Graph Colouring is Hard for Algorithms Based on Hilbert's Nullstellensatz and Gröbner Bases

Massimo Lauria[1] **and Jakob Nordström**[2]

1 **Sapienza — Università di Roma**
Piazzale Aldo Moro 5, 00185, Rome, Italy
massimo.lauria@uniroma1.it
2 **KTH Royal Institute of Technology**
SE-100 44 Stockholm, Sweden
jakobn@kth.se

Abstract

We consider the graph k-colouring problem encoded as a set of polynomial equations in the standard way. We prove that there are bounded-degree graphs that do not have legal k-colourings but for which the polynomial calculus proof system defined in [Clegg et al. '96, Alekhnovich et al. '02] requires linear degree, and hence exponential size, to establish this fact. This implies a linear degree lower bound for any algorithms based on Gröbner bases solving graph k-colouring using this encoding. The same bound applies also for the algorithm studied in a sequence of papers [De Loera et al. '08, '09, '11, '15] based on Hilbert's Nullstellensatz proofs for a slightly different encoding, thus resolving an open problem mentioned, e.g., in [De Loera et al. '09] and [Li et al. '16]. We obtain our results by combining the polynomial calculus degree lower bound for functional pigeonhole principle (FPHP) formulas over bounded-degree bipartite graphs in [Mikša and Nordström '15] with a reduction from FPHP to k-colouring derivable by polynomial calculus in constant degree.

1998 ACM Subject Classification F.2.2 [Analysis of Algorithms and Problem Complexity]: Nonnumerical Algorithms and Problems — *Computations on discrete structures*; F.1.3 [Computation by Abstract Devices]: Complexity Measures and Classes — *Relations among complexity measures*; G.2.2 [Discrete Mathematics] Graph Theory — *Graph algorithms* F.4.1 [Mathematical Logic and Formal Languages]: Mathematical Logic — *Computational logic*

Keywords and phrases proof complexity, Nullstellensatz, Gröbner basis, polynomial calculus, cutting planes, 3-colouring

Digital Object Identifier 10.4230/LIPIcs.CCC.2017.26

1 Introduction

Given an undirected graph $G = (V, E)$ and a positive integer k, can the vertices $v \in V$ be coloured with at most k colours so that no vertices connected by an edge have the same colour? This *graph colouring problem* is perhaps one of the most extensively studied NP-complete problems. It is widely believed that any algorithm for this problem has to run in exponential time in the worst case, and indeed the currently fastest algorithm for 3-colouring runs in time $O(1.3289^n)$ [8]. A survey on various algorithms and techniques for so-called exact algorithms is [26].

Many graph colouring instances of interest might not exhibit worst-case behaviour, however, and therefore it makes sense to study algorithms without worst-case guarantees and examine how they perform in practice. Dually, it can be of interest to study weak models of computation, which are nevertheless strong enough to capture the power of such algorithms,

© Massimo Lauria and Jakob Nordström;
licensed under Creative Commons License CC-BY

and prove unconditional lower bounds for these models. Obtaining such lower bounds is the goal of this work.

1.1 Brief Background

Since current state-of-the-art algorithms for propositional satisfiability such as *conflict-driven clause learning (CDCL)* [4, 32, 38] are ultimately based on *resolution* [11], it is perhaps not so surprising that this approach can be used to solve colouring problems as well. According to [6], McDiarmid developed a method for deciding k-colourability that captures many concrete algorithms [35]. This method, viewed as a proof system, is simulated by resolution.

There are exponential lower bounds for resolution proofs of non-k-colourability that apply to any such method. In particular, [6] presents average-case exponential lower bounds for random graph k-colouring instances sampled so that the graphs are highly likely not to be k-colourable. This ultimately boils down to proving width lower bounds, i.e., lower bounds on the size of a largest clause in any resolution refutation of the formula, and then using that linear width lower bounds implies exponential size lower bounds [10].

Another possible approach is to attack the k-colouring problem using algebra. Various algebraic methods have been considered in [3, 31, 33, 34]. The thesis [5] contains the first explicit attempt we know of to encode the 3-colouring problem using Hilbert's Nullstellensatz. At a high level, the idea is to write the problem as a set of polynomial equations $\{f_i(x_1, \ldots, x_n) = 0 \mid i \in [m]\}$ over a suitable field $\mathbb{F}$ so that legal colourings correspond to solutions, and if this is done in the right way it holds that this system of equations has no solution if and only if there are polynomials $g_1, \ldots, g_m$ such that $\sum_{i=1}^{m} g_i f_i = 1$. This latter equality is referred to as a *Nullstellensatz certificate* of non-colourability, and the *degree* of this certificate is the largest degree of any polynomial $g_i f_i$ in the sum. Later papers based on Nullstellensatz and Gröbner bases such as [17, 25, 37] have attracted a fair amount of attention. For this work, we are particularly interested in the sequence of papers [19, 21, 20, 18], which uses an encoding of the k-colouring problem that will be discussed more in detail later in the paper.

There seem to be no formally proven lower bounds for these algebraic methods. On the contrary, the authors of [21] report that essentially all of the benchmarks they have studied have Nullstellensatz certificates of constant (and very small) degree. Indeed, no lower bounds for graph colouring is known for the corresponding proof systems *Nullstellensatz* [7] or the stronger system *polynomial calculus* [1, 15]. Intriguingly, in a close parallel to the case for resolution it is known that strong enough lower bounds on polynomial calculus degree imply exponential lower bounds on proof size [27], but the techniques for proving degree lower bounds are much less developed than the width lower bound techniques for resolution.

Even if there are no know degree lower bounds for graph colouring, a sequence of such results exists for other formulas, although in most cases these formula are obviously false and do not express any hard computational problem. In some cases, degree lower bounds can be obtained by making an affine transformation from $\{0, 1\}$-valued variables to $\{-1, +1\}$-valued variables [9, 13], but this only works for polynomial equations with the right structure and only for fields of characteristic distinct from 2. A general and powerful method, which is independent of the field characteristic, was developed in [2], but has turned out to be not so easy to to apply (except in a few papers such as [22, 23]). A slightly different, and in some aspects more general, version of the approach in [2] was recently presented in [36], which also highlighted the similarities and differences between resolution width lower bound techniques and polynomial calculus degree lower bound techniques. This new framework yielded a new degree lower bound which plays a key role in our paper.

1.2 Our Contributions

We exhibit families of non-k-colourable graphs of bounded degree such that the canonical encoding of the corresponding k-colouring instances into systems of polynomial equations over $\{0,1\}$-valued variables require linear degree to be refuted in polynomial calculus.

▶ **Theorem 1.1** (informal). *For any constant $k \geq 3$ there are explicit families of graphs $\{G_n\}_{n\in\mathbb{N}}$ of size $\mathrm{O}(n)$ and constant vertex degree, which are not k-colourable but for which the polynomial calculus proof system requires linear degree, and hence exponential size, to prove this fact, regardless of the underlying field.*

Our degree lower bound also applies to a slightly different encoding with primitive kth roots of unity used in [19, 20] to build k-colouring algorithms based on Hilbert's Nullstellensatz. These algorithms construct certificates of non-k-colourability by solving linear systems of equations over the coefficients of all monomials up to a certain degree.

Just as the algorithms in [19, 20], our lower bound does not work for all fields (the field must have an extension field in which there is a primitive kth root of unity). For simplicity, we state below a concrete result for Nullstellensatz certificates over GF(2) for non-3-colourability, which is one of the main cases considered in [19, 20]. We remark that this answers an open question raised in, for example, [21, 30].

▶ **Corollary 1.2.** *There are explicit families of non-3-colourable graphs such that the algorithms based on Hilbert's Nullstellensatz over* GF(2) *in [19, 20] need to find certificates of linear degree, and hence must solve systems of linear equations of exponential size, in order to certify non-3-colourability.*

Finally, we want to mention that the graph colouring instances that we construct turn out to be easy for the proof system *cutting planes* [16], which formalizes the integer linear programming algorithm in [14, 24] and underlies so-called *pseudo-Boolean* SAT solvers such as, for instance, *Sat4j* [29, 40].

▶ **Proposition 1.3.** *The graph colouring instances for the non-k-colourable graphs in Theorem 1.1 have polynomial-size refutations in the cutting planes proof system.*

1.3 Techniques

Perhaps somewhat surprisingly, no heavy-duty machinery is required to establish Theorem 1.1. Instead, all that is needed is a nifty reduction. Our starting point is the so-called *functional pigeonhole principle (FPHP) formula* restricted to a bipartite graph of bounded left degree k. This formula expresses the claim that a set of pigeons $i \in I$ can be mapped to a set of pigeonholes $j \in J$ in a one-to-one fashion, where in addition the pigeons are constrained so that every pigeon can choose not between all available holes but only between a set of k holes as specified by the bipartite graph. Clearly, FPHP formulas are unsatisfiable when $|I| > |J|$.

Any instance of a graph FPHP formula can be viewed as a constraint satisfaction problem by ordering the available holes for every pigeon in some arbitrary but fixed way, and then keeping track of where each pigeon is mapped by recording the ordinal number of its chosen pigeonhole. If the cth hole for pigeon i and the c'th hole for pigeon i' is one and the same hole j, then pigeons i and i' cannot be allowed to make choices c and c' simultaneously. If we view this constraint as an edge in graph with the pigeons I as vertices, this is already close to a graph colouring instance, except that what is forbidden for the neighbours i and i' is not the same colour c but some arbitrary pair of possibly distinct colours (c, c'). However,

the idea outlined above can be turned into a proper reduction from graph FPHP formulas to k-colouring instances by using appropriately constructed gadgets of constant size.

