Computer Algebra, Combinatorics, and Complexity: Hilbert's Nullstellensatz and NP-complete Problems

by

Susan Margulies

B.A. (University of California at Berkeley) 1993

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Computer Science

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, DAVIS

Committee in charge:
Professor Jesús De Loera, Chair
Professor Zhaojun Bai
Professor Matthew Franklin
Professor Charles Martel

Summer 2008

The dissertation of Susan Margulies is approved:

Chair Date

Date

Date

Date

University of California, Davis

Summer 2008

Computer Algebra, Combinatorics, and Complexity: Hilbert's Nullstellensatz and NP-complete Problems

Susan Margulies

Abstract

Computer Algebra, Combinatorics, and Complexity: Hilbert's Nullstellensatz and NP-complete Problems

by

Susan Margulies

Doctor of Philosophy in Computer Science

University of California, Davis

Professor Jesús De Loera, Chair

Systems of polynomial equations over an algebraically-closed field $\mathbb{K}$ can be used to concisely represent combinatorial decision problems. In this way, a combinatorial problem is feasible (e.g., a graph is 3-colorable, Hamiltonian, etc.) if and only if a related system of polynomial equations has a solution over $\mathbb{K}$. If the system of polynomial equations has no solution, then Hilbert's Nullstellensatz yields a certificate that the underlying combinatorial problem is infeasible. We investigate an algorithm aimed at proving combinatorial infeasibility based on the experimentally-observed low degree of Hilbert's Nullstellensatz and large-scale, sparse linear algebra computations over $\mathbb{K}$.

We explore the Nullstellensatz Linear Algebra algorithm (**N**ul**LA**) from both a computational and a theoretical perspective. From the computational perspective, we compare computations over the rationals to computations over finite fields; we discuss mathe-

matical ideas for optimizing **N**ul**LA** ranging from the algebraic to the probabilistic, and we report on experiments proving the non-3-colorability of graphs with almost two thousand vertices and tens of thousands of edges.

From a theoretical perspective, we observe that if an NP-complete problem (e.g. graph 3-colorability) is represented as a system of polynomial equations, the resulting infeasibility certificate is a coNP certificate. Thus, if P $\neq$ NP and NP $\neq$ coNP, there must exist an infinite family of instances (e.g. an infinite family of graphs) where the minimum-degree of the associated Nullstellensatz certificate grows linearly in the input size and the certificates contain a super-polynomial number of monomials. In the case of graph 3-colorability, we show that the minimum-degree of a Nullstellensatz certificate (associated with a particular encoding) follows the sequence 1,4,7,..., etc.. In the case of the independent set decision problem, we show that a minimum-degree Nullstellensatz certificate (associated with a particular encoding) proving the non-existence of an independent set of size k is equal to the size of the largest independent set in the graph. Moreover, such a Nullstellensatz certificate contains one monomial for each independent set in the graph.

Professor Jesús De Loera
Dissertation Committee Chair

To my family,

Caryl, Bill, David and Susanna Margulies,

for their unwavering support.

Contents

List of Figures **iv**

List of Tables **v**

1 Introduction **1**

2 Encodings **6**
2.1 Independent Set . . . 9
2.2 Graph k-Colorability . . . 10
2.3 Hamiltonian Cycle . . . 19
2.4 Graph Planarity . . . 26
2.5 Edge-Chromatic Number . . . 30
2.6 Max-Cut . . . 32
2.7 SAT . . . 33

3 Nullstellensatz Linear Algebra Algorithm (NulLA) **35**
3.1 Hilbert's Nullstellensatz . . . 36
3.2 **N**ul**LA**: Examples, Pseudocode and Running Time . . . 41

4 Theoretical Complexity of NulLA **47**
4.1 P, NP and the Nullstellensatz . . . 47
4.2 Independent Set and the Nullstellensatz . . . 59
4.3 Graph 3-Coloring and the Nullstellensatz . . . 72
4.3.1 **N**ul**LA** 2-colorability is in P . . . 73
4.3.2 Minimum-degree non-3-colorability Nullstellensatz certificates . . . 74
4.3.3 Subgraphs . . . 85
4.4 SAT and the Nullstellensatz . . . 95

5 Experimental Results **97**
5.1 Four Mathematical Ideas to Optimize **N**ul**LA** . . . 97
5.1.1 $\mathbb{C}$ vs. $\mathbb{Q}$ and $\overline{\mathbb{F}_p}$ vs. $\mathbb{F}_p$ . . . 98
5.1.2 Subgraph Equations as Degree-cutters . . . 99
5.1.3 Alternative Nullstellensätze . . . 102

