbols of distinct cedents in the derivation. If we count the number of symbols in *all* occurrences of cedents we use the adjective *tree-like*. A depth-d Frege *refutation* of $\mathcal{F}$ is a derivation of the empty cedent.

The definition of depth-d Frege in [BB19] is essentially the one given above with the CONTRACTION rule given implicitly, since cedents are sets. For us it is more convenient to have it given explicitly. The propositional proof system *resolution* is depth-0 Frege. In this system the $\bigwedge$ and $\bigvee$ rules of Fig. 2.1 cannot be applied.

Theorem 2.3 ([BB19, Theorem 4][1])**.** *Let $d \in \mathbb{N}$, $n \in \mathbb{N}$ and φ_n a collection of Θ_d-cedents. Then φ_n has a depth-d Frege refutation of size polynomial in n if and only if φ_n has a tree-like depth-$(d+1)$ Frege refutation of size polynomial in n.*

[1] The same result was proved in [BB05, Theorem 10] but one of the constructions there was incorrect and [BB19] corrects them.

Definition 2.4 (depth-(d,k) Frege and Res(k))**.** Let $d,k \in \mathbb{N}$. The system depth-(d,k) Frege is the restriction of depth-d Frege where the $\bigwedge$-INTRODUCTION rule from Fig. 2.1 is limited to sets of Θ_{d-1} formulas Φ of size at most k. The system depth-$(1,k)$ Frege is also called Res(k).

Given $\boldsymbol{\varphi} = (\varphi_n)_{n\in\mathbb{N}}$ a family of unsatisfiable cedents, for instance when φ_n is the pigeonhole principle $\mathtt{PHP}_n^{n+1}$ (see Section 4 for the definition of $\mathtt{PHP}_n^{n+1}$), the notion of depth-d Frege + $\boldsymbol{\varphi}$ has been considered for instance in [Ajt90; BP96] and in the context of bounded arithmetic it is also very common (see for instance [Kra19]). Informally, this is the system depth-d Frege where we would be allowed to infer substitution instances of the tautology $\neg\varphi_n$ for free.

Here we want to limit the number of times we are allowed to use the tautology $\neg\varphi_n$ and we want to limit a bit depth-d Frege as-well. That is, we are interested in defining depth-(d,k) Frege + $\boldsymbol{\varphi}$. Informally, depth-(d,k) Frege + $\boldsymbol{\varphi}$ is depth-(d,k) Frege where we have the extra power to reduce the formula we want to refute to a substitution instance of some φ_n, and φ_n is given for free in the sense that we already know it is unsatisfiable. In some sense, in the system depth-(d,k) Frege + $\boldsymbol{\varphi}$ we only allow the formulas φ_n to be used only once. Formally, the definition is the following.

Definition 2.5 (depth-(d,k) Frege + $\boldsymbol{\varphi}$)**.** Let $\boldsymbol{\varphi} = (\varphi_n)_{n\in\mathbb{N}}$, where φ_n is a set of s many Θ_d-cedents in n variables $x_1,\dots,x_n$. A refutation of a set of Θ_d-cedents F in depth-(d,k) Frege + $\boldsymbol{\varphi}$ is a set of depth-(d,k) Frege derivations $\Gamma_1,\dots,\Gamma_s$ of $G_1,\dots,G_s$ such that: either (1) $G_1=\emptyset$, i.e. Γ_1 is refutation of F and $s=1$, or (2) there is an $n\in\mathbb{N}$ such that the set of cedents $\{G_1,\dots,G_s\}$ is a substitution instance of φ_n.[2] The *height* of the refutation is the maximum height of $\Gamma_1,\dots,\Gamma_s$. The *size* of the refutation is the sum of the sizes of $\Gamma_1,\dots,\Gamma_s$.

2.2 Circular depth-d Frege

Atserias and Lauria [AL19] introduced the notion of *circular proofs* as a way to enrich the DAG structure of Frege derivations while preserving the soundness of the proof system. This notion is not strictly needed to understand the main results in this paper. The reading of this section can be deferred until the theorems/corollaries about circular Resolution and circular depth-d Frege. The formal definition of circular proofs in the context of depth-d Frege is the following.

Definition 2.6 (circular depth-d Frege)**.** Given the inference rules of depth-d Frege (that is the ones in Fig. 2.1), a set of depth-d formulas $\mathcal{H}$ and a depth-d formula C, a circular depth-d Frege proof of F from $\mathcal{H}$ is a bipartite direct graph $\mathcal{G}=(V,E)$ with V a multiset with bipartition $(\mathcal{I},\mathcal{F})$, where $\mathcal{I}$ is a multiset of inference rules of depth-d Frege and $\mathcal{F}$ is a multiset of formulas. We have edges $(I,F)\in E$ if F is a conclusion of the rule I and $(F,I)\in E$ if F is a premise of the rule I. Let $N^{\mathsf{in}}(F) = \{\, I\in\mathcal{I} : (I,F)\in E \,\}$ and $N^{\mathsf{out}}(F) = \{\, I\in\mathcal{I} : (F,I)\in E \,\}$. We require that there exists a function $f\colon \mathcal{I}\to\mathbb{N}^+$ such that

$$\sum_{I\in N^{\mathsf{in}}(C)} f(I) - \sum_{I\in N^{\mathsf{out}}(C)} f(I) > 0$$

and if for some formula $F\in\mathcal{F}$

$$\sum_{I\in N^{\mathsf{in}}(F)} f(I) - \sum_{I\in N^{\mathsf{out}}(F)} f(I) < 0$$

then $F\in\mathcal{H}$. If all the formulas in $\mathcal{F}$ have depth at most d then we say the derivation is in circular depth-d Frege. The *size* of a circular depth-d Frege derivation is the sum of all the length of all the formulas in $\mathcal{F}$.

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

Notice that the function f is not part of the circular proof, instead, given a circular proof to check it's correctness we need to calculate f and we can do it in polynomial time via linear programming (see [AL19] for more details).

In [AL19] the authors proved that circular depth-0 Frege, i.e. circular Resolution, is polynomially equivalent to Sherali-Adams (see Definition 2.8) while adding the power of "circularity" to Frege or TC^0-Frege does not make the systems stronger. Bonet and Levy [BL20] proved that circular resolution is also polynomially equivalent to *MaxSAT resolution with Extension* (see Definition 3.1).

A natural question is then to ask what is the strength of circular depth-d Frege. It is clearly stronger than depth-d Frege, but how much stronger? Theorem 5.11 is an answer to this question.

2.3 Algebraic and semi-algebraic proof systems

In this section we define formally the proof systems Nullstellensatz [BIKPP94] and Sherali-Adams [SA90].

Definition 2.7 (Nullstellensatz)**.** Given a ring R, variables $x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n$ and polynomials $p_0, \ldots, p_\ell \in R[x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n]$, a *Nullstellensatz over* R (NS_R) *proof of the equality* $p_0 = 0$ *from the equalities* $p_1 = 0, \ldots, p_\ell = 0$ 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 $R[x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n]$. A *refutation* of $p_1 = 0, \ldots, p_\ell = 0$ is a derivation of the equality $c = 0$ where $c \in R \setminus \{0\}$. 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 2.8 (Sherali-Adams)**.** Given a ordered ring $(R, <)$, variables $x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n$ and polynomials $p_0, \ldots, p_\ell \in R[x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n]$, a *Sherali-Adams over* R (SA_R) *proof of* $p_0 \geqslant 0$ *from* $p_1 \geqslant 0, \ldots, p_\ell \geqslant 0$ 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 $R[x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n]$ and the q_is are polynomials with positive coefficients. A *refutation* of $p_1 \geqslant 0, \ldots, p_\ell \geqslant 0$ is a derivation of $c \geqslant 0$ where $c \in R$ 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.

If in the definitions above we restrict the polynomials r'_j to be identically 0, the resulting systems are known to be exponentially weaker [RLNS21], with respect to size. The degree of the two versions of the systems is obviously the same.

In this paper we consider only Nullstellenstatz and Sherali-Adams over the ring $\mathbb{Z}$, resp. $\mathsf{NS}_{\mathbb{Z}}$ and $\mathsf{SA}_{\mathbb{Z}}$, hence from now we refer to them simply 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 system as unary NS and unary SA.

Theorem 2.9 (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*

$$p_0 = \sum_{i=1}^{\ell} c p_i + \sum_{j=1}^{n} r_j (x_j^2 - x_j) + \sum_{j=1}^{n} r''_j (x_j + \bar{x}_j - 1) - \sum_{i=1}^{\ell} q'_i p_i \tag{3}$$

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)*, there is a (unary)* SA *derivation of p_0 of the form*

$$p_0 = \sum_{i=1}^{\ell} c p_i + \sum_{j=1}^{n} r_j (x_j^2 - x_j) + \sum_{j=1}^{n} r''_j (x_j + \bar{x}_j - 1) + q_0 - \sum_{i=1}^{\ell} q'_i p_i \tag{4}$$

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

Proof. Let $a x_j m$ be a monomial in q_i. If $a < 0$ consider this monomial to be part of q'_i. If $a > 0$ then we can rewrite $a m x_j p_i$ as

$$a m x_j p_i = a m p_i (x_j + \bar{x}_j - 1) - a m \bar{x}_j p_i + a m p_i \,,$$

where the polynomial $a m p_i$ is going to be part of r''_j and the polynomial $a m \bar{x}_j$ is going to be part of q'_i. We then rewrite $a m p_i$ in an analogous way, variable by variable. We repeat this for all the monomials in all the q_is. This way the sum $\sum_{i \in [\ell]} q_i p_i$ is rewritten as $\sum_{i \in [m]} c_i p_i$ 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 $\sum_{i \in [\ell]} c_i p_i$ as

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

To conclude we just consider all monomials $c - c_i$ as part of q'_i. ∎

Notice that, if $p_1, \ldots, p_\ell$ are polynomials with negative coefficients, then the Normal form for SA in the theorem above gets further simplified to

$$p_0 = \sum_{i=1}^{m} c p_i + \sum_{j=1}^{n} r_j (x_j^2 - x_j) + \sum_{j=1}^{n} r''_j (x_j + \bar{x}_j - 1) + q'_0 \,, \tag{5}$$

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

This is exactly what happens for the natural encoding of sets of clauses in the context of (semi)algebraic proof systems. A clause $C = \{ x_i, \neg x_j : i \in I,\ j \in J \}$ 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 $\geqslant 0$ in SA. In the algebraic context, we follow the common convention that a variable being 0 means it is true, while in the propositional contenxt 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 corresponding to its clauses.