We then combine this reduction with the recent polynomial calculus degree lower bound in [36], which works as long as the underlying bipartite graph is a *boundary expander* (a.k.a. *unique-neighbour expander*). More precisely, we show that the reduction from FPHP to graph k-colouring sketched above can be computed in polynomial calculus in low degree. Therefore, any low-degree polynomial calculus refutations of the graph k-colouring instances could be used to obtain low-degree refutations of FPHP instances, but [36] tells us that FPHP instances over expander graphs require linear degree.

In order to obtain Corollary 1.2, we assume that we have a low-degree Nullstellensatz certificate (or, more generally, a polynomial calculus proof) of non-colourability for the roots-of-unity encoding in [19, 20]. Then it is not hard to show that if the field we are working in contains a primitive kth root of unity, we can apply a linear variable substitution to obtain a polynomial calculus refutation in essentially the same degree of the colouring instance in the encoding with $\{0,1\}$-valued variables. The corollary now follows from Theorem 1.1.

As should be clear from the discussion above, the hardness of our graph colouring instances ultimately derives from the pigeonhole principle. However, this combinatorial principle is well-known to be easy for cutting planes. We establish Proposition 1.3 by showing that cutting planes can unpack the reduction described above to recover the original pigeonhole principle instance, after which this instance can be efficiently refuted.

1.4 Outline of This Paper

The rest of this paper is organized as follows. We start by presenting some proof complexity preliminaries and discussing how to encode the graph colouring problem in Section 2. In Section 3 we describe our graph k-colouring instances and prove that they are hard for polynomial calculus, and in Section 4 we show that the same instances are easy for cutting planes. We conclude in Section 5 by discussing some directions for future research. We refer to the upcoming full-length version for all missing proofs.

2 Preliminaries

Throughout this paper $x_1, \ldots, x_n$ denote $\{0,1\}$-valued variables, where we think of 1 as true and 0 as false. We write $\mathbb{N} = \{0, 1, 2, \ldots\}$ for the natural numbers and denote $\mathbb{N}^+ = \mathbb{N} \setminus \{0\}$. For $n \in \mathbb{N}^+$ we use the standard notation $[n] = \{1, 2, \ldots, n\}$. For a set E, we use the shorthand $e \neq e' \in E$ to index over pairs of distinct elements $e, e' \in E$, $e \neq e'$.

2.1 Proof Complexity

Polynomial calculus (PC) [15] is a proof system based on algebraic reasoning where one expresses constraints over Boolean variables as polynomial equations and applies algebraic manipulations to deduce new equations. The constraints are over $\{0,1\}$-valued variables $x_1, \ldots, x_n$, and each constraint is encoded as a polynomial Q in the ring $\mathbb{F}[x_1, \ldots, x_n]$, where $\mathbb{F}$ is some fixed field. The intended meaning is that $Q = 0$ if and only if the constraint is satisfied, but we omit "$= 0$" below and only write the polynomial Q. A *PC derivation* of a polynomial R from a set of polynomials $\mathcal{S} = \{Q_1, \ldots, Q_m\}$ is a sequence $(P_1, \ldots, P_\tau)$ such that $P_\tau = R$ and for $1 \leq t \leq \tau$ the polynomial P_t is obtained by one of the following derivation rules:

- **Boolean axiom:** P_t is $x^2 - x$ for some variable x;

- **Initial axiom:** P_t is one of the polynomials $Q_j \in \mathcal{S}$;
- **Linear combination:** $P_t = \alpha P_i + \beta P_j$ for $1 \leq i, j < t$ and some $\alpha, \beta \in \mathbb{F}$;
- **Multiplication:** $P_t = xP_i$ for $1 \leq i < t$ and some variable x.

A *PC refutation* of $\mathcal{S}$ is a derivation of the multiplicative identity 1 of $\mathbb{F}$ from $\mathcal{S}$. Note that the Boolean axioms make sure that variables can only take values 0 and 1. For this reason, we can assume without loss of generality that all polynomials appearing in PC derivations are multilinear.

The *size* of a polynomial P is the number of distinct monomials in it when it is expanded out as a linear combination of monomials,[1] and the *degree* of P is the largest (total) degree of any monomial in P. The size of a PC derivation π is the sum of the sizes of all polynomials in π, and the degree is the maximal degree of any polynomial in π. One can also define the *length* of a PC derivation as the number of derivation steps in it, but this not so interesting a measure since it may fail to take account of polynomials of exponential size.[2] A fundamental fact about PC is that the size and degree measures are tightly related as stated next.

▶ **Theorem 2.1** ([27]). *For any set $\mathcal{S}$ of inconsistent polynomials of degree at most d' over n variables it holds that if the minimum degree of any PC refutation for $\mathcal{S}$ is at least d, then any PC refutation of $\mathcal{S}$ has size* $\exp\big(\Omega\big((d-d')^2/n\big)\big)$.

In particular, if the polynomials in $\mathcal{S}$ have constant degree but require refutations of degree linear in the number of variables n, then any refutation must have exponential size.

We remark that there is also a slightly more general version of this proof system known as *polynomial calculus (with) resolution (PCR)* [1]. The difference is that PCR has separate formal variables x and $\overline{x}$ to represent both positive and negative literals when translating CNF formulas into sets of polynomials, as well as *complementarity axioms* $x + \overline{x} - 1$ to ensure that x and $\overline{x}$ take opposite values. This yields a nicer and more well-behaved proof system. The change from PC to PCR does not affect the degree needed to refute an inconsistent set of polynomial equations, however, and Theorem 2.1 holds also for PCR. Therefore, the lower bounds we show in this paper apply both to PC and PCR.

Another aspect worth noticing is that it makes perfect sense to define polynomial calculus also for sets of polynomial equations that do not include Boolean axioms $x^2 - x$. One variant studied in the literature is to include axioms $x^k - 1$ instead, i.e., insisting that the value of x is a kth root of unity. In such a setting it is no longer necessarily true that large degree implies large space, however.

In this paper we will also consider *cutting planes (CP)* [16], which is a proof system based on manipulation of inequalities $\sum_i a_i x_i \geq \gamma$ where a_i and γ are integers and $x_1, \ldots, x_n$ are $\{0,1\}$-valued variables. A *CP derivation* of an inequality B from a set of inequalities $\mathcal{S} = \{A_1, \ldots, A_m\}$ is a sequence $(B_1, \ldots, B_\tau)$ such that $B_\tau = B$ and for $1 \leq t \leq \tau$ the inequality B_t is obtained by one of the following derivation rules:

- **Variable axiom:** B_t is either $x \geq 0$ or $-x \geq -1$ for some variable x.
- **Initial axiom:** B_t is some $A_j \in \mathcal{S}$;
- **Sum:** $B_t = B_i + B_j$ for $1 \leq i, j < t$.
- **Scalar multiplication:** $B_t = cB_i$ for $1 \leq i < t$ and $c \in \mathbb{N}$;

[1] Just to make terminology precise, in this paper a *monomial* is a product of variables, a *term* is a monomial multiplied by a non-zero coefficient from the field $\mathbb{F}$, and a *polynomial* is always considered as a linear combinations of terms over pairwise distinct monomials.

[2] Indeed, if multiplication is defined to multilinearize polynomials automatically, as in, e.g., [2], then any unsatisfiable CNF formula encoded into polynomials in the natural way can be refuted in linear length—see [36] for details.

- **Division:** The inequality B_t is

$$\sum_i \frac{a_i}{c} x_i \geq \left\lceil \frac{\gamma}{c} \right\rceil \tag{1}$$

where c divides all $a_1, \ldots, a_n$ and $\sum_i a_i x_i \geq \gamma$ is some inequality B_i for $1 \leq i < t$.

A CP refutation of $\mathcal{S} = \{A_1, \ldots, A_m\}$ is a derivation from $\mathcal{S}$ of the inequality $0 \geq 1$. In what follows, we will often write $\sum_i a_i x_i \leq \gamma$ as an alias for $\sum_i -a_i x_i \geq -\gamma$, and we will also use $\sum_i a_i x_i = \gamma$ as a shorthand for the two inequalities $\sum_i a_i x_i \leq \gamma$ and $\sum_i a_i x_i \geq \gamma$.

The *length* of a CP derivation is the number of derivation steps in it. The *size* of a linear inequality $\sum_i a_i x_i \geq \gamma$ is the number of variables plus the bit size of representations of the constant term γ and all coefficients a_i, and the size of a CP derivation π is the sum of the sizes of all inequalities in π. We do not know of any degree-like measure for CP that would yield relation as that between size and degree for PC in Theorem 2.1. One usually does not distinguish too carefully between length and size for CP since by [12] all coefficients in a CP refutation can be assumed to have at most exponential size, and are hence representable with a linear number of bits.