5.1.4 Probabilistic Nullstellensätze . . . 105
5.2 Graph 3-colorability Experimental Results . . . 108
5.2.1 Methods . . . 108
5.2.2 Test Cases . . . 109
5.2.3 Experimental Results over $\mathbb{Q}$ . . . 110
5.2.4 Experimental Results over $\mathbb{F}_2$ . . . 111
5.2.5 **N**ul**LA** over $\mathbb{F}_2$ vs. other graph coloring algorithms . . . 115
5.2.6 Hard Instances of 3-colorability . . . 120
5.3 Beyond 3-colorability . . . 126

6 Summary and Future Work **131**
6.1 Summary . . . 131
6.2 Future Work . . . 133

Bibliography **142**

List of Figures

2.1 Via Schnyder's theorem, $P(G)$ has dimension at most three. 27

4.1 Turán graph $T(5,3)$ 64
4.2 Converting G (the 5-odd-wheel) to H (the 3-odd-wheel) via node merges. . 87
4.3 The odd-wheels are non-3-colorable. 89
4.4 The $(2k+1)$-odd-wheel to the $(2(k+1)+1)$-odd-wheel. 91

5.1 Koester graph 101
5.2 A graph with a degree four non-3-colorability certificate over $\mathbb{F}_2$. 103
5.3 Probability tests on cliques and odd-wheels over $\mathbb{Q}$. 106
5.4 Probability tests on odd-wheels and cat-ear graphs over $\mathbb{F}_2$. 128
5.5 Probability tests on Kneser and flower graphs over $\mathbb{F}_2$. 128
5.6 $3, 4$ and 5 flowers. 129
5.7 $2, 3$ and 4 cat-ears. 129
5.8 A uniquely 3-colorable graph, the Grötzsch graph, and the Jin graph. . . . 129
5.9 4-critical, near-4-clique-free minimum unsolvable graphs (MUGs). 130
5.10 An example of a Liu-Zhang 4-CGU. 130

6.1 The line graphs. 137
6.2 The Jin graph and a 4-critical subgraph. 139

List of Tables

5.1 Experimental investigations of graph 3-colorability over $\mathbb{Q}$. 112
5.2 Graphs without 4-cliques over $\mathbb{F}_2$. 113
5.3 Original graph vs. non-3-colorable subgraph. 114
5.4 Graphs with 4-cliques over $\mathbb{F}_2$. 115
5.5 **N**ul**LA**, GB and AT on odd-wheel graphs. 118
5.6 **N**ul**LA** vs. Branch-and-Cut and DSATUR. 118
5.7 **N**ul**LA**, GB, AT on graphs with 4-cliques. 119
5.8 **N**ul**LA**, GB, AT on graphs without 4-cliques. 119
5.9 Hard instances of graph 3-colorability: MUGs. 123
5.10 Hard instances of graph 3-colorability: 4-CGUs. 124
5.11 K_n: minimum-degrees for non-$(n-1)$-colorability certificates. 126

6.1 Line graphs: minimum-degrees for non-hamiltonicity 137
6.2 Minimum-degrees for IND_n 139

Acknowledgments

The process of writing this dissertation was helped along immeasurably by dozens of people who were extremely busy with dozens of projects of their own.

I would like to thank Bernd Sturmfels and Dave Bayer for their support of this project from its earliest stages to the final draft of this dissertation, and I would like to thank Eric Rains for his timely suggestion of exploring encodings over finite fields, which evolved into our most significant computational success.

I would like to thank Chris Hillar for his thoughts and suggestions regarding the Hamiltonian cycle problem, and Alex Woo for his extensive help at the drop of a hat on topics ranging from encodings to Gröbner bases to abstract algebra.

I would like to thank my committee members, Zhaojun Bai, Matthew Franklin and Chip Martel, for kindly reading my thesis over the summer break, and for facilitating my unexpected graduation. I would particularly like to thank Dr. Martel for his extensive mentoring throughout my last quarter in regards to the Chancellor's Teaching Fellowship; co-teaching 122A with him was a pleasure and an inspiration.