Under this natural representation it is well-known that SA p-simulates resolution (see for instance [AO18, Lemma 3.5]) and NS with unary coefficients p-simulates tree-like resolution. Moreover, both p-simulations are known to be strict.

3 Weighted depth-d Frege and (semi)-algebraic proof systems

A *weighted Θ_d-cedent over $\mathbb{Z}$* is a pair $[\Gamma; w]$ where Γ is a Θ_d-cedent and $w \in \mathbb{Z}$. Given two weighted cedents $[\Gamma; w]$ and $[\Delta; z]$ we say that $[\Gamma; w]$ is a *weakening* of $[\Delta; z]$ if $\Gamma \supseteq \Delta$.

In this paper we only consider proof systems handling weighted depth-d formulas over $\mathbb{Z}$, although the definitions can be extended easily to weighted polynomials, linear inequalities etc.

$$\frac{[\Gamma, \varphi, \varphi; w]}{[\Gamma, \varphi; w]} \ (\textsc{contraction})$$

$$\frac{[\Gamma, \neg\varphi; w] \quad [\Gamma, \varphi; w]}{[\Gamma; w]} \ (\textsc{symmetric cut})$$

$$\frac{[\Gamma, \varphi; w] \qquad \text{for } \varphi \in \Phi}{[\Gamma, \bigwedge \Phi; w]} \ (\textstyle\bigwedge\textsc{-introduction})$$

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

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

$$\frac{}{[\varphi, \neg\varphi; w]} \ (\textsc{excluded middle})$$

$$\frac{[\Gamma; w]}{[\Gamma, \neg\varphi; w] \quad [\Gamma, \varphi; w]} \ (\textsc{split})$$

$$\frac{[\Gamma, \Phi; w]}{[\Gamma, \bigvee \Phi; w]} \ (\textstyle\bigvee\textsc{-introduction})$$

$$\frac{[\Gamma, \bigvee \Phi; w]}{[\Gamma, \Phi; w]} \ (\textstyle\bigvee\textsc{-elimination})$$

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

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

Figure 3.1: Inference rules of weighted depth-d Frege. The cedents $\Gamma, \Phi, \bigvee \Phi, \bigwedge \Phi, \varphi, \neg\varphi$ all are Θ_d-cedents, $u, w \in \mathbb{Z}$.

Definition 3.1 (weighted depth-d Frege)**.** A *weighted depth-d Frege* 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 $z > 0$, and either
   (2.1) all cedents in $\mathcal{L}_s \setminus \{[\Gamma; z]\}$ have positive weights ($\textsc{soundness}$-SA), or
   (2.2) all cedents in $\mathcal{L}_s \setminus \{[\Gamma; z]\}$ have positive weights and, moreover, they are also weakenings of cedents in $\mathcal{F}$ ($\textsc{soundness}$-NS).
3. each $\mathcal{L}_i$ is obtained from $\mathcal{L}_{i-1}$ applying one of the inference rules in Fig. 3.1 as <u>substitution</u> rules, i.e. removing the premises from $\mathcal{L}_{i-1}$ and adding the conclusions.

A weighted depth-d Frege *refutation* of $\mathcal{F}$ is a weighted depth-d Frege derivation of the empty cedent. The *size* of a weighted depth-d Frege 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 weighted depth-d Frege. In the system with weights in unary there are no applications of the $\textsc{fold}$/$\textsc{unfold}$ rules and the weighted cedents in $\mathcal{L}_1$ are given as a multiset, instead of $[\Gamma_i; w]$ we have $\{\underbrace{[\Gamma_i; 1], \dots, [\Gamma_i; 1]}_{w}\}$.

We refer to weighted depth-d Frege with the SOUNDNESS-SA condition, simply as weighted depth-d Frege. *Weighted Resolution* is weighted depth-0 Frege. This system comes essentially from [BL20; LR20b].

Remark 3.2. The rules in Fig. 3.1 are redundant, e.g. the SPLIT rule can be simulated using the others. Moreover, only one among FOLD/UNFOLD is enough.

We don't use a minimal set of rules just to highlight the natural symmetry among the rules and to have more freedom to write down weighted resolution proofs.

Remark 3.3. Restricting weighted depth-d Frege to have negative weights only in the INTRODUCTION/REMOVAL rules results in a system p-equivalent to weighted depth-d Frege. For instance, a way to p-simulate the rule $\frac{[\Gamma,\varphi;-1] \quad [\Gamma,\neg\varphi;-1]}{[\Gamma;-1]}$ is the following:

$$\cfrac{\cfrac{\cfrac{\cfrac{[\Gamma,\varphi;-1] \qquad [\Gamma,\neg\varphi;-1]}{[\Gamma,\varphi;-1] \qquad [\Gamma,\neg\varphi;-1] \qquad [\Gamma;1] \qquad [\Gamma;-1]}}{[\Gamma,\varphi;-1] \qquad [\Gamma,\neg\varphi;-1] \qquad [\Gamma,\varphi;1] \qquad [\Gamma,\neg\varphi;1] \qquad [\Gamma;-1]}}{[\Gamma,\neg\varphi;-1] \qquad [\Gamma,\neg\varphi;1] \qquad [\Gamma;-1]}}{[\Gamma;-1]\,.}$$

Moreover, weighted depth-d Frege is also p-equivalent to weighted depth-d Frege with all weights restricted to be powers of 2.

Remark 3.4. In the definition of weighted depth-d Frege (with the SOUNDNESS-NS/SA condition) the first property required is that $\mathcal{L}_1 = \{[\Gamma_1; w], \dots [\Gamma_m; w]\}$ where $w \in \mathbb{N}$. We could have required instead
(1) $\mathcal{L}_1 = \{[\Gamma_1; w_1], \dots [\Gamma_m; w_m]\}$ where $w_1, \dots, w_m \in \mathbb{N}$ or even
(2) $\mathcal{L}_1 = \{[\Delta_1; w_1], \dots [\Delta_m; w_m]\}$ with $w_1, \dots, w_m \in \mathbb{N}$ and for all $i \in [m]$, $[\Delta_i; w_i]$ weakening of $[\Gamma_i; w_i]$.

All the three possibilities would have resulted in p-equivalent systems. The reason is that, in the first case, we can always take $w = \max_{i\in[m]} w_i$. In the second case, given cedents Γ_i, Δ'_i, it is immediate to see that it is possible to infer in depth-d Frege from the weighted cedent $[\Gamma_i; w_i]$ a set S of weighted cedents containing $[\Gamma_i, \Delta'_i; w_i]$. Moreover, all cedents in S are weakenings of $[\Gamma_i; w_i]$. This proof is just a sequence of $|\Delta'_i|$ applications of the SPLIT rule.

Lemma 3.5. *For every $d \in \mathbb{N}$, the proof system weighted depth-d Frege is sound. The same is true for weighted depth-d Frege with the* SOUNDNESS-NS *condition.*

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

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

Suppose $\mathcal{F}$ has a weighted depth-d Frege refutation $(\mathcal{L}_1, \dots, \mathcal{L}_s)$. If $\mathcal{F}$ was satisfiable, then there would exist an assignment α satisfying all cedents in $\mathcal{F}$, hence $W(\mathcal{L}_1, \alpha) = 0$. Since $[\bot; w] \in \mathcal{L}_s$ for some $w > 0$ and $\mathcal{L}_s$ satisfies SOUNDNESS-NS or SOUNDNESS-SA, then

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

On the other hand, the inference rules of Fig. 3.1 guarantee that in the derivation $(\mathcal{L}_1, \dots, \mathcal{L}_s)$

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

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

We now prove the p-equivalences and some of the p-simulations summarized in Fig. 1.1, 1.2 and 1.3.

Proposition 3.6. *For all $d \in \mathbb{N}$, weighted depth-d Frege is p-equivalent to circular depth-d Frege.*

Proof. (sketch) Lemma 4 and 5 in [BL20] prove, in our language, that weighted Resolution is p-equivalent to circular Resolution. Their argument consider clauses, i.e. Θ_0-cedents, but it is immediate to adapt the argument to Θ_d-cedents. ∎

Theorem 3.7. *For every $d \in \mathbb{N}$, weighted depth-d Frege p-simulates depth-d Frege.*

Proof. (sketch) Derivations in depth-d Frege are special cases of circular depth-d Frege proofs, hence, by Proposition 3.6, weighted depth-d Frege p-simulates depth-d Frege.

Alternatively, and more directly, it is easy to see that, given a depth-d Frege refutation π it is possible to assign weights to Θ_d-cedents to respect the rules of weighted depth-d Frege. Basically the idea is to set $[\emptyset; 1]$ and then proceed bottom-up in π setting the weight of any Θ_d-cedent Γ looking at all the times it is used and summing the weights of those weighted cedents. ∎

Theorem 3.8. *For all $d \in \mathbb{N}$, unary weighted depth-d Frege with the* SOUNDNESS-NS *condition p-simulates tree-like depth-d Frege.*

This result is basically a generalisation of the proof that NS p-simulates tree-like Resolution.