For a partial mapping $\rho : D \to R$ from a domain D to a range R we let $\mathrm{dom}(\rho)$ denote the set of element with an image. For $d \in D \setminus \mathrm{dom}(\rho)$ we write $\rho(d) = *$. Given a partial assignment or *restriction* ρ of variables $x_1, \ldots, x_n$ to values in $\{0, 1\}$ and a polynomial P or a linear inequality A, we denote by $P\restriction_\rho$ and $A\restriction_\rho$ the polynomial and linear inequality obtained from P and A by restricting the variables in the domain of ρ to the corresponding values and making obvious syntactic simplifications. Given a derivation π in PC or CP, we denote by $\pi\restriction_\rho$ the sequence of restricted polynomials or linear inequalities, respectively. It is straightforward to verify that if π is a CP derivation of an inequality A from $\mathcal{S}$, then $\pi\restriction_\rho$ can be viewed (after simple syntactic manipulations) as a derivation of $A\restriction_\rho$ from $\mathcal{S}\restriction_\rho$ of at most the same length, and the same holds for PC with respect to size and degree.

2.2 The Graph Colouring Problem

A *legal k-colouring* of an undirected graph $G = (V, E)$ with vertices $V(G) = V$ and edges $E(G) = E$ is a mapping $\chi : V \to [k]$ such that for every edge $(u, v) \in E$ it holds that $\chi(u) \neq \chi(v)$. The *chromatic number* $\chi(G)$ of G is the smallest k such that a legal k-colouring of G exists. In the rest of this paper, colourings will often be assumed to be legal unless specified otherwise, so we will sometimes omit this prefix when no misunderstanding can occur. Also, it will sometimes be convenient to number the k colours $0, 1, \ldots, k-1$ instead of $1, 2, \ldots, k$, and we will be fairly relaxed about this issue, implicitly identifying colours 0 and k whenever convenient.

Given a graph G we can encode the k-colourability problem in a natural way as a system of polynomial equations over Boolean variables

$$\sum_{j=1}^{k} x_{v,j} = 1 \qquad v \in V(G), \tag{2a}$$

$$x_{v,j} x_{v,j'} = 0 \qquad v \in V(G),\ j \neq j' \in [k], \tag{2b}$$

$$x_{u,j} x_{v,j} = 0 \qquad (u, v) \in E(G),\ j \in [k], \tag{2c}$$

with the intended meaning that $x_{v,j} = 1$ if vertex v has colour $\chi(v) = j$. It is clear that this system of equations has a solution if and only if the graph G is k-colourable.

We will also be interested in an alternative algebraic representation of the k-colouring problem appearing, e.g., in [19, 20, 21]. In this encoding every vertex $v \in V$ has a single

associated variable y_v which takes values in $\{1, \omega, \omega^2, \ldots, \omega^{k-1}\}$, where ω is a primitive kth root of unity. The intended meaning is that $y_v = \omega^j$ if vertex v has colour $j \in \{0, 1, \ldots, k-1\}$. The colouring constraints are enforced by the polynomial equations

$$y_v^k = 1 \qquad v \in V(G), \tag{3a}$$

$$\sum_{j=0}^{k-1} (y_u)^j (y_v)^{k-1-j} = 0 \qquad (u,v) \in E(G), \tag{3b}$$

where the polynomials live in a polynomial ring over a field of characteristic that is not a positive number dividing k. Clearly, Equation (3a) forces the vertex v to take some colour. A moment of thought reveals that Equation (3b) correctly encodes an edge constraint: if $y_u = \omega^a$ and $y_v = \omega^b$, then the sum evaluates to $\omega^{b(k-1)} \sum_{j=0}^{k-1} \omega^{j(a-b)}$, which equals 0 when $a \neq b$ and $k\omega^{b(k-1)} \neq 0$ otherwise. The latter formulation of k-colouring only makes sense if the characteristic of the underlying field $\mathbb{F}$ is either 0 or a positive integer that does not divide k. In this case, we also know that there exists an extension field $\mathbb{E}$ of $\mathbb{F}$ that contains a primitive kth root of unity ω [28, Chapter VI.3].

A simple but important observation for us is that the choice of the polynomial encoding is not too important if we want to study how large degree is needed in polynomial calculus when proving that some graph G is not k-colourable, provided that the field we are in contains, or can be extended to contain, a primitive kth root of unity.

▶ **Proposition 2.2.** *Suppose that Equations* (3a)–(3b) *have a polynomial calculus refutation of degree d over some field $\mathbb{F}$ of characteristic that is not a positive number dividing k. Then $\mathbb{F}$ can be extended to a field $\mathbb{E}$ containing a primitive kth root of unity ω, and it holds that Equations* (2a)–(2c) *have a polynomial calculus refutation over $\mathbb{E}$ of degree* $\max\{k, d\}$.

Proof sketch. Given any polynomial calculus refutation π of Equations (3a)–(3b), we apply the linear substitutions

$$y_v \mapsto \sum_{j=1}^{k} x_{v,j} \omega^j \tag{4}$$

to all variables in all polynomials in this refutation to obtain a new sequence of polynomials π' in variables $x_{v,j}$. All applications of the linear combination rule in π remain valid in π', and all applications of multiplication in π can be carried out in π' by a combination of multiplication and linear combination steps. The final line of the refutation, i.e., the multiplicative identity 1, is the same in π and π'. What remains to argue is that the substituted versions of the initial axioms (3a)–(3b) in π can be derived from the axioms (2a)–(2c) available to π'. We refer to the upcoming full-length version for the details. ◀

For later use, we note that we can also encode the k-colourability problem for a graph G as a system of linear inequalities

$$\sum_{j=1}^{k} x_{v,j} \geq 1 \qquad v \in V(G), \tag{5a}$$

$$x_{v,j} + x_{v,j'} \leq 1 \qquad v \in V(G),\ j \neq j' \in [k], \tag{5b}$$

$$x_{u,j} + x_{v,j} \leq 1 \qquad (u,v) \in E(G),\ j \in [k], \tag{5c}$$

in a format amenable to cutting planes reasoning.

3 Worst-Case Lower Bound for Polynomial Calculus

We now show how to explicitly construct a family of graphs which are not k-colourable but for which polynomial calculus proofs of this fact (over any field) require degree linear in the number of vertices in the graphs. We do this in three steps:

1. First, we show how to reduce instances of *functional pigeonhole principle (FPHP) formulas* defined over bipartite graphs of bounded degree to graph colouring instances so that there is a one-to-one mapping of pigeons to holes if and only if the graph is k-colourable.
2. Then we show that polynomial calculus is able to carry out this reduction in constant degree, so that a low-degree PC proof of graph non-colourability can be used to obtain a low-degree refutation of the corresponding FPHP instance.
3. Finally, we appeal to a linear lower bound on degree for refuting FPHP instances over bipartite expander graphs from [36].

Let us start by giving a precise description of our functional pigeonhole principle instances. We have a set of pigeons I which want to fly into a set of holes J, with each pigeon flying into exactly one hole in a one-to-one fashion. However, the choices of holes for the pigeons are constrained, so that pigeon i can fly only to the holes in $J(i) \subseteq J$, where we have $|J(i)| = k$. If we use variables $p_{i,j}$ to denote that pigeon i flies into hole j, we can write the constraints on such a mapping as a set of polynomial equations

$$\sum_{j \in J(i)} p_{i,j} = 1 \qquad i \in I, \tag{6a}$$

$$p_{i,j}p_{i,j'} = 0 \qquad i \in I,\ j \neq j' \in J(i). \tag{6b}$$

$$p_{i,j}p_{i',j} = 0 \qquad i \neq i' \in I,\ j \in J(i) \cap J(i'). \tag{6c}$$

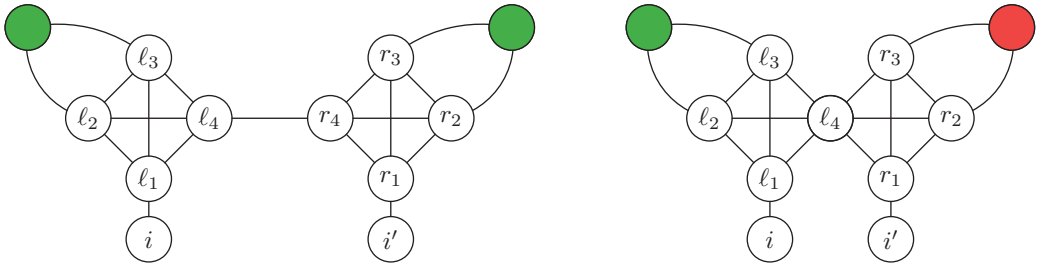
Note that an instance encoded by Equations (6a)–(6c) can also be naturally viewed as a bipartite graph B with left vertex set I, right vertex set J, and edges from each $i \in I$ to all $j \in J(i)$. In what follows, we will mostly reason about FPHP instances in terms of their representations as bipartite graphs.

In the standard setting, we let $I = [n]$ and $J = [n-1]$ for some $n \in \mathbb{N}$, in which case the collection of constraints (6a)–(6c) is clearly unsatisfiable. Nevertheless, it was shown in [36] that if the underlying bipartite graph is a so-called *boundary expander*, then any PC refutation of Equations (6a)–(6c) requires $\Omega(n)$ degree and thus, by Theorem 2.1, exponential size. For our results, we do not need to go into the technical details of this lower bound, but it suffices to use the following claim as a black box.

▶ **Theorem 3.1** ([36]). *For any integer $k \geq 3$ there is an efficiently constructible family of bipartite graphs $\{B_n\}_{n \in \mathbb{N}}$ with n vertices on the left side, $n-1$ vertices on the right side, left degree k, and right degree $O(k)$, such that any polynomial calculus refutation of the corresponding constraints* (6a)–(6c) *requires degree $\Omega(n)$.*

To be precise, the lower bound in Theorem 3.1 was proven for a slightly different encoding of Equations (6a)–(6c)—namely the one obtained from the natural translation of CNF formulas into polynomial equations—but the two encodings imply each other and can be used to derive each other in degree $O(k)$ by the implicational completeness of polynomial calculus. Hence, the lower bound holds for both encodings.