I would also like to thank my **N**ul**LA** collaborators Jon Lee and Peter Malkin. I would like to thank Peter Malkin for the inspiring pictures on his blackboard, for his software engineering expertise, for his unfailing ability to make clean code cleaner and fast code faster, and for his generosity during the last few weeks, when he read and re-read sections of my thesis with inexhaustible patience. I would like to thank Jon Lee for arranging the cluster of 2 GHz machines with 12 MBytes of RAM on behalf of IBM specifically for the **N**ul**LA** experiments, for his "hello from [insert exotic location here]" emails which were

always a mixture of quirky humor and 10 papers' worth of off-hand comments in the form of "you might want to try this...", for his unfailing latex expertise, and in the end, for his prompt response to my panicked emails begging for last-minute help reading sections of my thesis as my deadlines loomed like a darkening storm... his last comment to me of "write faster!" was the most helpful of all.

And finally, I would like to thank my thesis advisor, Jesús De Loera, for suggesting the Nullstellensatz Linear Algebra algorithm to me in the very beginning as my dissertation topic; for his patience during my stumbling moments of inarticulate ignorance; for his guidance and insight during critical moments of despair; for his encouragement, optimism and support, and finally for his commitment to academic excellence– without his exacting high standards, this dissertation would be less than half of what it is.

Chapter 1

Introduction

"If I have the belief that I can do it,
I shall surely acquire the capacity to do it
even if I may not have it at the beginning."
–Mohandas Karamchand Gandhi,
1869 - 1948 .

"Perplexity is the beginning of knowledge."
–Kahlil Gibran,
1883-1931 .

It is well-known that systems of polynomial equations over an algebraically-closed field can yield compact representations of combinatorial problems. This contrasts with the exponential sizes of systems of *linear* inequalities that describe the convex hull of incidence vectors of many combinatorial structures (see [61]).

In [1], N. Alon surveys the use of non-linear polynomials in solving combinatorial problems. Although this technique is not yet as widely used by combinatorists as polyhedral or probabilistic techniques, the literature in this subject continues to expand. Prior work on encoding combinatorial properties includes colorings [2, 38, 19, 24, 41, 43, 44, 49, 39], independent sets [38, 36, 41, 58, 40], matchings [20], and flows [2, 49, 47].

While these polynomial system encodings often suggest an algorithmic approach to solving combinatorial problems (see [1] and therein), they have not yet been widely used for computation. A key issue that we investigate in this dissertation is the use of such polynomial systems to efficiently decide whether a graph, or other combinatorial structure, has a property captured by the polynomial system and its associated ideal. We call this the *combinatorial feasibility problem*. We are particularly interested in whether this can be accomplished in practice for large combinatorial structures, such as graphs with many vertices.

It is certainly well-known that the combinatorial feasibility problem can be solved using standard tools in computational algebra such as Gröbner bases. Nevertheless, it has been experimentally demonstrated that current Gröbner bases implementations often cannot directly solve polynomial systems with hundreds of equations. This dissertation proposes the Nullstellensatz Linear Algebra algorithm (**N**ul**LA**), which instead relies on the experimentally-observed low degrees of Hilbert's Nullstellensatz for *combinatorial* polynomial systems, and on large-scale, sparse linear algebra computations.

For a hard combinatorial problem (e.g., graph 3-colorability), we associate a system of polynomial equations $J = \{f_1 = \cdots = f_s = 0\}$ such that the system J has a solution if and only if the combinatorial problem has a feasible solution. Hilbert's Nullstellensatz (see e.g.,[13]) states that the system of polynomial equations has *no* solution over an algebraically-closed field $\mathbb{K}$ if and only if there exist polynomials $\beta_1, \ldots, \beta_s \in \mathbb{K}[x_1, \ldots, x_n]$ such that $1 = \sum \beta_i f_i$. Thus, when the polynomial system J has no solution, there exists a *certificate* that J has no solution, and thus a certificate that the combinatorial problem is

infeasible.

The central idea behind **N**ul**LA** is to generate a finite sequence of linear algebra systems based on Nullstellensatz certificates of increasing degree. These linear algebra systems eventually become *feasible* if and only if the underlying combinatorial problem is *infeasible*. Given a system of polynomial equations, we fix a tentative degree k for the coefficient polynomials β_i in the certificates. We decide whether there is a Nullstellensatz certificate with coefficients of degree $\leq k$ by solving a system of *linear* equations over the field $\mathbb{K}$ whose variables are in bijection with the coefficients of the monomials of the polynomials $\beta_1, \ldots, \beta_s$. If this linear system has a solution, we have found a certificate; otherwise, we repeat and try a higher degree for the polynomials β_i. This process is guaranteed to terminate because, in order for a Nullstellensatz certificate to exist, the degrees of the polynomials β_i cannot be more than known bounds (see [30] and references therein). We explain the details of **N**ul**LA** in Chapter 3.