Proof. (sketch) Consider a depth-d Frege derivation of Γ from a set of cedents $\Delta_1, \dots \Delta_\ell$. This proof is a tree T with cedents labeling its nodes. First, transform T "pushing all the weakenings towards the leaves". That is, whenever in a node v it is applied a WEAKENING $\frac{\Gamma}{\Gamma,\Gamma'}$, instead of doing this we add Γ' to all cedents in the subtree of T rooted in v. After doing this for all WEAKENINGS in T we obtain a depth-d Frege derivation of Γ from $\Delta_1, \dots \Delta_\ell$ with size just polynomially bigger than T and such that all WEAKENING rules are just applied immediately on $\Delta_1, \dots \Delta_\ell$ to get $\Delta'_1, \dots \Delta'_\ell$. Let T' be the tree corresponding to this proof pruned of the leaves $\Delta_1, \dots \Delta_\ell$, i.e. the leaves of T' have labels $\Delta'_1, \dots \Delta'_\ell$.

Now, give weight 1 to each cedent in T' and take an ordering on the vertices of T' respecting the depth, first the leaves, then nodes at depth 1, then depth 2 etc. Let $\mathcal{L}_1$ be the multiset $\{[\Delta'_1; 1], \dots, [\Delta'_\ell; 1]\}$. Then $\mathcal{L}_2$ is the multiset obtained removing the premises of the first node v_1 of depth 1 in T' and adding the label of it. Then form $\mathcal{L}_2$ remove the premises of the second node v_2 in T' with depth 1 and add the label of v_2 to form $\mathcal{L}_3$. Proceeding in this orderly way, from nodes f low depth to nodes of higher depth in T' we can process all the nodes of T', finishing in $\mathcal{L}_s$ which will contain the weighted cedent labeling the root of T'. By Remark 3.4 this is p-equivalent to a refutation in weighted depth-d Frege with the SOUNDNESS-NS condition. ∎

One of the reasons we introduced weighted depth-d Frege is that, varying the soundness condition, it gives a characterisation of distinct (semi)-algebraic proof systems in a more logic language.

Theorem 3.9. 1. *(Unary) SA is p-equivalent to (unary) weighted Resolution.*

2. *(Unary) NS is p-equivalent to (unary) weighted Resolution with the* SOUNDNESS-NS *condition.*

The part of this theorem for SA is already known: weighted Resolution is p-equivalent to circular Resolution [BL20] and circular Resolution is p-equivalent to SA [AL19]. As far as we know, there is no natural restriction of circular Resolution characterising unary SA nor (unary) NS. Here we present a proof covering all cases at the same time, modulo minor changes. Hence, in particular, we re-prove the result for SA in a more direct way.[3]

Proof. 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 viceversa 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 $F = \{C_1, \dots, C_\ell\}$. By Theorem 2.9 a NS refutation of F can be supposed to have the form

$$- z - \sum_{i=1}^{\ell'} w_i m_i M(C_i) = - \sum_{i=1}^{\ell} w M(C_i) + \sum_{j=1}^{n} r_j (x_j^2 - x_j) + \sum_{j=1}^{n} r'_j (x_j + \bar{x}_j - 1) , \tag{6}$$

for some $z > 0$, $w, w_i \geqslant 0$, polynomials r_j, r'_j, monomials m_i and $\ell' \geqslant \ell$. If $j > \ell$, C_j is one among the clasues $C_1, \dots, C_\ell$. 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. (6) 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 $-\sum_{i=1}^{\ell} w M(C_i)$. 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

$$\mathcal{L}_{j+1} = \mathcal{L}_j \cup \{[C(m), \neg x_j, \neg x_j; -w], [C(m), \neg x_j; w]\} . \tag{7}$$

We need to justify how to obtain $\mathcal{L}_{i+1}$ from $\mathcal{L}_i$ applying the rules of Fig. 3.1. 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

$$\mathcal{L}_{k+1} = \mathcal{L}_k \cup \{[C(m), \neg x_j; -w],\ [C(m), x_j; -w],\ [C(m); w]\} . \tag{8}$$

Again, we need to justify how to obtain $\mathcal{L}_{k+1}$ from $\mathcal{L}_k$ applying the rules of Fig. 3.1. Again, this is immediate. We interleave intermediate multisets between $\mathcal{L}_k$ and $\mathcal{L}_{k+1}$

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

[3] If circular Resolution was defined with the function witnessing the balances given explicitly as part of the proof, then the p-equivalence from [BL20] would have worked also in the unary case.

After we finish this process let $\mathcal{L}_{s'}$ the multiset we got. We exhausted all the terms from the RHS of eq. (6) and all the monomials, except the ones in the LHS of eq. (6), 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 and hence concludes the proof that weighted Resolution with the SOUNDNESS-NS condition p-simulates NS.

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

$$-z - \sum_{i \in J} w'_i m'_i = -\sum_{i=1}^{\ell} wM(C_i) + \sum_{j=1}^{n} r_j(x_j^2 - x_j) + \sum_{j=1}^{n} r'_j(x_j + \bar{x}_j - 1) , \tag{9}$$

for some $z > 0$, $w_i, w'_i \geqslant 0$, 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 SOUNDNESS-SA.

The other direction of the p-simulations is easier. Given a weighted Resolution refutation $(\mathcal{L}_1, \dots, \mathcal{L}_s)$ 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 a EXCLUDED 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 of the other rules let $S_{i+1} = S_i$.[4]

It is immediate to see that S_{i+1} constructed following the procedures above has the property 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. ∎

[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 weighted Resolution, all the cancellations between weighted clauses are done explicitly by applications of some rules.

4 The pigeonhole principle and unary NS/SA

In this section we prove the p-simulations relative to the *unary* parts of Fig. 1.1, 1.2 and 1.3.

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

$$\mathtt{PHP}^m_n = \{\, \{p_{i,1}, \dots, p_{i,n}\} \,:\, i \in [m] \,\} \cup \{\, \{\neg p_{i,j}, \neg p_{i',j}\} \,:\, i, i' \in [m] \text{ distinct}, j \in [n] \,\} .$$

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

$$\{\, \{\neg p_{i,j}, \neg p_{i,j'}\} \,:\, i \in [m] \quad j, j' \in [n] \text{ distinct} \,\} \qquad \text{(functionality axioms)} \tag{10}$$

and the set

$$\{\, \{\, p_{i,j} \,:\, i \in [m] \,\} \,:\, j \in [n] \,\} \qquad \text{(onto axioms)} . \tag{11}$$

Given a bipartite graph $G = (P \cup H, E)$ with $|P| = m$ and $|H| = n$, the *graph* pigeonhole principle $\mathtt{PHP}^m_n(G)$ is the formula $\mathtt{PHP}^m_n$ 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 $\mathtt{PHP}^m_n$ where it appears and remove all clauses of $\mathtt{PHP}^m_n$ containing $\neg p_{i,j}$. The onto-functional *graph* pigeonhole principle $\mathtt{ofPHP}^m_n(G)$ is defined in the same way.

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

$$\sum_{j \in [n+1]} \sum_{i \in [n]} p_{i,j} - (n+1) \geqslant 0 \tag{12}$$

$$n - \sum_{i \in [n]} \sum_{j \in [n+1]} p_{i,j} \geqslant 0 . \tag{13}$$

Then, sum the two inequalities to get $-1 \geqslant 0$. For completeness, a proof of (12) and (13) is in Appendix A. The same argument can be easily adapted to show the results for unary NS. Moreover, for a bipartite graph G with maximum degree d, $\mathtt{PHP}^{n+1}_n(G)$ has degree-d unary SA refutations and $\mathtt{ofPHP}^{n+1}_n(G)$ has degree-d unary NS refutations.

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

Theorem 4.2. *For every d, tree-like Res(d) + $\mathtt{PHP}^{n+1}_n(G)$ 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 Res(d) + $\mathtt{PHP}^{n+1}_n(G)$ derivations is 5.*

Notice that tree-like Res(n) is tree-like depth-1 Frege. The $\mathtt{PHP}^{n+1}_n(G)$ in this result could be replaced by the propositional formulas $\mathtt{HEX}$, $\mathtt{SINK}$ and $\mathtt{2SINK}$ from [Bus06]. This would result in an analogous p-simulation. The proof of this result is loosely inspired by the proof of [BBIMM18, Theorem 4].

Proof. We use the characterisation of SA given by Theorem 3.9. Let $(\mathcal{L}_1, \dots, \mathcal{L}_s)$ be a weighted resolution refutation of some set of clauses $F = \{C_1, \dots, C_m\}$. Since the weights are in unary, all the weights in π are just ± 1. In this proof there will be no application of the FOLD/UNFOLD rules. Without loss of generality we can assume that all 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 at most d literals (*d-terms*) such that we have small tree-like Res(d) derivations of the cedents

$$\{\, p_{\alpha,\beta} \,:\, \beta \in H \,\} \quad \text{for all } \alpha \in P$$

and

$$\{\neg p_{\alpha,\beta}, \neg p_{\alpha',\beta}\} \quad \text{for all } \beta \in H \text{ and distinct } \alpha, \alpha' \in P.$$

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.

To define $p_{\alpha,\beta}$ we introduce some terminology. Given $\alpha \in P$, let $\alpha = [C_\alpha; w_\alpha]$ and let i_α be the unique index such that α belong to $\mathcal{L}_{i_\alpha}$; 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 applied 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 and 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 $\mathcal{L}_{i_\alpha+1}$).