We proceed to describe the reduction from functional pigeonhole principle instances to graph colouring instances. Our starting point is an FPHP instance on a bipartite graph B with pigeons $I = [n]$ and holes J where every pigeon has exactly $d_I = k$ holes to choose from



(a) Forbidding $i \leftarrow c$ and $i' \leftarrow c$. **(b)** Forbidding $i \leftarrow c$ and $i' \leftarrow c'$ for $c \neq c'$.

Figure 1 Injectivity constraint gadgets $G_{(i,i') \not\leftarrow (c,c')}$ for $k = 4$.

and every hole can take $O(k)$ pigeons; i.e., the bipartite graph B is left-regular of degree k and has right degree $O(k)$. Based on this instance we construct a graph $G = G(B)$ such that G is k-colourable if and only if the functional pigeonhole principle on B is satisfiable.

By way of overview, the graph $G(B)$ has n special vertices corresponding to the pigeons, and the colours of these vertices encode how the pigeons are mapped to holes. For every pair of pigeons i, i' that can be mapped to the same hole j we add a gadget that forbids the colouring of the pigeon vertices i and i' that corresponds to them being mapped to hole j. These gadgets have a couple of pre-coloured vertices, but we eliminate such pre-colouring by adding one more simple gadget.

In more detail, the main idea behind the reduction is to view the choices $J(i)$ for each pigeon $i \in I$ as taking the first, second, ..., kth edge. We fix an arbitrary enumeration of the elements of $J(i)$ for each $i \in I$, associating distinct numbers $1, 2, \ldots, k$ to the edges out of the vertex i in B. We say that *pigeon i flies to hole j using its cth edge* if the edge connecting pigeon i to hole j is labelled by $c \in [k]$, and use the notation $i \leftarrow c$ for this (suppressing the information about the hole j). Pigeon i taking the cth edge corresponds to the special ith pigeon vertex being coloured with colour c.

Consider two distinct pigeons $i \neq i' \in I$ and a hole $j \in J(i) \cap J(i')$. If pigeon i flies to hole j using its cth edge and pigeon i' flies to hole j using its c'th edge, then the translation of the injectivity constraint (6c) expressed in terms of k-colourings is that vertices i and i' cannot be simultaneously coloured by colours c and c', respectively.

Let us now give a precise description of the graph gadgets we employ to enforce such injectivity constraints. These will be partially pre-coloured graphs $G_{(i,i') \not\leftarrow (c,c')}$ as depicted in Figures 1a and 1b. The gadget constructions start with two disjoint k-cliques for pigeons i and i', which we will refer to as the left and right cliques, respectively. We refer to the vertices in the left clique as $\ell_1, \ldots, \ell_k$ numbered in a clockwise fashion starting with the first vertex at the bottom, and in a symmetric fashion the vertices in the right clique are referred to as $r_1, \ldots, r_k$ numbered anti-clockwise starting at the bottom.

To the first vertex ℓ_1 in the left k-clique we connect the vertex i. To vertices $\ell_2, \ldots, \ell_{k-1}$ we connect a new vertex pre-coloured with colour c. For the right k-clique we do a similar construction: to the first vertex r_1 we connect the vertex i' and to the next $k-2$ vertices $r_2, \ldots, r_{k-1}$ we connect a new vertex pre-coloured with colour c'.

The final step of the construction depends on whether $c = c'$ or not. If $c = c'$, then we add an edge between the final two vertices ℓ_k and r_k in the cliques. If $c \neq c'$, then we instead merge these two vertices into a single vertex as shown in Figure 1b. We want to stress that except for i and i' all vertices in the construction are new vertices that do not occur in any other gadget. Let us collect for the record some properties of this gadget construction.

▶ **Claim 3.2.** *The pre-coloured graph gadget $G_{(i,i')\not\leftarrow(c,c')}$ has the following properties:*

1. *$G_{(i,i')\not\leftarrow(c,c')}$ has $\mathrm{O}(k)$ vertices.*
2. *$G_{(i,i')\not\leftarrow(c,c')}$ has two pre-coloured vertices of degree $\mathrm{O}(k)$.*
3. *For every $(b,b') \neq (c,c')$ there is a legal k-colouring χ of $G_{(i,i')\not\leftarrow(c,c')}$ extending the pre-colouring and satisfying $\chi(i) = b$ and $\chi(i') = b'$. No such legal k-colouring of $G_{(i,i')\not\leftarrow(c,c')}$ exists for $(b,b') = (c,c')$.*

Proof. The first two properties obviously hold by construction.

To prove Property 3, let us focus on the left clique in either of the two variant of the gadget. If $\chi(i) = c$, then clearly vertex ℓ_1 in the left clique cannot take colour c. Since the pre-coloured vertex connected to vertices $\ell_2, \ldots, \ell_{k-1}$ of the clique also has colour c, and since any legal colouring must use all available colours for the clique, this forces $\chi(\ell_k) = c$. If $\chi(i) \neq c$, however, then we can colour vertex ℓ_1 with colour c, and then choose any permutation of the remaining colours for the other vertices in the left clique, giving the vertex ℓ_k at least two distinct colours to choose between.

Consider now the case $c = c'$, so that we have the graph gadget $G_{(i,i')\not\leftarrow(c,c)}$ in Figure 1a. By symmetry, if $\chi(i') = c'$, then this forces $\chi(r_k) = c$, but there are at least two choices for the colour of r_k if $\chi(i') \neq c'$. It follows that if $i \leftarrow c$ and $i' \leftarrow c$, then vertices ℓ_k and r_k both have to get the same colour c to avoid conflicts in the left and right k-cliques, respectively, which causes a conflict along the edge (ℓ_k, r_k). As long as one of i and i' is assigned a colour other than c, however, $G_{(i,i')\not\leftarrow(c,c)}$ can be legally k-coloured. For $c \neq c'$ we reason analogously but use instead the graph gadget $G_{(i,i')\not\leftarrow(c,c')}$ in Figure 1b. ◀

We write $\widehat{G} = \widehat{G}(B)$ to denote the graph consisting of the union of all gadgets $G_{(i,i')\not\leftarrow(c,c')}$ for all $i \neq i' \in I$ and all c, c' such that if pigeon i uses its cth edge and pigeon i' uses its c'th edge in B, then they both end up in the same hole $j \in J$. All vertices corresponding to pigeons $i \in I$ are shared between gadgets $G_{(i,i')\not\leftarrow(c,c')}$ in $\widehat{G}$, but apart from this all subgraphs $G_{(i,i')\not\leftarrow(c,c')}$ are vertex-disjoint. We next state some properties of $\widehat{G}$.

▶ **Lemma 3.3.** *Consider an FPHP instance encoded by Equations* (6a)–(6c) *for a left-regular bipartite graph with left degree $d_I = k$ and bounded right degree $d_J = \mathrm{O}(k)$, and let $\widehat{G}$ be the partially k-coloured graph obtained as described above. Then $\widehat{G}$ has $\mathrm{O}\big(k^4|I|\big)$ vertices and maximal vertex degree $\mathrm{O}(k^2)$, and the number of pre-coloured vertices is $\mathrm{O}\big(k^2|I|\big)$. Furthermore, the partial k-colouring of $\widehat{G}$ can be extended to a complete, legal k-colouring of $\widehat{G}$ if and only if there is a way to map each pigeon $i \in I$ to some hole $j \in J$ without violating any constraint in* (6a)–(6c).

Proof. Without loss of generality we can assume that $|J| \leq k|I|$ (otherwise there are holes that cannot be used by any pigeon). Each gadget $G_{(i,i')\not\leftarrow(c,c')}$ has $\mathrm{O}(k)$ vertices and there are at most $(d_J)^2 = \mathrm{O}\big(k^2\big)$ distinct pairs of pigeons that can fly to any single hole j, meaning that we have a total of at most $\mathrm{O}\big(k^2|J|\big)$ injectivity constraint gadgets $G_{(i,i')\not\leftarrow(c,c')}$. Therefore, by a crude estimate $\widehat{G}$ has at most $\mathrm{O}\big(k^4|I|\big)$ vertices in total.

By Claim 3.2 at most $\mathrm{O}\big(k^2|I|\big)$ vertices in $\widehat{G}$ are pre-coloured. Each pigeon vertex labelled by $i \in I$ is involved in at most $d_I d_J = \mathrm{O}\big(k^2\big)$ injectivity constraint gadgets, so such vertices have degree $\mathrm{O}\big(k^2\big)$, while all other vertices have degree $\mathrm{O}(k)$.