Informally, we want the formulas $p_{\alpha,\beta}$ to express that if the clause C_α is true then α flies to itself (as a hole). That is, we set $p_{\alpha,\alpha}$ to be the formula $\bigvee C_\alpha$.

If C_α is false and its weight is $+1$, it flies to the false premise C_β used to derive it or to its appearing sibling. 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 or the opposite.

If C_α is an initial clause α always flies to itself. So we set $p_{\alpha,\alpha} = x \vee \neg x$.

If C_α is false and the weight of C_α is -1 then α flies to its copy C_β in the direction of the proof, or to its disappearing sibling. The way to define $p_{\alpha,\beta}$ is analogous as before.

Formally,

$$p_{\alpha,\beta} = \begin{cases} x \vee \neg x & \text{if } \alpha = \beta \text{ and } \alpha \in \mathcal{L}_1 \\ \bigvee C_\alpha & \text{if } \alpha = \beta \text{ and } \alpha \notin \mathcal{L}_1 \\ \neg \bigvee C_\beta & \text{if } \begin{cases} \alpha \text{ is a positive-copy of } \beta \\ \beta \text{ is a SYMM.CUT-premise of } \alpha \\ \beta \text{ is a CONTRACTION-premise of } \alpha \\ \alpha, \beta \text{ are appearing siblings and } w_\alpha = 1 \\ \beta \text{ is a negative-copy of } \alpha \\ \alpha, \beta \text{ are disappearing siblings and } w_\alpha = -1 \end{cases} \\ \neg \bigvee C_\alpha & \text{if } \beta \text{ is a SPLIT-premise of } \alpha \\ \bot & \text{otherwise}\,. \end{cases}$$

The totality axioms $\{\, p_{\alpha,\beta} \,:\, \beta \in H \,\}$ are easily derivable in tree-like Res(d) from the initial clauses $C_1, \dots, C_m$. 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

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

  Let $\bigvee C_\alpha = \bigvee C'_\alpha \vee \ell$ and $\bigvee C_\beta = \bigvee C'_\alpha \vee \ell \vee \ell$ for some clause C'_α and some literal ℓ. But then it is immediate to derive this in $\mathrm{Res}(d)$ in height 4:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\bigvee C'_\alpha \vee \ell \vee \ell,\ \neg(\bigvee C'_\alpha \vee \ell \vee \ell)}}{C'_\alpha, \ell, \ell,\ \neg(\bigvee C'_\alpha \vee \ell \vee \ell)}}{C'_\alpha, \ell,\ \neg(\bigvee C'_\alpha \vee \ell \vee \ell)}}{\bigvee C'_\alpha \vee \ell,\ \neg(\bigvee C'_\alpha \vee \ell \vee \ell)}$$

- 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 $\beta,\ \beta'$. 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 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 weakenings 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 $\{\neg p_{\alpha,\beta},\ \neg p_{\alpha',\beta}\}$ are 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 $\{\neg \bigvee C_\beta, \bigvee C_\beta\}$ or $\{\neg \bigvee C_\beta,\ \bigvee C_\alpha\}$ 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 there are no axioms of the form $\{\neg p_{\alpha,\beta},\ \neg p_{\alpha',\beta}\}$ since in β can only fly two pigeons, β itself and the copy of β from the previous layer (or its disappearing sibling).

– If $w_\beta = 1$, having the variables $p_{\alpha,\beta}$ and $p_{\alpha',\beta}$ distinct from $\bot$ 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 Res(d) being a weakening of $x \vee \neg x$.

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

Remark 4.3. The construction of the formulas $p_{\alpha,\beta}$ in the previous proof satisfies clearly the *functionality axioms* of $\mathtt{ofPHP}_n^{n+1}(G)$ but it does not satisfy the *onto* axioms. The reason is that 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, but if $\mathcal{L}_s$ satisfies the SOUNDNESS-NS condition 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 4.4. *For every d, tree-like Res(d) + $\mathtt{ofPHP}_n^m(G)$ 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 Res(d) + $\mathtt{ofPHP}_n^m(G)$ derivations is 5.*

Proof. (sketch) We use the characterisation of unary NS given by Theorem 3.9 and we reason basically as in Theorem 4.2. We know that the problematic clauses in $\mathcal{L}_s$ are weakenings of initial axioms or several copies of $[\emptyset; 1]$. We can define the formula $p_{\alpha,\beta}$ as in Theorem 4.2. Now the onto axioms for the holes in $\mathcal{L}_s$ become weakenings of inital clauses except for the holes corresponding to the copies of $[\emptyset; 1]$. Those, as in the case of SA are copied in the layer $\mathcal{L}_{s+1}$. With the exception that 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 $\mathtt{PHP}_n^{n+1}(G)$ we use $\mathtt{ofPHP}_n^m(G)$. ∎

The proofs of Theorem 4.2 and 4.4 will work, almost without changes, if instead of clauses we have Θ_d-cedents.

Theorem 4.5. *For every $d \in \mathbb{N}$, tree-like depth-$(d+1)$ Frege + $\mathtt{PHP}_n^{n+1}(G)$ p-simulates unary weighted depth-d Frege, where G is restricted to bipartite graphs of degree at most 3 and the height of the tree-like depth-$(d+1)$ Frege + $\mathtt{PHP}_n^{n+1}(G)$ derivations is 5.*

Theorem 4.6. *For every $d \in \mathbb{N}$, tree-like depth-$(d+1)$ Frege + $\mathtt{ofPHP}_n^m(G)$ p-simulates unary weighted depth-d Frege and the condition SOUNDNESS-NS, where G is restricted to bipartite graphs of degree at most 3 and the height of the tree-like depth-$(d+1)$ Frege + $\mathtt{ofPHP}_n^m(G)$ derivations is 5.*

A consequence of the previous theorems is the following corollary.

Corollary 4.7. *For every $d \in \mathbb{N}$, tree-like depth-$(d+1)$ Frege + $\mathtt{PHP}_n^{n+1}(G)$ p-simulates tree-like depth-d Frege, where G is restricted to bipartite graphs of degree 3 and the height of the tree-like depth-$(d+1)$ Frege + $\mathtt{PHP}_n^{n+1}(G)$ derivations is 5.*

A priori, it is not even immediately clear why tree-like depth-$(d+1)$ Frege + $\mathtt{PHP}_n^{n+1}(G)$ should be complete, when we restrict the height of the derivations to 5. But it is, and it is strong enough to p-simulate tree-like depth-d Frege.

Proof. It follows from Theorem 4.5, Theorem 2.3 and Theorem 3.7. ■

We conclude this section with a couple of separations/lower-bounds.

Proposition 4.8. *For every $d = o(\frac{\log\log n}{\log\log\log n})$, depth-$d$ Frege does not p-simulate unary weighted depth-d Frege.*

Proof. Any refutation of $\mathtt{PHP}_n^{n+1}$ in depth-d Frege must have size at least $2^{n^{(1/6)^d}}$ (see for instance [UF96]). $\mathtt{PHP}_n^{n+1}$ has polynomial size unary SA refutations and hence it has polynomial size refutations in unary weighted depth-d Frege . ■

Definition 4.9 (MOD_2 principle)**.** Given $n \in \mathbb{N}$, the *MOD_2-principle* is the set of cedents in the variable $x_{i,j}$ for $i \neq j \in S$

$$\begin{aligned}\mathtt{MOD}_2^n = &\{\, \{x_{i,1}, \dots, x_{i,i-1}, x_{i,i+1}, \dots, x_{i,2n+1}\} \ :\ i \in [2n+1]\,\}\\ &\cup \{\, \{\neg x_{i,j}, \neg x_{i',j}\} \ :\ i, i' \in [2n+1] \text{ distinct}, j \in [2n+1]\,\}\,.\end{aligned}$$

Corollary 4.10. *Given $n \in \mathbb{N}$ and $d = o(\frac{\log\log n}{\log\log\log n})$, $\mathtt{MOD}_2^n$ has not polynomial-size unary weighted depth-d Frege refutations.*

Proof. Any refutation of $\mathtt{MOD}_2^n$ in depth-d Frege + $\mathtt{PHP}$ must require size at least $\exp(n^{\Omega(1/d4^d)})$ [BP96, Theorem 4]. By Theorem 4.5 depth-$(d+1)$ Frege + $\mathtt{PHP}$ p-simulates unary weighted depth-d Frege the lower bound follows: $\mathtt{MOD}_2^n$ requires unary weighted depth-d Frege refutations of size $\exp(n^{\Omega(1/(d+1)4^d)})$. ■

Definition 4.11 (*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 need to enforce injectivity:

$$\mathtt{bit\text{-}PHP}_n = \{\, \{b_{i1}^{1-h_1}, \dots, b_{ik}^{1-h_k}, b_{i'1}^{1-h_1}, \dots, b_{i'k}^{1-h_k}\} \ :\ i \neq i' \in [n+1],\ h \in [n]\ h = (h_1, \dots, h_k)_2]\,\}\,,$$

where $(h_1, \dots, h_k)_2$ is the binary expansion of the hole h and $b_{ij}^{h_j} = b_{ij}$ if $h_j = 1$ and $b_{ij}^{h_j} = \neg b_{ij}$ if $h_j = 0$.

Corollary 4.12. SA *does not p-simulate tree-like depth-1 Frege + $\mathtt{PHP}_n^{n+1}$.*

Proof. $\mathtt{bit\text{-}PHP}_n$ does not have polynomial-size SA refutations [DGM20]. To prove $\mathtt{bit\text{-}PHP}_n$ in tree-like depth-1 Frege + $\mathtt{PHP}_n^{n+1}$ we use the substitution $p_{ij} = b_{i1}^{j_1} \wedge \dots \wedge b_{ik}^{j_k}$ where $j = (j_1, \dots, j_k)_2$. For $i \neq i' \in [n+1]$ and $j \in [n]$, $\{\neg p_{ij}, \neg p_{i'j}\}$ 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 EXLUDED 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 *weighted* pigeonhole principle and SA

In this section we generalise the constructions given for unary SA/NS and unary weighted depth-d Frege to systems with binary weights/coefficients. We prove all remaining p-simulations in Fig. 1.1, 1.2 and 1.3.

It is not clear at all how to adapt Theorem 4.5 to show that tree-like depth-1 Frege + $\mathtt{PHP}_n^{n+1}(G)$ p-simulates SA. For this reason we introduce a new combinatorial principle, a *weighted* version of $\mathtt{PHP}$.

The weighted pigeonhole principle maps n^2+1 into n^2. First we partition both sets of pigeons and holes into n pieces of equal size, except for the first pigeon-part that also contains the pigeon n^2+1. Formally, given $n \in \mathbb{N}$ and $\ell \in [n]$, let $W_1 = \{n^2+1\} \cup [n]$ and, for $\ell \in \{2, \ldots, n\}$ let $W_\ell = [(\ell-1)n, \ell n] = \{(\ell-1)n+1, \ldots, \ell n\}$. Let $W_0 = W_{n+1} = \emptyset$.

Definition 5.1 (*weighted* pigeonhole principle, `wtPHP`)**.** The *weighted* pigeonhole principle has variables x_{ij} for each $i \in [n^2+1]$ and each $j \in [n^2]$. The formula $\mathtt{wtPHP}^{n^2+1}_{n^2}$ has the following clauses. For every $\ell \in [n]$, every pigeon $p \in W_\ell$ it has clauses

$$\begin{aligned}
\{\neg x_{pj}\} \quad & \text{for all}\, j \notin W_{\ell-1} \cup W_\ell \cup W_{\ell+1} \\
\{\, x_{pj} \,:\, j \in [n^2]\,\} \quad & \\
\{\, \neg x_{pj},\ x_{pj'} \,:\, j' \in W_{\ell-1} \setminus \{j\}\,\} \quad & \text{for all } \ j \in W_{\ell-1},\ j \neq n^2+1 \\
\{\neg x_{pj_1},\ \neg x_{pj_2},\ \neg x_{pj_3}\} \quad & \text{for all } \text{ distinct } j_1, j_2, j_3 \in W_{\ell-1},\ j_1, j_2, j_3 \neq n^2+1 \qquad (\star)
\end{aligned}$$

and every hole $h \in W_\ell$ $(h \neq n^2+1)$ it has clauses

$$\begin{aligned}
\{\neg x_{ih},\ \neg x_{i'h}\} \quad & \text{for all } \text{ distinct } i \in W_\ell \cup W_{\ell+1} \text{ and } \ i' \in W_{\ell-1} \cup W_\ell \cup W_{\ell+1} \\
\{\neg x_{i_1h},\ \neg x_{i_2h},\ \neg x_{i_3h}\} \quad & \text{for all } \text{ distinct } i_1, i_2, i_3 \in W_{\ell-1}\,.
\end{aligned}$$

The intended meaning of the variable x_{ij} for $i \in W_\ell$ depends on j: for $j \in W_\ell \cup W_{\ell+1}$, $x_{ij} = 1$ means "*the pigeon i flies to j with weight 2^ℓ*"; for $j \in W_{\ell-1}$, $x_{ij} = 1$ means "*i flies to j with weight $2^{\ell-1}$*" and i needs to fly somewhere else in $W_{\ell-1}$ with the same weight too. If $j \notin W_{\ell-1} \cup W_\ell \cup W_{\ell+1}$ then $x_{ij} = 0$.
Similar to the case of `PHP`, given a bipartite graph $G = (P \cup H, E)$ with $|P| = n^2+1$ and $|H| = n^2$, the *graph* weighted pigeonhole principle $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ is the formula $\mathtt{wtPHP}^{n^2+1}_{n^2} \restriction_\alpha$ where α is a partial restriction mapping $x_{i,j} = \perp$ for all $(i,j) \notin E$.

Remark 5.2. The clauses in $(\star)$ are not needed to have an unsatisfiable formula, but, they are useful to have a short proof of this principle in SA. Indeed Lemma B.1 uses them. When considering $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$, the graphs G we need to consider turn out to always have at most 2 edges of the form $(p,j),(p,j')$ with $p \in W_\ell$ and $j, j' \in W_{\ell-1}$. Hence, for those graphs G, the axioms in $(\star)$ will always be satisfied: one of the variables $x_{pj_1},\ x_{pj_2},\ x_{pj_3}$ is always set to $\perp$.

Remark 5.3. We defined $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ to have $W_1, \ldots, W_n$ of some particular form, in particular all of size n and W_1 to be of size $n+1$. It is easy to see that allowing $W_1, W_2, \ldots, W_n$ to be disjoint and of size *at most* n (resp. at most $n+1$ for W_1) does not make a more general principle. Basically we could just add "padding" to all W_js till they have the same size and change G to a graph that forces the new vertices in the padding to be mapped to themselves. We call the $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ where the sets $W_1, \ldots, W_n$ are disjoint and of size *at most* n (resp. at most $n+1$ for W_1) the $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ *without padding*.

Perhaps, it is not immediately clear that the formula $\mathtt{wtPHP}^{n^2+1}_{n^2}$ is unsatisfiable. Informally, a way to see this is to notice that for every pigeon p (say $p \in W_\ell$) the axioms of $\mathtt{wtPHP}^{n^2+1}_{n^2}$ can be interpreted to imply the weight flying away from p is at least 2^ℓ and, for every hole h (say $h \in W_\ell \cap [n^2]$), the weight it can accomodate is at most 2^ℓ. So the holes can, in total, accomodate 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 $2 + \sum_{\ell \in [n]} n2^\ell$.

Theorem 5.4. *The formula $\mathtt{wtPHP}^{n^2+1}_{n^2}$ has polynomial-size* SA *refutations. Moreover, for every bipartite graph $G = (P \dot\cup H, E)$ with $|P| = n^2+1$, $|H| = n^2$ and degree d, the formula $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ has* SA*-refutations of degree d.*

Proof. (sketch) First observe that the axioms imply, for every $i \in [n^2+1]$ with $i \in W_\ell$, the inequality

$$2\sum_{\substack{j\in W_\ell\cup W_{\ell+1}\\ j\neq n^2+1}} x_{ij} + \sum_{\substack{j\in W_{\ell-1}\\ j\neq n^2+1}} x_{ij} - 2 \geqslant 0\,, \tag{14}$$

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

$$2 - 2\sum_{i\in W_\ell\cup W_{\ell+1}} x_{ij} - \sum_{i\in W_{\ell-1}} x_{ij} \geqslant 0\,. \tag{15}$$

This is proved in Appendix B.

To conclude we want to sum (appropriate multiples of) (14) and (15) in a way that all the variables in equation (14) cancel with variables in (15) and after all the cancelations is just get some negative constant:

$$\sum_{\substack{\ell\in[n]\\ i\in W_\ell}} 2^\ell \left(2\sum_{\substack{j\in W_\ell\cup W_{\ell+1}\\ j\neq n^2+1}} x_{ij} + \sum_{\substack{j\in W_{\ell-1}\\ j\neq n^2+1}} x_{ij} - 2 \right) + \sum_{\substack{\ell\in[n]\\ j\in W_\ell\\ j\neq n^2+1}} 2^\ell \left(2 - 2\sum_{i\in W_\ell\cup W_{\ell+1}} x_{ij} - \sum_{i\in W_{\ell-1}} x_{ij} \right) \geqslant 0\,. \tag{16}$$

Consider a variable x_{ij} in (16), with $i \in W_\ell$. If $j \in W_\ell$, the coefficient multiplying x_{ij} is $2^\ell \cdot 2 - 2^\ell \cdot 2 = 0$. If $j \in W_{\ell+1}$, the coefficient multiplying x_{ij} is $2^\ell \cdot 2 - 2^{\ell+1} = 0$. If $j \in W_{\ell-1}$, the coefficient multiplying x_{ij} is $2^\ell - 2 \cdot 2^{\ell-1} = 0$. That is, all the variables x_{ij} cancel out in (16).

The constants in (16) sum to

$$\begin{aligned} -2\sum_{\substack{\ell\in[n]\\ i\in W_\ell}} 2^\ell + 2\sum_{\substack{\ell\in[n]\\ j\in W_\ell\\ j\neq n^2+1}} 2^\ell &= \sum_{\ell\in[n]} 2^{\ell+1}(-|W_\ell| + |W_\ell \cap [n^2]|) \\ &= -2\,. \end{aligned}$$

That is, the sum in (16) after cancelations reduces to the trivial contradiction $-2 \geqslant 0$. ∎

Notice that there is a different way to use (14) and (15) to infer a contradiction (priv. comm. by Sam Buss). For a generic $\ell \in [n]$ we can sum (14) and (15) just for all pigeons and holes in W_ℓ:

$$\sum_{i\in W_\ell} \left(2\sum_{\substack{j\in W_\ell\cup W_{\ell+1}\\ j\neq n^2+1}} x_{ij} + \sum_{\substack{j\in W_{\ell-1}\\ j\neq n^2+1}} x_{ij} - 2 \right) + \sum_{\substack{j\in W_\ell\\ j\neq n^2+1}} \left(2 - 2\sum_{i\in W_\ell\cup W_{\ell+1}} x_{ij} - \sum_{i\in W_{\ell-1}} x_{ij} \right) \geqslant 0\,. \tag{17}$$

Let

$$\mathsf{Net}(\ell) = \sum_{\substack{i\in W_\ell\\ j\in W_{\ell+1}\\ j\neq n^2+1}} x_{ij} - \sum_{\substack{i\in W_{\ell+1}\\ j\in W_\ell\\ j\neq n^2+1}} x_{ij}\,.$$

After cancelations, the inequality in (17), depending on ℓ, simplifies to

$$\begin{cases} -\mathsf{Net}(n-1) \geqslant 0 \\ -2 + 2\mathsf{Net}(1) \geqslant 0 \\ 2\mathsf{Net}(\ell) - \mathsf{Net}(\ell-1) \geqslant 0 \qquad \text{for } 1 \leqslant \ell < n-1\,. \end{cases} \tag{18}$$

The last inequality allows us to infer that, for each ℓ, $\mathsf{Net}(\ell) > 0$. This is a contradiction when $\ell = n-1$.