For any complete colouring of $\widehat{G}$ extending the pre-colouring, the colours $\chi(i) = c_i$ assigned to pigeon vertices $i \in I$ define a mapping from pigeons to holes via the chosen edges c_i. It follows from Claim 3.2 that this colouring is legal only if pigeons are mapped to holes in a one-to-one fashion, which implies that Equations (6a)–(6c) are satisfiable. In the other direction, for any one-to-one mapping of pigeons to holes we can colour vertex i by the

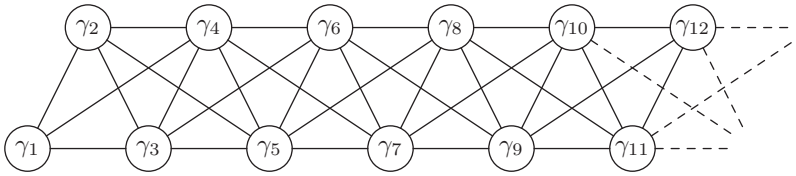


Figure 2 Pre-colouring gadget with vertices to be identified with the pre-coloured vertices in $\widehat{G}$.

colour c_i corresponding to the edge it uses to fly to its hole, and such a colouring can be combined with the pre-colouring complete, to produce a legal k-colouring. ◀

To finalize our reduction we need to get rid of the pre-coloured vertices in $\widehat{G}$. To this end, we first make the following observation. Recall that for every every pigeon $i \in I$ we fixed an enumeration of the edges to holes $j \in J(i)$ in B, so that the choice of an edge corresponds to the choice of a colour. Suppose we apply some arbitrary but fixed permutation σ on $[k]$ to all such enumerations for the pigeons $i \in I$. Clearly, this does not change the instance in any significant way. If it was the case before that pigeon i and i' could not simultaneously take the cth and c'th edges, respectively, then now these pigeons cannot simultaneously take the $\sigma(c)$th and $\sigma(c')$th edges, respectively. In other words, Lemma 3.3 is invariant with respect to any permutation of the colours $[k]$, and we could imagine the reduction as first picking some permutation σ and then constructing $\widehat{G}$ with respect to this permutation.

A simple way of achieving this effect would be to construct a separate "pre-colouring k-clique" consisting of k special vertices $\gamma_1, \ldots, \gamma_k$, and then identify all vertices in $\widehat{G}$ pre-coloured with colour c with the vertex γ_c. It is not hard to see that the resulting graph would be k-colourable if and only if the pre-colouring of $\widehat{G}$ could be extended to a complete, legal k-colouring, and using Lemma 3.3 we would obtain a valid reduction from the functional pigeonhole principle to graph k-colouring. However, the final graph would have degree $\Omega\big(k^3|I|\big)$, and we would like to obtain a graph of bounded degree.

To keep the vertex degrees independent of the size $|I|$ of the left-hand side of the FPHP bipartite graph B, we instead construct a pre-colouring gadget using a slight modification of the above idea. Consider a set $\{\gamma_1, \gamma_2, \ldots, \gamma_M\}$ of new vertices, for M to be fixed later. For every segment of k consecutive vertices $\{\gamma_t, \gamma_{t+1}, \ldots, \gamma_{t+k-1}\}$ we add all edges $\big\{(\gamma_c, \gamma_{c'}) \,\big|\, c \neq c' \in \{t, t+1, \ldots, t+k-1\}\big\}$ so that they form a k-clique as illustrated in Figure 2 (where as in Figure 1 we have $k = 4$). Next, we go through all the pre-coloured vertices in $\widehat{G}$: if a vertex should be pre-coloured by c, then we identify it with the first vertex γ_t such that $t \equiv c \pmod{k}$ and such that γ_t has not already been used at a previous step. If we choose $M = \mathrm{O}\big(k^3|I|\big)$, then we are guaranteed to have enough vertices γ_t to be able to process all pre-coloured vertices in this way.

Our final graph $G = G(B)$ is the previous graph $\widehat{G}$ with pre-coloured vertices identified with (uncoloured) vertices in the additional pre-colouring gadget as just described. Clearly, G is k-colourable if and only if the pre-colouring of $\widehat{G}$ can be completed to a legal k-colouring. We summarize the properties of our reduction in the following proposition, stated here without proof.

▶ **Proposition 3.4.** *Given a graph FPHP formula over a left-regular bipartite graph B with left degree $d_I = k$ and bounded right degree $d_J = \mathrm{O}(k)$, there is an explicit construction of a graph $G = G(B)$ such that G has $\mathrm{O}\big(k^4|I|\big)$ vertices of maximal vertex degree $\mathrm{O}(k^2)$ and is k-colourable if and only if Equations* (6a)–(6c) *are simultaneously satisfiable.*

Since our reduction encodes local injectivity constraints into local colouring constraints, it stands to reason that we should be able to translate between these two types of constraints using low degree derivations. In particular, it seems reasonable to expect that any low-degree refutation of the k-colouring problem for $G(B)$ should yield a low-degree refutation for the functional pigeohole principle on B. This is indeed the case, as stated in the next lemma.

▶ **Lemma 3.5.** *Consider the graph $G = G(B)$ obtained from a bipartite graph B as in Proposition 3.4. If the k-colourability constraints* (2a)–(2c) *for G have a PC refutation in degree d, then the functional pigeonhole principle constraints* (6a)–(6c) *defined over B have a PC refutation of degree at most $2d$.*

We will spend what remains of this section on proving this lemma. The proof is quite similar in spirit to that of Proposition 2.2. We start by assuming that we have a PC refutation of Equations (2a)–(2c) in degree d. Our first step is to substitute all variables $x_{v,j}$ in this refutation with polynomials of degree at most 2 in variables $p_{i,j}$. In the second step, we argue that if we apply this substitution to the axioms in (2a)–(2c), then we can derive the resulting substituted polynomials from Equations (6a)–(6c) by PC derivations in low degree. Taken together, this yields a PC refutation in low degree of the FPHP instance (6a)–(6c).

To describe the substitution, let us focus on a single gadget $G_{(i,i')\not\leftarrow(c,c')}$. The first step is to express all equations for this gadget as equations over variables $x_{i,1}, \dots x_{i,k}, x_{i',1}, \dots x_{i',k}$. Note that these variables are essentially the same as those from the pigeonhole principle instance, except that instead of $p_{i,j}$ we use the variable $x_{i,c}$ where c is the number of the edge pigeon i uses to fly to hole j, but for the sake of exposition we want to keep using the language of colourings.

Let w and w' be the vertices that are supposed to be pre-coloured with colours c and c', respectively. We stress that now we are considering the graph G which has no pre-coloured vertices, and in particular all the variables mentioning the vertices w and w' are unassigned. Recall that w and w' also appear in the gadget depicted in Figure 2, where they are identified with some vertices γ_t and $\gamma_{t'}$ such that $t \equiv c$ and $t' \equiv c' \pmod k$.

For any pair (b, b') of colours different from (c, c'), Claim 3.2 guarantees that we can pick some colouring $\chi_{(b,b')}$ for the gadget $G_{(i,i')\not\leftarrow(c,c')}$ such that $\chi_{(b,b')}(i) = b$, $\chi_{(b,b')}(i') = b'$, $\chi_{(b,b')}(w) = c$ and $\chi_{(b,b')}(w') = c'$. Fix for the rest of this proof such a colouring $\chi_{(b,b')}$ for the gadget $G_{(i,i')\not\leftarrow(c,c')}$ for every $(b, b') \neq (c, c')$. Then we can write the colour of any vertex v in $G_{(i,i')\not\leftarrow(c,c')}$ other than the pigeon vertices i and i' as a function of (b, b'). In more detail, we can express every variable $x_{v,j}$, for $v \notin \{i, i'\}$, as a degree-2 polynomial over the variables $x_{i,1}, \dots x_{i,k}, x_{i',1}, \dots x_{i',k}$ by summing over the monomials $x_{i,b}x_{i',b'}$ corresponding to the choices of colours (b, b') for (i, i') for which the colouring $\chi_{(b,b')}$ assigns colour j to vertex v, or in symbols

$$x_{v,j} \mapsto \sum_{(b,b')\neq(c,c'),\, \chi_{(b,b')}(v)=j} x_{i,b}x_{i',b'} \ . \tag{7}$$

Notice that for the vertices w and w' the substitutions we obtain from (7) are

$$x_{w,c} \mapsto \sum_{(b,b')\neq(c,c')} x_{i,b}x_{i',b'} \ , \tag{8a}$$

$$x_{w',c'} \mapsto \sum_{(b,b')\neq(c,c')} x_{i,b}x_{i',b'} \ , \tag{8b}$$

$$x_{w,b} \mapsto 0 \qquad (\text{for } c \neq b), \tag{8c}$$

$$x_{w',b'} \mapsto 0 \qquad (\text{for } c' \neq b'), \tag{8d}$$

since w always gets colour c and w' always gets colour c' in any colouring $\chi_{(b,b')}$.

Let us next discuss how the polynomials obtained from (2a)–(2c) after the substitution (7) can be derived in PC from (6a)–(6c). More precisely we argue that all substituted axioms can be derived from the equations

$$\sum_{b=1}^{k} x_{i,b} = 1 \ , \tag{9a}$$

$$x_{i,b}x_{i,b'} = 0 \qquad \text{(for } b \neq b'\text{)}, \tag{9b}$$

$$\sum_{b'=1}^{k} x_{i',b'} = 1 \ , \tag{9c}$$

$$x_{i',b}x_{i',b'} = 0 \qquad \text{(for } b \neq b'\text{)}, \tag{9d}$$

$$x_{i,c}x_{i',c'} = 0 \ , \tag{9e}$$

which are just the same, except for variables renaming, as the pigeon axioms (6a) and (6b) for pigeons i and i' plus the collision axiom (6c) for the hole which is the common neighbour of i and i'. In what follows we will need the equation

$$\sum_{b=1}^{k}\sum_{b'=1}^{k} x_{i,b}x_{i',b'} - x_{i,c}x_{i',c'} - 1 = 0 \tag{10}$$

which has the degree-2 proof

$$\sum_{b=1}^{k} x_{i,b}\left(\sum_{b'=1}^{k} x_{i',b'} - 1\right) + \left(\sum_{b=1}^{k} x_{i,b} - 1\right) - x_{i,c}x_{i',c'} = 0 \tag{11}$$

from (9a)–(9e).