Remark 5.5. The system of inequalities in (18) does not have polynomial-size unary SA refutations. This can be seen by a minor modification of the techniques in [Hak21]. Hence this set of polynomial inequalities separates SA and unary SA. Recently, Mika Göös (together with Alexandros Hollender, Siddhartha Jain, Gilbert Maystre, William Pires, Robert Robere, and Ran Tao) showed that SA and unary SA are separated also by polynomial inequalities encoding propositional formulas (priv. comm.). In other words, SA and unary SA are not p-equivalent.

It is easy to see that depth-1 Frege + $\mathtt{wtPHP}$ proves $\mathtt{PHP}$ in polynomial size. We don't know if the opposite is true, but we conjecture it is not.

Conjecture 5.6. *For every constant d, the formula $\mathtt{wtPHP}$ does not have polynomial size refutations in depth-d Frege + $\mathtt{PHP}$.*

The conjecture above implies, via Theorem 4.2 and Theorem 4.5, not only that $\mathtt{wtPHP}^{n^2+1}_{n^2}$ is hard to refute in unary SA but even in unary weighted depth-d Frege at least for constant d.

We now prove a sort of converse of Theorem 5.4.

Theorem 5.7. *For every $d \in \mathbb{N}$, tree-like Res(d)+$\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ p-simulates degree-d SA, where G is restricted to bipartite graphs of degree at most 3 and the tree-like Res(d) + $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ derivations have height 5.*

As for Theorem 4.2 the choice of the principle here could be substituted by a weighted version of $\mathtt{SINK}$ or some weighted version of the *flow tautologies*.

Proof. The structure of the proof is similar to the proof of Theorem 4.2. By Theorem 3.9 it is enough to prove the result for weighted Resolution. Let $\pi = \mathcal{L}_1, \ldots, \mathcal{L}_s$ be a weighted resolution refutation of a set of clauses $\{C_1, \ldots, C_m\}$. 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 FOLD/UNFOLD rules have positive weights. Moreover, since π is a refutation, we can assume $[\emptyset; 1] \in \mathcal{L}_s$, indeed if the last layer of π had $[\emptyset; 2^z]$ for some $z \geqslant 0$ using the UNFOLD rule we can reduce to a proof just slightly longer with last layer containing $[\emptyset; 1]$.

We want to define a substitution instance of $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ without padding (see Remark 5.3) such that we have shallow Res(d) derivations of it.

Let $S+1$ be the size of π, let $\mathcal{L}_{s+1} = \{[\emptyset; 1]\}$ and let $W_1, \ldots, W_S$ be a partition of $\mathcal{L}_1 \cup \cdots \cup \mathcal{L}_{s+1}$ according to the weight, i.e. all the weighted clauses in W_j have weight 2^{j-1} or -2^{j-1}. By assumption all those multisets have size at most S except W_1 that has size at most $S+1$. Let $W_0 = W_{S+1} = \emptyset$.

Let P be the multiset $W_1 \cup \cdots \cup W_S$ and $H = W_1 \cup \cdots \cup W_S \setminus \mathcal{L}_{s+1}$. Now, for all $\ell \in [S]$ and all $\alpha \in W_\ell$ and $\beta \in W_\ell \setminus \mathcal{L}_{s+1}$ we want to define $\bigwedge$-formulas $x_{\alpha,\beta}$ such that we can easily derive from $C_1, \ldots, C_m$ the cedents

$$\{\neg x_{\alpha\beta}\} \quad \text{for all } \beta \notin W_{\ell-1} \cup W_\ell \cup W_{\ell+1} \tag{19}$$

$$\{\, x_{\alpha\gamma} \,:\, \gamma \in H \,\} \tag{20}$$

$$\{\, \neg x_{\alpha\gamma}, x_{\alpha\gamma'} \,:\, \gamma' \in W_{\ell-1} \setminus \{\gamma\} \,\} \quad \text{for all } \gamma \in W_{\ell-1} \tag{21}$$

$$\{\neg x_{\alpha\gamma_1},\ \neg x_{\alpha\gamma_2},\ \neg x_{\alpha\gamma_3}\} \quad \text{for all distinct } \gamma_1, \gamma_2, \gamma_3 \in W_{\ell-1}\,, \tag{22}$$

$$\{\neg x_{\gamma\beta},\ \neg x_{\gamma'\beta}\} \quad \text{for all distinct } \gamma \in W_\ell \cup W_{\ell+1} \text{ and } \gamma' \in W_{\ell-1} \cup W_\ell \cup W_{\ell+1} \tag{23}$$

$$\{\neg x_{\gamma_1\beta},\ \neg x_{\gamma_2\beta} \vee \neg x_{\gamma_3\beta}\} \quad \text{for all distinct } \gamma_1, \gamma_2, \gamma_3 \in W_{\ell-1}\,, \tag{24}$$

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), if it is false and its weight is positive, it flies to *all* the false premises used to derive it 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. But those types of collisions are exactly the ones we are allowed to have in the $\texttt{wt}\,\texttt{PHP}^{n^2+1}_{n^2}(G)$ principle.

Given $\alpha \in \pi \cup \mathcal{L}_{s+1}$ let i_α be the unique index such that α belong to $\mathcal{L}_{i_\alpha}$ and w_α is 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 $\mathcal{L}_{i_\alpha}$ there is applied one of the inference rules of Fig. 3.1 with β one of the premises and α one of the conclusions. It is a *UNFOLD-premise* if β is a premise of α and the rule applied is a UNFOLD rule. The rest of terminology is the same as in the proof of Theorem 4.2.

Using the terminology from Theorem 4.2 then the definition of $x_{\alpha,\beta}$ is exactly the same as the definition of $p_{\alpha,\beta}$ just that now we have more inference rules

$$x_{\alpha,\beta} = \begin{cases} \{x, \neg x\} & \text{if } \alpha = \beta \text{ and } \alpha \in \mathcal{L}_1 \\ \bigvee C_\alpha & \text{if } \alpha = \beta \text{ and } \alpha \notin \mathcal{L}_1 \\ \neg \bigvee C_\beta & \text{if } \begin{cases} \alpha \text{ is a positive-copy of } \beta \\ \beta \text{ is a } \textsc{symm.cut}\text{-premise of } \alpha \\ \beta \text{ is a } \textsc{contraction}\text{-premise of } \alpha \\ \beta \text{ is a } \textsc{fold}\text{-premise of } \alpha \\ \beta \text{ is a } \textsc{unfold}\text{-premise of } \alpha \\ \alpha, \beta \text{ are appearing siblings and } w_\alpha > 0 \\ \beta \text{ is a negative-copy of } \alpha \\ \alpha, \beta \text{ are disappearing siblings and } w_\alpha < 0 \end{cases} \\ \neg \bigvee C_\alpha & \text{if } \beta \text{ is a } \textsc{split}\text{-premise of } \alpha \\ \bot & \text{otherwise}\,. \end{cases}$$

The axioms that require a slightly different argument from the proof of Theorem 4.2 are (21)–(24). The axiom (21) 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

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

The axiom (22) is always a weakening of $\top$ since all inference rules have at most 2 premises and hence at least one among the variables $x_{\alpha\gamma_1}, x_{\alpha\gamma_2}, x_{\alpha\gamma_3}$ is $\bot$.

The axioms in (23) cover all the cases of the injectivity as in Theorem 4.2 with the new case when β is a UNFOLD-premise. Only one among γ and γ' can be a conclusion of β, the other variable is set to $\bot$ and hence, again the axioms in (23) are weakening of $\top$.

The axiom (24) is always a weakening of $\top$ since even in the UNFOLD rule there are two conclusions, hence one among the variables $x_{\gamma_1\beta}, x_{\gamma_2\beta}, x_{\gamma_3\beta}$ is always $\bot$.

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

Remark 5.8. The construction of the formulas $x_{\alpha,\beta}$ in the previous proof does not satisfy the *onto* axioms

$$\{\, x_{\alpha\gamma} \,:\, \alpha \in P \,\}\,. \tag{25}$$

The reason is the same we had for $\mathtt{PHP}$ 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, but if $\mathcal{L}_s$ satisfies the SOUNDNESS-NS condition we can adapt the definition of $x_{\alpha,\beta}$ in the proof of Theorem 5.7 to satisfy the *onto* axioms of the weighted pigeonhole principle (as in eq. (25)).

Let $\mathtt{of\text{-}wtPHP}^{n^2+1}_{n^2}$ be the set of cedents

$$\mathtt{of\text{-}wtPHP}^{n^2+1}_{n^2} = \mathtt{wtPHP}^{n^2+1}_{n^2} \cup \{ \{ x_{ih} \,:\, i \in [n^2+1] \} \,:\, h \in [n^2] \} \,.$$

For the $\mathtt{of\text{-}wtPHP}^{n^2+1}_{n^2}$ we place the pigeon n^2+1 not in W_1 but we allow it to be in any of the $W_1, \ldots, W_n$.

It is immediate to see that $\mathtt{of\text{-}wtPHP}^{n^2+1}_{n^2}$ has polynomial-size NS refutations. A minor adaptation of Theorem 5.4 will prove this.

Theorem 5.9. *For every d, tree-like Res(d) + $\mathtt{of\text{-}wtPHP}^{n^2+1}_{n^2}(G)$ 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) + $\mathtt{of\text{-}wtPHP}^{n^2+1}_{n^2}(G)$ derivations is 5.*

Proof. (sketch) We use the characterisation of NS given by Theorem 3.9 and we reason basically as in Theorem 4.4. We know that the problematic clauses in $\mathcal{L}_s$ are weakenings of initial axioms or a single instance of $[\emptyset; z]$. We copy $[\emptyset; z]$ to a $\mathcal{L}_{s+1}$ and we define the formula $x_{\alpha,\beta}$ as in Theorem 5.7. Now the onto axioms for the holes in $\mathcal{L}_s$ become weakenings of inital clauses except for the hole $[\emptyset; z]$, which receive a pigeon flying there from the layer $\mathcal{L}_{s+1}$. ■

It is immediate to generalize Theorem 5.7 and 5.9 from clauses to Θ_d-cedents.

Theorem 5.10. *For every $d \in \mathbb{N}$, tree-like depth-$(d+1)$ Frege + $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ p-simulates weighted/circular depth-d Frege, where G is restricted to bipartite graphs of degree at most 3 and the tree-like depth-$(d+1)$ Frege + $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ derivations are of height 3.*

Recall that the systems circular depth-d Frege and weighted depth-d Frege are p-equivalent by Proposition 3.6.

Theorem 5.11. *For every $d \in \mathbb{N}$, tree-like depth-$(d+1)$ Frege + $\mathtt{of\text{-}wtPHP}^{n^2+1}_{n^2}(G)$ p-simulates weighted depth-d Frege with the SOUNDNESS-NS condition, where G is restricted to bipartite graphs of degree at most 3 and the tree-like depth-$(d+1)$ Frege + $\mathtt{of\text{-}wtPHP}^{n^2+1}_{n^2}(G)$ derivations are of height 3.*

Corollary 5.12. *For every $d \in \mathbb{N}$, tree-like depth-$(d+1)$ Frege + $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ p-simulates depth-d Frege and the tree-like tree-like depth-$(d+1)$ Frege + $\mathtt{wt\,PHP}^{n^2+1}_{n^2}(G)$ derivations are of height 3.*

Proof. It follows from Theorem 5.10, Theorem 2.3 and Theorem 3.7. ■

Regarding size lower bounds we conjecture the following.

Conjecture 5.13. *For every constant d, the formula $\mathtt{MOD}_2$ (see Definition 4.9) does not have polynomial size refutations in depth-d Frege + $\mathtt{wtPHP}$.*

This conjecture, together with the previous results, implies super-polynomial size lower bounds for weighted depth-d Frege and circular depth-d Frege (at least for d constant).

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. In particular the way to refute $\mathtt{wtPHP}^{n^2+1}_{n^2}$ dynamically is the result of conversations with Sam Buss.

References

[Ajt90] M. Ajtai. “Parity and the Pigeonhole Principle”. In: ed. by Samuel R. Buss and Philip J. Scott. Boston, MA: Birkhäuser Boston, 1990, pp. 1–24.

[AL19] Albert Atserias and Massimo Lauria. “Circular (Yet Sound) Proofs”. In: *Theory and Applications of Satisfiability Testing - SAT 2019 - 22nd International Conference, SAT 2019, Lisbon, Portugal, July 9-12, 2019, Proceedings*. Ed. by Mikolás Janota and Inês Lynce. Vol. 11628. Lecture Notes in Computer Science. Springer, 2019, pp. 1–18.

[AO18] Albert Atserias and Joanna Ochremiak. “Proof Complexity Meets Algebra”. In: *ACM Trans. Comput. Logic* 20.1 (Dec. 2018).

[BB05] Arnold Beckmann and Samuel R. Buss. “Separation results for the size of constant-depth propositional proofs”. In: *Annals of Pure and Applied Logic* 136.1 (2005), pp. 30–55.

[BB19] Arnold Beckmann and Sam Buss. “On transformations of constant depth propositional proofs”. In: *Ann. Pure Appl. Log.* 170.10 (2019), pp. 1176–1187.

[BBIMM18] Maria Luisa Bonet, Sam Buss, Alexey Ignatiev, João Marques-Silva, and António Morgado. “MaxSAT Resolution With the Dual Rail Encoding”. In: *AAAI*. 2018, pp. 6565–6572.

[BCEIP98] Paul Beame, Stephen Cook, Jeff Edmonds, Russell Impagliazzo, and Toniann Pitassi. “The Relative Complexity of NP Search Problems”. In: *Journal of Computer and System Sciences* 57.1 (1998), pp. 3–19.

[BIKPP94] P. Beame, R. Impagliazzo, J. Krajicek, T. Pitassi, and 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.

[BL20] Maria Luisa Bonet and Jordi Levy. “Equivalence Between Systems Stronger Than Resolution”. In: *Theory and Applications of Satisfiability Testing – SAT 2020*. Cham: Springer International Publishing, 2020, pp. 166–181.

[BP96] Paul Beame and Toniann Pitassi. “An Exponential Separation between the Parity Principle and the Pigeonhole Principle”. In: 80 (Oct. 1996), pp. 195–228.

[Bus06] Samuel R. Buss. “Polynomial-size Frege and resolution proofs of st-connectivity and Hex tautologies”. In: *Theoretical Computer Science* 357.1 (2006), pp. 35–52.

[Bus98] Samuel R. Buss. “Lower bounds on Nullstellensatz proofs via designs”. In: *Proof complexity and feasible arithmetics (Rutgers, NJ, 1996)*. Vol. 39. DIMACS Ser. Discrete Math. Theoret. Comput. Sci. Amer. Math. Soc., Providence, RI, 1998, pp. 59–71.

[CR79] Stephen A. Cook and Robert A. Reckhow. “The Relative Efficiency of Propositional Proof Systems”. In: *The Journal of Symbolic Logic* 44.1 (1979), pp. 36–50.

[DGM20] Stefan S. Dantchev, Abdul Ghani, and Barnaby Martin. “Sherali-Adams and the Binary Encoding of Combinatorial Principles”. In: *LATIN 2020: Theoretical Informatics - 14th Latin American Symposium, São Paulo, Brazil, January 5-8, 2021, Proceedings*. Ed. by Yoshiharu Kohayakawa and Flávio Keidi Miyazawa. Vol. 12118. Lecture Notes in Computer Science. Springer, 2020, pp. 336–347.

[DMNR21] Susanna F. De Rezende, Or Meir, Jakob Nordström, and Robert Robere. "Nullstellensatz Size-Degree Trade-offs from Reversible Pebbling". In: *computational complexity* 30.1 (2021), p. 4.

[FKP19] Noah Fleming, Pravesh Kothari, and Toniann Pitassi. "Semialgebraic Proofs and Efficient Algorithm Design". In: *Foundations and Trends® in Theoretical Computer Science* 14.1-2 (2019), pp. 1–221.

[FMSV20] Yuval Filmus, Meena Mahajan, Gaurav Sood, and Marc Vinyals. "MaxSAT Resolution and Subcube Sums". In: *Theory and Applications of Satisfiability Testing – SAT 2020*. Ed. by Luca Pulina and Martina Seidl. Cham: Springer International Publishing, 2020, pp. 295–311.

[Hak21] Tuomas Hakoniemi. "Monomial size vs. Bit-complexity in Sums-of-Squares and Polynomial Calculus". In: *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*. IEEE, 2021, pp. 1–7.

[IS06] Russell Impagliazzo and Nathan Segerlind. "Constant-Depth Frege Systems with Counting Axioms Polynomially Simulate Nullstellensatz Refutations". In: *ACM Trans. Comput. Logic* 7.2 (Apr. 2006), pp. 199–218.

[Kra19] Jan Krajíček. *Proof Complexity*. Ed. by Csmbridge University Press. Cambridge University Press, 2019.

[Kra94] Jan Krajiček. "Lower Bounds to the Size of Constant-Depth Propositional Proofs". In: *The Journal of Symbolic Logic* 59.1 (1994), pp. 73–86.

[LR20a] Javier Larrosa and Emma Rollon. "Augmenting the Power of (Partial) MaxSat Resolution with Extension". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 34.02 (Apr. 2020), pp. 1561–1568.

[LR20b] Javier Larrosa and Emma Rollon. "Towards a Better Understanding of (Partial Weighted) MaxSAT Proof Systems". In: *Theory and Applications of Satisfiability Testing – SAT 2020*. Ed. by Luca Pulina and Martina Seidl. Cham: Springer International Publishing, 2020, pp. 218–232.

[PR17] Toniann Pitassi and Robert Robere. "Strongly Exponential Lower Bounds for Monotone Computation". In: *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2017. New York, NY, USA: Association for Computing Machinery, 2017, pp. 1246–1255.

[PR18] Toniann Pitassi and Robert 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. New York, NY, USA: Association for Computing Machinery, 2018, pp. 1207–1219.

[RLNS21] Susanna F. de Rezende, Massimo Lauria, Jakob Nordström, and Dmitry Sokolov. "The Power of Negative Reasoning". In: *36th Computational Complexity Conference (CCC 2021)*. Ed. by Valentine Kabanets. Vol. 200. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021, 40:1–40:24.

[RPRC16] Robert Robere, Toniann Pitassi, Benjamin Rossman, and Stephen 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.