We consider first axioms $\sum_{j=1}^{k} x_{v,j} = 1$ as in (2a) for vertices v that are not a pigeon vertex i or i'. It is straightforward to verify that such an axiom after substitution as in (7) becomes an equality on the form (10). If v is a pigeon vertex i or i', then no substitution is made and we simply keep the axiom (9a) or (9c), respectively.

Next, we consider axioms (2b) on the form $x_{v,j}x_{v,j'} = 0$, where we assume that v is not a pigeon vertex i or i' since in that case we have one of the axioms (9b) and (9d). After substitution an axiom (2b) for $v \notin \{i, i'\}$ becomes a sum of degree-4 terms of the form $x_{i,b_1}x_{i',b'_1}x_{i,b_2}x_{i',b'_2}$. Recall that the substitution associates disjoint sets of pairs (b, b') to the colours for v. Therefore, for each term $x_{i,b_1}x_{i',b'_1}x_{i,b_2}x_{i',b'_2}$ it must be that either $b_1 \neq b_2$ or $b'_1 \neq b'_2$ holds, and such a term can be derived from (9b) or (9d) by multiplication.

Let us finally consider axioms on the form $x_{u,j}x_{v,j} = 0$ for $(u, v) \in E(G)$ as in (2c). There is no edge between i and i' in our constructed graph, so for the size of the intersection between $\{u, v\}$ and $\{i, i'\}$ it holds that $0 \leq \left|\{u, v\} \cap \{i, i'\}\right| \leq 1$.

If $\left|\{u, v\} \cap \{i, i'\}\right| = 0$, then after substitution the axiom (2c) becomes a sum of degree-4 terms of the form $x_{i,b_1}x_{i',b'_1}x_{i,b_2}x_{i',b'_2}$. Consider any such term. If either $b_1 \neq b_2$ or $b'_1 \neq b'_2$, then the term can be derived from (9b) or (9d). We claim that no term can have $b_1 = b_2$ and $b'_1 = b'_2$. To see this, note that this would imply that when performing substitution as in (7) the variables $x_{u,j}$ and $x_{v,j}$ both get expanded to a sum containing $x_{i,b_1}x_{i',b'_1}$. But this would in turn mean that the colouring $\chi_{(b_1,b'_1)}$ that we fixed for the gadget $G_{(i,i')\not\leftarrow(c,c')}$ at the start of the proof assigned colours $\chi_{(b_1,b'_1)}(u) = \chi_{(b_1,b'_1)}(v)$, which is impossible since there is an edge between u and v and $\chi_{(b_1,b'_1)}$ was chosen to be a legal colouring.

The remaining case is when we have intersection size $\left|\{u, v\} \cap \{i, i'\}\right| = 1$. Without loss of generality because of symmetry we can assume that we have an axiom $x_{u,j}x_{v,j} = 0$ for

$u \notin \{i, i'\}$ and $v = i$. The axiom becomes after substitution a sum of terms of the form $x_{i,b}x_{i',b'}x_{i,j}$. If for some term we would have $b = j$, then $\chi_{(j,b')}$ would assign the same colour j to both u and i. This is again impossible since $\chi_{(j,b')}$ is a legal colouring of the gadget by construction. Hence we have $b \neq j$ and it follows that $x_{i,b}x_{i',b'}x_{i,j}$ is derivable from (9b).

We are now almost done with the proof of Lemma 3.5. We have defined how to substitute variables $x_{v,j}$ in (2a)–(2c) and have shown that the equations that we obtain after these substitutions can be derived from Equations (6a)–(6c) in low degree. The final issue that remains it to get rid of all vertices γ_t in the pre-colouring gadget in Figure 2 that are not members of any injectivity constraint gadget $G_{(i,i')\not\leftarrow(c,c')}$. For such variables the substitution is simply an assignment: we let $x_{\gamma_t,b} \mapsto 1$ when $t \equiv b \pmod k$ and $x_{\gamma_t,b} \mapsto 0$ otherwise.[3] This immediately satisfies all axioms (2a) and (2b) for these vertices, removing these axioms from the refutation. It remains to check the axioms (2c) for any pair of connected vertices γ_t and $\gamma_{t'}$. But by construction, if γ_t and $\gamma_{t'}$ are connected it holds that $t \not\equiv t' \pmod k$. Therefore, for every $b \in [k]$ we have that either $x_{\gamma_t,b} \mapsto 0$ or $x_{\gamma_{t'},b} \mapsto 0$ holds, regardless of whether these two vertices are in some gadget $G_{(i,i')\not\leftarrow(c,c')}$ or not.

To summarize what we have done, we started with any arbitrary refutation of (2a)–(2c) and substituted all variables with degree-2 polynomials over the variables $x_{i,j}$ for $i \in [n]$. Then we proved that all these substituted axioms (and therefore the whole refutation) follow from Equations (9a)–(9c). It is straightforward to verify that, up to variable renaming, these axioms are nothing other than the FPHP axioms in (6a)–(6c). This concludes the proof of Lemma 3.5. Putting everything together, we can now state and prove our main theorem.

▶ **Theorem 3.6.** *For any integer $k \geq 3$ there is an efficiently constructible family of graphs $\{G_n\}_{n\in\mathbb{N}}$ with $\mathrm{O}(k^4 n)$ vertices of degree $\mathrm{O}(k^2)$ that do not possess k-colourings, but for which the corresponding system of polynomial equations* (2a)–(2c) *require degree $\Omega(n)$, and hence size $\exp(\Omega(n))$, to be refuted in polynomial calculus.*

Proof. Take the family of bipartite graphs $\{B_n\}_{n\in\mathbb{N}}$ as in Theorem 3.1 and apply Proposition 3.4 to this family. This yields a family of graphs $\{G_n\}_{n\in\mathbb{N}}$ as in the theorem statement. Any sublinear degree refutation for k-colouring of G_n would imply, by Lemma 3.5, a sublinear degree refutation for the functional pigeonhole principle for B_n, but this is impossible by the choice of B_n. ◀

4 Short Proofs for k-Colouring Instances in Cutting Planes

Theorem 3.6 tells us that there are non-k-colourable graphs G_n for which it is impossible for polynomial calculus to certify non-k-colourability efficiently. As is clear from our reduction, the k-colouring formulas for these graphs are essentially obfuscated instances of the functional pigeonhole principle.

It is well-known that cutting planes can easily prove that pigeonhole principle formulas are unsatisfiable by just counting the number of pigeons and holes and deduce that the pigeons are too many to fit in the holes [16]. As we show in this section, the instances of k-colouring obtained via the reduction from FPHP also have short cutting planes refutations. What these refutations do is essentially to "de-obfuscate" the k-colouring formulas to recover the original functional pigeonhole principle instances, which can then be efficiently refuted.

[3] Note that here the substitution for $x_{\gamma_t,b}$ where $t \equiv b \pmod k$ is different from the one used for vertices that are members of some gadget $G_{(i,i')\not\leftarrow(c,c')}$ in (8a) and (8b). For variables $x_{\gamma_t,b}$ where $t \not\equiv b \pmod k$ the substitution is the same as in (8c) and (8d), though.

We are going to describe our cutting planes refutation as a decision tree such that at every leaf we have a cutting planes refutation of the formula restricted by the partial assignment defined by the tree branch reaching that leaf. These refutations of the restricted versions of the formula can then be combined to yield a refutation of the original, unrestricted formula as stated in Lemma 4.1 and Proposition 4.2. The proofs of these statements are fairly routine and we omit them in this conference version of the paper.

We recall that as discussed in Section 2 we will use $\sum_i a_i x_i \leq \gamma$ as an alias for $\sum_i -a_i x_i \geq -\gamma$ and $\sum_i a_i x_i = \gamma$ as an alias for the combination of $\sum_i a_i x_i \leq \gamma$ and $\sum_i a_i x_i \geq \gamma$. In particular, we will frequently write $x = b$ for some variable x and $b \in \{0,1\}$ as a shorthand for the pair of inequalities $x \leq b$ and $-x \leq -b$.

▶ **Lemma 4.1.** *Let $b \in \{0,1\}$ and suppose that there exists a cutting planes derivation $(B_1, \ldots, B_L)$ in length L of the inequality $\sum_i a_i x_i \leq \gamma$ from the system of inequalities $\mathcal{S} \cup \{x = b\}$. Then for some $K \in \mathbb{N}$ there is a CP derivation in length $\mathrm{O}(L)$ of the inequality*

$$(-1)^{1-b} K \cdot (x - b) + \sum_i a_i x_i \leq \gamma \tag{12}$$

from $\mathcal{S}$.

▶ **Proposition 4.2.** *Let G be a graph and $k \geq 2$ be a positive integer, and let $\mathcal{S}$ be the set of inequalities* (5a)–(5c) *for G and k. If for a fixed set of vertices $u_1, u_2, \ldots, u_\ell$ in G and every choice of colours $(c_1, c_2, \ldots, c_\ell) \in [k]^\ell$ for these vertices there is a CP refutation in length at most L of the set of inequalities $\mathcal{S} \cup \{x_{u_1,c_1} = 1, x_{u_2,c_2} = 1, \ldots, x_{u_\ell,c_\ell} = 1\}$, then there is a CP refutation of $\mathcal{S}$ in length $k^{\mathrm{O}(\ell)} \cdot L$.*

We can now state the main result of this section, namely that the hard k-colouring instances for polynomial calculus constructed in Section 3 are easy for cutting planes.