[SA90] Hanif D. Sherali and Warren P. Adams. “A Hierarchy of Relaxations between the Continuous and Convex Hull Representations for Zero-One Programming Problems”. In: *SIAM Journal on Discrete Mathematics* 3.3 (1990), pp. 411–430.

[UF96] Alasdair Urquhart and Xudong Fu. “Simplified Lower Bounds for Propositional Proofs”. In: *Notre Dame Journal of Formal Logic* 37.4 (1996), pp. 523–544.

A Upper bound of $\mathtt{PHP}_n^{n+1}$ in Sherali-Adams

In this section we show how to infer from $\mathtt{PHP}_n^{n+1}$ the inequalities (12) and (13). The crucial lemma is below. This is not needed to understand the main results of this paper, but it might be useful to compare this argument with the proof of Lemma B.1.

Lemma A.1. *Given variables $a_1, \dots, a_m$, from the axioms $-\prod_{i\in[m]} \bar{a}_i \geqslant 0$, there is a polynomial-size* SA *derivation of the inequality*

$$\sum_{i\in[m]} a_i - 1 \geqslant 0\,. \tag{26}$$

From the axioms $\{\, -a_i a_j \geqslant 0 \,:\, i,j \in [m],\ i \neq j \,\}$ there is a polynomial-size SA *derivation of the inequality*

$$1 - \sum_{i\in[m]} a_i \geqslant 0\,. \tag{27}$$

Proof. Let $A_0 = 1$ and, for $k \geqslant 1$, let $A_k = \prod_{\ell\in[k]} \bar{a}_\ell$. First notice we have the algebraic equalities

$$A_i - 1 = \sum_{j\in[i]} (A_{j-1}(a_j + \bar{a}_j - 1) - A_{j-1}a_j)\,, \tag{28}$$

$$\sum_{i\in[m]} a_i - 1 = -A_m + \sum_{i\in[m]} (A_{i-1}(a_i + \bar{a}_i - 1) + (a_i - a_i A_{i-1}))\,. \tag{29}$$

Hence, multiplying (28) by a_{i+1} we get a SA derivation of

$$a_{i+1}A_i - a_{i+1} \geqslant 0$$

from $\{\, -a_i a_j \geqslant 0 \,:\, i,j \in [m]\ i \neq j \,\}$. By multiplying (28) by $-a_{i+1}$ we get a SA derivation of

$$a_{i+1} - a_{i+1}A_i \geqslant 0$$

from $-\prod_{i\in[m]} \bar{a}_i \geqslant 0$. To prove (26) just substitute the SA derivation of $a_i - a_i A_{i-1} \geqslant 0$ in (29). To prove (27) we just fist multiply both sides of (29) by -1, recall that A_m is a monomial and substitute the SA derivation of $a_i A_i - a_i \geqslant 0$ in (29). ■

B Upper bound of $\mathtt{wtPHP}_{n^2}^{n^2+1}$ in Sherali-Adams

In this section we show how to infer from $\mathtt{wtPHP}_{n^2}^{n^2+1}$ the inequalities (14) and (15), i.e. the part that was missing in the proof of Theorem 5.4. The crucial lemma is below.

In this section we use, unlike the rest of the paper, the convention $[n] = \{0, \dots, n-1\}$.

Lemma B.1. *Given variables $a_0, \ldots, a_{n-1}, b_1, \ldots, b_{m-1}$, from the axioms*

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

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

$$2\sum_{i\in[n]} a_i + \sum_{j\in[m]} b_j - 2 \geqslant 0\,. \tag{30}$$

From the axioms

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

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

$$2 - 2\sum_{i\in[n]} a_i - \sum_{j\in[m]} b_j \geqslant 0\,. \tag{31}$$

Proof. (sketch) Let $A_0 = 1$, $B_0 = 1$, $B_{0,1} = 1$ and for every $j,k \geqslant 1$ let $A_k = \prod_{\ell\in[k]} \bar{a}_\ell$, $B_k = \prod_{\ell\in[k]} \bar{b}_\ell$ and $B_{j,k} = \prod_{\ell\in[k]\smallsetminus\{j\}} \bar{b}_\ell$. For sake of shortness let also use the notation $\widetilde{a_i}$ to denote the axiom $a_i + \bar{a}_i - 1$ and $\widetilde{b_i}$ to denote the axiom $b_i + \bar{b}_i - 1$. We have the following equalities

$$A_k - 1 = \sum_{j\in[k]} (A_j \widetilde{a_j} - A_j a_j)\,, \tag{32}$$

$$B_k - 1 = \sum_{\ell\in[k]} (B_\ell \widetilde{b_\ell} - B_\ell b_\ell)\,, \tag{33}$$

$$B_{i,k} - 1 = \sum_{\ell\in[k]\smallsetminus\{i\}} (B_{i,\ell} \widetilde{b_\ell} - B_{i,\ell} b_\ell)\,, \tag{34}$$

$$\sum_{i\in[n]} a_i - 1 = -A_n + \sum_{i\in[n]} (A_i \widetilde{a_i} + (a_i - a_i A_i))\,, \tag{35}$$

$$\sum_{\substack{j\in[m]\\ j\neq i}} b_j - 1 = -B_{i,m} + \sum_{\substack{j\in[m]\\ j\neq i}} (B_{i,j} \widetilde{b_j} + (b_j - B_{i,j} b_j))\,. \tag{36}$$

Now, multiplying (36) by b_i and summing for every $i \in [m]$, we get

$$2\sum_{i\in[m]}\sum_{j\in[i]} b_i b_j - \sum_{i\in[m]} b_i = -\sum_{i\in[m]} b_i B_{i,m} + \sum_{i\in[m]}\sum_{\substack{j\in[m]\\ j\neq i}} B_{i,j} b_i \widetilde{b_j} + (b_i b_j - B_{i,j} b_i b_j) \tag{37}$$

$$= -\sum_{i\in[m]} b_i B_{i,m} + \sum_{i\in[m]}\sum_{\substack{j\in[m]\\ j\neq i}} B_{i,j} b_i \widetilde{b_j} + (b_i b_j - B_{i,j} b_i b_j)\,, \tag{38}$$

and, we can multiply the equality in (34) by $b_i b_j$, and substitute for $b_i b_j - B_{i,j} b_i b_j$ in the equality above. What we get is a polynomial-size SA derivation of the inequality

$$2\sum_{i\in[m]}\sum_{j\in[i]} b_i b_j - \sum_{i\in[m]} b_i \geqslant 0 \tag{39}$$

from S and the axioms $-b_i B_{i,m} \geqslant 0$.

We now multiply (35) by B_m and we add to both sides of the obtained equality $B_m + \sum_{i\in[n]}(a_i - a_i B_m)$ and we get

$$\sum_{i\in[n]} a_i = B_m + \sum_{i\in[n]} (a_i - a_i B_m) - A_n B_m + \sum_{i\in[n]} (A_i B_m \widetilde{a_i} + (a_i B_m - a_i A_i B_m)) . \qquad (40)$$

We need to expand a bit on B_m. From (33), we get

$$B_m = 1 + \sum_{\ell\in[m]} (B_\ell \widetilde{b_\ell} - b_\ell B_\ell) .$$

That is we can substitute back inside this expression the corresponding expression for B_ℓ and again another time:

$$\begin{aligned}
B_m &= 1 + \sum_{\ell\in[m]} (B_\ell \widetilde{b_\ell} - b_\ell B_\ell) \\
&= 1 + \sum_{\ell\in[m]} \Big(B_\ell \widetilde{b_\ell} - b_\ell \Big(1 + \sum_{k\in[\ell]} (B_k \widetilde{b_k} - B_k b_k)\Big)\Big) \\
&= 1 - \sum_{\ell\in[m]} b_\ell + \sum_{\ell\in[m]} \sum_{k\in[\ell]} b_\ell b_k B_k + \sum_{\ell\in[m]} \Big(B_\ell \widetilde{b_\ell} - \sum_{k\in[\ell]} b_\ell B_k \widetilde{b_k}\Big) \\
&= 1 - \sum_{\ell\in[m]} b_\ell + \sum_{\ell\in[m]} \sum_{k\in[\ell]} b_\ell b_k \Big(1 + \sum_{z\in[k]} (B_z \widetilde{b_z} - b_z B_z)\Big) + \sum_{\ell\in[m]} \Big(B_\ell \widetilde{b_\ell} - \sum_{k\in[\ell]} b_\ell B_k \widetilde{b_k}\Big) \\
&= 1 - \sum_{\ell\in[m]} b_\ell + \sum_{\ell\in[m]} \sum_{k\in[\ell]} b_\ell b_k - \sum_{\ell\in[m]} \sum_{k\in[\ell]} \sum_{z\in[k]} b_\ell b_k b_z B_z + \sum_{\ell\in[m]} \Big(B_\ell \widetilde{b_\ell} - \sum_{k\in[\ell]} \Big(b_\ell B_k \widetilde{b_k} - \sum_{z\in[k]} b_\ell b_k B_z \widetilde{b_z}\Big)\Big) .
\end{aligned}$$

That is, substituting this last expression for B_m in (40) and multiplying by 2, we get a polynomial size SA derivation of

$$2 \sum_{i\in[n]} a_i + 2 \sum_{\ell\in[m]} b_\ell - 2 \sum_{\ell\in[m]} \sum_{k\in[\ell]} b_\ell b_k - 2 \geqslant 0 \qquad (41)$$

from S and the axioms $-b_\ell b_k b_z \geqslant 0$. Hence there is a polynomial-size SA derivation of the sum of (39) and (41), that is we proved in SA that

$$2 \sum_{i\in[n]} a_i + \sum_{\ell\in[m]} b_\ell - 2 \geqslant 0 .$$

This equality implies (30). If we multiply (39) and (41) by -1 and we sum we get an analogue expression implying (31). ■

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il