▶ **Proposition 4.3.** *Let B be a left-regular bipartite graph B with left degree k and bounded right degree $\mathrm{O}(k)$, and consider the graph $G = G(B)$ in Proposition 3.4. Then if there is no complete matching of the left-hand side of B into the right-hand side, then the set of inequalities* (5a)–(5c) *encoding the k-colouring problem on G has a cutting planes refutation in length $k^{\mathrm{O}(k)} \cdot |V(B)|^{\mathrm{O}(1)}$.*

Proof sketch. Consider the first k vertices $\gamma_1, \ldots, \gamma_k$ in the pre-colouring gadget in G as depicted in Figure 2, which form a k-clique. For every partial colouring $(c_1, c_2, \ldots, c_k) \in [k]^k$ of this k-clique we build a cutting planes refutation of

$$\mathcal{S} \cup \{x_{\gamma_1,c_1} = 1, x_{\gamma_2,c_2} = 1, \ldots, x_{\gamma_k,c_k} = 1\} \ . \tag{13}$$

The result then follows by combining all of these refutations using Proposition 4.2.

Fix a choice of colours $(c_1, c_2, \ldots, c_k) \in [k]^k$. Notice that if some colour occurs twice in this tuple, then we can derive contradiction in length $\mathrm{O}(1)$ from (13) since one of the edge axioms (5c) is violated. Suppose therefore that $(c_1, c_2, \ldots, c_k)$ is a permutation of $[k]$. We will construct a CP refutation of (13) in length $k^{\mathrm{O}(k)} \cdot |V(B)|^{\mathrm{O}(1)}$.

The system of inequalities $\mathcal{S}$ is symmetric which respect to the permutation of the colour indices, so without loss of generality we focus on giving a refutation for

$$\mathcal{S} \cup \{x_{\gamma_1,1} = 1, x_{\gamma_2,2} = 1, \ldots, x_{\gamma_k,k} = 1\} \ . \tag{14}$$

The equations $\{x_{\gamma_1,1} = 1, x_{\gamma_2,2} = 1, \ldots, x_{\gamma_k,k} = 1\}$ taken together with $\mathcal{S}$ allow us to efficiently infer $x_{\gamma_i, i \bmod k} = 1$ for all the vertices γ_i, $i \in [M]$, in the gadget in Figure 2 (where

we recall from Section 2 that we identify colours 0 and k when convenient). The resulting set of equalities and inequalities $\mathcal{S} \cup \{x_{\gamma_i, i \bmod k} = 1 \mid i \in [M]\}$ is essentially an encoding of the k-colouring problem for the partially colored graph $\widehat{G}$ in Lemma 3.3 consisting of the gadgets in Figure 1. Indeed, since the partial assignment $\{x_{\gamma_1,1} = 1, x_{\gamma_2,2} = 1, \ldots, x_{\gamma_k,k} = 1\}$ forces the colours of all vertices γ_i, $i \in [M]$, in Figure 2, this gives us back the pre-coloured vertices in the gadgets in Figure 1.

As argued in (the proof of) Lemma 3.3, $\widehat{G}$ is the union of at most $\mathrm{O}\big(k^2|V(B)|\big)$ injectivity constraint gadgets $G_{(i,i')\not\leftarrow(c,c')}$ that forbid pigeons i and i' taking their cth and c'th edges, respectively, colliding in some hole j. If we introduce the alias $p_{i,j}$ for $x_{i,c}$, where j is the hole to which the cth edge from pigeon i leads, then our goal can be described as deriving the pigeonhole axiom $p_{i,j} + p_{i',j} = x_{i,c} + x_{i',c'} \leq 1$ from the set of inequalities of the corresponding gadget $G_{(u,v)\not\leftarrow(c,c')}$. We will see shortly how to do so in length $\mathrm{O}\big(k^{\mathrm{O}(k)}\big)$. Once we extract these pigeonhole inequalities we observe that the collection of these inequalities together with the inequalities (5a) form a cutting plane encoding

$$\sum_{j \in J(i)} p_{i,j} \geq 1 \qquad i \in I, \tag{15a}$$

$$p_{i,j} + p_{i',j} \leq 0 \qquad i \neq i' \in I,\ j \in J(i) \cap J(i'). \tag{15b}$$

of the graph pigeonhole principle on the bipartite graph B with left-hand side I and right-hand side J. Such a system of inequalities has a cutting plane refutation in length $\mathrm{O}\big(|V(B)|^3\big)$ [16].

In order to derive $x_{i,c} + x_{i',c'} \leq 1$ we consider the inequalities involving vertices of $G_{(i,i')\not\leftarrow(c,c')}$ plus the equations $x_{i,c} = 1$ and $x_{i,c'} = 1$. By Claim 3.2 this is an unsatisfiable system of inequalities of size $\mathrm{O}(k)$. By the refutational completeness of cutting planes, and using Lemma 4.1 twice, we obtain a derivation of $K_1(x_{i,c} - 1) + K_2(x_{i',c'} - 1) \leq -1$ in length $\exp(\mathrm{O}(k))$. Adding multiples of axioms on the form $x - 1 \leq 0$ we get the inequality $K(x_{i,c} - 1) + K(x_{i',c'} - 1) \leq -1$ for some positive integer K, and division by K yields $x_{i,c} + x_{i',c'} \leq 1$.

We have shown how to derive contradiction is length $k^{\mathrm{O}(k)}|V(B)|^{\mathrm{O}(1)}$ for any given colouring of the vertices $\gamma_1, \ldots, \gamma_k$. We take such refutations for all k^k possible ways of assigning colours to these vertices and joint them together using Proposition 4.2 into a refutation of the original, unrestricted formula. The proposition follows. ◀

5 Concluding Remarks

In this work we exhibit explicitly constructible graphs which are non-k-colourable but which require large degree in polynomial calculus to certify this fact for the canonical encoding of the k-colouring problem into polynomial equations over $\{0, 1\}$-valued variables. This, in turn, implies that the size of any polynomial calculus proof of non-k-colourability for these graphs must be exponential measured in the number of vertices.

Our degree lower bound also applies to a slightly different encoding with primitive kth roots of unity used in [19, 20] to build k-colouring algorithms based on Hilbert's Nullstellensatz. These algorithms construct certificates of non-k-colourability by solving linear systems of equations over the coefficients of all monomials up to a certain degree. The current paper yields explicit instances for which this method needs to consider monomials up to a very large degree, and therefore has to produce a linear system of exponential size. This answers an open question raised in, e.g., [21, 30].

This leads to an important observation, however. The degree lower bound applies to both polynomial encodings discussed above, but the size lower bound only applies to the encoding

using $\{0,1\}$-valued variables. It is still conceivable that proofs of non-k-colourability in the roots of unity encoding can be small although they must have large degree. This raises the following question.

▶ **Open Problem 5.1.** *Is there a family of non-3-colourable graphs such that any polynomial calculus proof of non-3-colourability using the roots of unity encoding must require large size?*

If the answer to the question is positive, then no matter how we choose the monomials to consider for the linear system construction in [19, 20], the size of the system will have to be large.

To further reduce the size of the linear system, the algorithms in [19] make use of the symmetries in the graphs. It is a natural question how much such an approach could help for our non-k-colourable instances. It seems plausible that if we apply our construction to a randomly generated bipartite graph with appropriate parameters, then the final graph will not have many symmetries except for the local symmetries inside the gadgets. In that case our lower bound might apply for the improved version of the algorithm as well.

One serious limitation of our result is that our hard graphs are very specific, and arguably somewhat artificial. For the weaker resolution proof system an average-case exponential lower bound has been shown for Erdős–Rényi random graphs $\mathcal{G}(n,p)$ where p is slightly above the threshold value $p_k(n)$ at which the graph becomes highly likely to be non-k-colourable [6]. It is natural to ask whether these instances are hard for polynomial calculus too.

▶ **Open Problem 5.2.** *Consider a random graph sampled according to $\mathcal{G}(n,p)$ with $p > p_k(n)$, so that the graph is non-k-colourable with high probability. Does polynomial calculus require large degree to certify non-k-colourability of such graphs with high probability?*

In this paper, we also show that the graph colouring instances that are provably hard for polynomial calculus are very easy for the cutting planes proof system. We do not quite believe that graph colouring is an easy problem for cutting planes, however, and it would be interesting to find explicit candidates for hard instances for cutting planes, even if proving the actual lower bounds may be very hard. This question is also interesting for the Lasserre/Sums-of-Squares proof system. Our instances seem likely to be easy for Lasserre, since they are based on the hardness of the pigeonhole principle and this combinatorial principle is easy for Lasserre.

▶ **Open Problem 5.3.** *Find candidates for explicit hard instances of non-3-colourability for cutting planes and for Lasserre/Sums-of-squares proof systems, and then prove formally that these instances are indeed hard.*

An intriguing observation is that even though the graph colouring instances in our paper are easy for cutting planes, results from the *Pseudo-Boolean Competition 2016* indicate that they are quite hard in practice for state-of-the-art pseudo-Boolean solvers [39]. This is even more interesting considering that the cutting planes refutations that we construct have small rank (i.e., the maximum number of application of the division rules along any path in the proof graph is small).

Acknowledgements

We are grateful to Mladen Mikša and Alexander Razborov for stimulating discussions and helpful feedback during various stages of this project. We would also like to thank Jan Elffers for running experiments with pseudo-Boolean solvers on instances obtained from our reduction

from functional pigeonhole principle formulas to graph colouring, demonstrating that these formulas are hard in practice. Last but not least, a big thanks to the anonymous CCC reviewers, who helped us catch some typos and bugs that really should not have been there, and whose suggestions helped improve the exposition considerably.

Part of this research was done while the first author was at KTH Royal Institute of Technology funded by the European Research Council (ERC) under the European Union's Seventh Framework Programme (FP7/2007–2013) / ERC grant agreement no. 279611. Later work at Universitat Politècnica de Catalunya for this project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement ERC-2014-CoG 648276 AUTAR). The second author was supported by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007–2013) / ERC grant agreement no. 279611 as well as by Swedish Research Council grants 621-2010-4797, 621-2012-5645, and 2016-00782.

References

1 Michael Alekhnovich, Eli Ben-Sasson, Alexander A. Razborov, and Avi Wigderson. Space complexity in propositional calculus. *SIAM Journal on Computing*, 31(4):1184–1211, 2002. Preliminary version in *STOC '00*.

2 Michael Alekhnovich and Alexander A. Razborov. Lower bounds for polynomial calculus: Non-binomial case. *Proceedings of the Steklov Institute of Mathematics*, 242:18–35, 2003. Available at `http://people.cs.uchicago.edu/~razborov/files/misha.pdf`. Preliminary version in *FOCS '01*.

3 Noga Alon and Michael Tarsi. Colorings and orientations of graphs. *Combinatorica*, 12(2):125–134, June 1992.

4 Roberto J. Bayardo Jr. and Robert Schrag. Using CSP look-back techniques to solve real-world SAT instances. In *Proceedings of the 14th National Conference on Artificial Intelligence (AAAI '97)*, pages 203–208, July 1997.

5 David Allen Bayer. *The Division Algorithm and the Hilbert Scheme*. PhD thesis, Harvard University, Cambridge, MA, USA, June 1982. Available at `https://www.math.columbia.edu/~bayer/papers/Bayer-thesis.pdf`.

6 Paul Beame, Joseph C. Culberson, David G. Mitchell, and Cristopher Moore. The resolution complexity of random graph k-colorability. *Discrete Applied Mathematics*, 153(1-3):25–47, December 2005.

7 Paul Beame, Russell Impagliazzo, Jan Krajíček, Toniann Pitassi, and Pavel Pudlák. Lower bounds on Hilbert's Nullstellensatz and propositional proofs. In *Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science (FOCS '94)*, pages 794–806, November 1994.

8 Richard Beigel and David Eppstein. 3-coloring in time $O(n^{1.3289})$. *Journal of Algorithms*, 54(2):168–204, February 2005.

9 Eli Ben-Sasson and Russell Impagliazzo. Random CNF's are hard for the polynomial calculus. *Computational Complexity*, 19:501–519, 2010. Preliminary version in *FOCS '99*.

10 Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow—resolution made simple. *Journal of the ACM*, 48(2):149–169, March 2001. Preliminary version in *STOC '99*.

11 Archie Blake. *Canonical Expressions in Boolean Algebra*. PhD thesis, University of Chicago, 1937.

12 Samuel R. Buss and Peter Clote. Cutting planes, connectivity and threshold logic. *Archive for Mathematical Logic*, 35:33–63, 1996.

13 Samuel R. Buss, Dima Grigoriev, Russell Impagliazzo, and Toniann Pitassi. Linear gaps between degrees for the polynomial calculus modulo distinct primes. *Journal of Computer and System Sciences*, 62(2):267–289, March 2001. Preliminary version in *CCC '99*.

14 Vašek Chvátal. Edmond polytopes and a hierarchy of combinatorial problems. *Discrete Mathematics*, 4(1):305–337, 1973.

15 Matthew Clegg, Jeffery Edmonds, and Russell Impagliazzo. Using the Groebner basis algorithm to find proofs of unsatisfiability. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC '96)*, pages 174–183, May 1996.

16 William Cook, Collette Rene Coullard, and György Turán. On the complexity of cutting-plane proofs. *Discrete Applied Mathematics*, 18(1):25–38, November 1987.

17 Jesús A. De Loera. Gröbner bases and graph colorings. *Beiträge zur Algebra und Geometrie*, 36(1):89–96, January 1995. Available at `https://www.emis.de/journals/BAG/vol.36/no.1/`.

18 Jesús A. De Loera, Susan Margulies, Michael Pernpeintner, Eric Riedl, David Rolnick, Gwen Spencer, Despina Stasi, and Jon Swenson. Graph-coloring ideals: Nullstellensatz certificates, Gröbner bases for chordal graphs, and hardness of Gröbner bases. In *Proceedings of the 40th International Symposium on Symbolic and Algebraic Computation (ISSAC '15)*, pages 133–140, July 2015.

19 Jesús A. De Loera, Jon Lee, Peter N. Malkin, and Susan Margulies. Hilbert's Nullstellensatz and an algorithm for proving combinatorial infeasibility. In *Proceedings of the 21st International Symposium on Symbolic and Algebraic Computation (ISSAC '08)*, pages 197–206, July 2008.

20 Jesús A. De Loera, Jon Lee, Peter N. Malkin, and Susan Margulies. Computing infeasibility certificates for combinatorial problems through Hilbert's Nullstellensatz. *Journal of Symbolic Computation*, 46(11):1260–1283, November 2011.

21 Jesús A. De Loera, Jon Lee, Susan Margulies, and Shmuel Onn. Expressing combinatorial problems by systems of polynomial equations and Hilbert's Nullstellensatz. *Combinatorics, Probability and Computing*, 18:551–582, July 2009.

22 Nicola Galesi and Massimo Lauria. On the automatizability of polynomial calculus. *Theory of Computing Systems*, 47:491–506, August 2010.

23 Nicola Galesi and Massimo Lauria. Optimality of size-degree trade-offs for polynomial calculus. *ACM Transactions on Computational Logic*, 12:4:1–4:22, November 2010.

24 Ralph E. Gomory. An algorithm for integer solutions of linear programs. In R.L. Graves and P. Wolfe, editors, *Recent Advances in Mathematical Programming*, pages 269–302. McGraw-Hill, New York, 1963.

25 Christopher J. Hillar and Troels Windfeldt. Algebraic characterization of uniquely vertex colorable graphs. *Journal of Combinatorial Theory, Series B*, 98(2):400–414, March 2008.

26 Thore Husfeldt. Graph colouring algorithms. In Lowell W. Beineke and Robin J. Wilson, editors, *Topics in Chromatic Graph Theory*, Encyclopedia of Mathematics and its Applications, chapter 13, pages 277–303. Cambridge University Press, May 2015.

27 Russell Impagliazzo, Pavel Pudlák, and Jiří Sgall. Lower bounds for the polynomial calculus and the Gröbner basis algorithm. *Computational Complexity*, 8(2):127–144, 1999.

28 Serge Lang. *Algebra*. Springer New York, 2005.

29 Daniel Le Berre and Anne Parrain. The Sat4j library, release 2.2. *Journal on Satisfiability, Boolean Modeling and Computation*, 7:59–64, 2010.

30 Bo Li, Benjamin Lowenstein, and Mohamed Omar. Low degree Nullstellensatz certificates for 3-colorability. *The Electronic Journal of Combinatorics*, 23(1), January 2016.

31 László Lovász. Stable sets and polynomials. *Discrete Mathematics*, 124(1–3):137–153, January 1994.

32 João P. Marques-Silva and Karem A. Sakallah. GRASP: A search algorithm for propositional satisfiability. *IEEE Transactions on Computers*, 48(5):506–521, May 1999. Preliminary version in *ICCAD '96*.

33 Yuri V. Matiyasevich. A criterion for vertex colorability of a graph stated in terms of edge orientations. *Diskretnyi Analiz*, 26:65–71, 1974. English translation of the Russian original. Available at `http://logic.pdmi.ras.ru/~yumat/papers/22_paper/`.

34 Yuri V. Matiyasevich. Some algebraic methods for calculating the number of colorings of a graph. *Journal of Mathematical Sciences*, 121(3):2401–2408, May 2004.

35 Colin McDiarmid. Colouring random graphs. *Annals of Operations Research*, 1(3):183–200, October 1984.

36 Mladen Mikša and Jakob Nordström. A generalized method for proving polynomial calculus degree lower bounds. In *Proceedings of the 30th Annual Computational Complexity Conference (CCC '15)*, volume 33 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 467–487, June 2015.

37 Michal Mnuk. Representing graph properties by polynomial ideals. In *Proceedings of the 4th International Workshop on Computer Algebra in Scientific Computing (CASC '01)*, pages 431–444, September 2001.

38 Matthew W. Moskewicz, Conor F. Madigan, Ying Zhao, Lintao Zhang, and Sharad Malik. Chaff: Engineering an efficient SAT solver. In *Proceedings of the 38th Design Automation Conference (DAC '01)*, pages 530–535, June 2001.

39 Pseudo-Boolean competition 2016: Results by benchmark for category no optimisation, small integers, linear constraints (DEC-SMALLINT-LIN). `http://www.cril.univ-artois.fr/PB16/results/globalbybench.php?idev=81&idcat=47`, 2016.

40 Sat4j: The Boolean satisfaction and optimization library in Java. `http://www.sat4j.org/`.