Our method can be seen as a general-field variation of work by Lasserre [33], Laurent [34], Parrilo [53] and many others, who study the problem of minimizing a general polynomial function $f(x)$ over a real algebraic variety with finitely many points. Laurent proved that when the variety consists of the solutions to a zero-dimensional radical ideal I, the optimization problem $\min\{f(x) : x \in \text{variety}(I)\}$ is equivalent to a finite sequence of semidefinite programs terminating with the optimal solution (see [34]). There are two key observations that speed up practical calculations considerably: (1) when dealing with feasibility, rather than optimization, linear algebra replaces semidefinite programming, and (2) there are methods for controlling the length of the sequence of linear algebra systems,

including finite field computations instead of calculations over the reals, and the reduction of matrix size by symmetries [39]. See Section 5.1 for details.

From commutative algebra, there are well-known *upper bounds* on the degrees of the coefficients β_i in the Hilbert Nullstellensatz certificates for *general* systems of polynomials, and they turn out to be sharp (see [30]). For instance, the following well-known example (due to Mora, Lazard, Masser, Philippon, and Kollár) shows that the degree of β_1 is at least d^m :

$$f_1 = x_1^d,\ f_2 = x_1 - x_2^d,\ \ldots,\ f_{m-1} = x_{m-2} - x_{m-1}^d,\ f_m = 1 - x_{m-1}x_m^{d-1}\ .$$

But polynomial systems for combinatorial optimization problems are not necessarily pathologically complicated. In fact, polynomial systems for combinatorial optimization problems are often extremely symmetric with homogeneous polynomials of similar structure, and we now know that the upper bounds on their Nullstellensatz certificates are sometimes much lower. The natural question is: *How large are the minimum-degrees of the associated Nullstellensatz certificates of infeasibility?*

There is a fascinating connection between Hilbert's Nullstellensatz and computational complexity. As we will see in Section 4.1, unless P = NP , for every hard combinatorial problem, there must exist an infinite sequence of infeasible instances for which the minimum-degree of a Nullstellensatz certificate, for the associated system of polynomial equations, grows arbitrarily large. This was first observed by L. Lovász, who then proposed the problem of explicitly finding graphs exhibiting such growth (see [41]). A main contribution of this dissertation is to explicitly describe the growth in degree of specific families of graphs. In particular, we establish the following main theorem (Section 4.2):

Given a graph G, let $\alpha(G)$ denote the size of the largest independent set in G. A minimum-degree Nullstellensatz certificate (associated with the Lovász encoding of Lemma 2.1.1) for the non-existence of an independent set of size greater than $\alpha(G)$ has degree equal to $\alpha(G)$ and contains at least one monomial per independent set in G.

This dissertation is organized as follows. In Chapter 2, we describe various encodings for an assortment of NP-complete and combinatorial decision problems. We survey existing encodings for independent set, graph k-colorability and SAT, and present new encodings for Hamiltonian cycle, graph k-colorability, max cut, edge chromatic number and graph planarity. In Chapter 3, we describe the Nullstellensatz Linear Algebra (**N**ul**LA**) algorithm. In Chapter 4, we explore the relationship between Nullstellensatz certificates and complexity theory, touching P vs. NP, NP vs. coNP and NP as a proper or improper subset of EXPTIME. We also prove a range of theoretical results concerning growth in the minimum-degree of both non-k-colorability, and non-existence of an independent set of size k, Nullstellensatz certificates. In Chapter 5, we present our experimental results, and a variety of mathematical techniques for optimizing **N**ul**LA**, and in Chapter 6, we presents our conclusions and suggestions for future directions of research.

Chapter 2

Encodings

"The formulation of a problem is often more essential
than its solution, which may be merely a matter
of mathematical or experimental skill."
–Albert Einstein,
1879–1955 .

In this chapter, we demonstrate the ease and simplicity of encoding combinatorial problems as systems of polynomial equations. Our constant goal throughout this chapter is computation with algebraic methods; towards that end, we will often present more than one encoding for the same combinatorial problem. The purpose of these multiple encodings is not only to demonstrate a variety of encoding techniques, but also to eventually compare and contrast encodings from a computational perspective (see Chapter 5).

By *encoding*, we simply mean the following: given an instance of a "yes/no" decision question, we can convert it in polynomial-time (in the input size of the instance) to a system of polynomial equations such that the system has a solution if and only if the underlying instance has a "yes" answer. It is important to emphasize the polynomial-time nature of these conversions; otherwise, we could simply solve the "yes/no" decision ques-

tion beforehand and represent every "yes" instance as the equation "1=1", and every "no" instance as "1=0". This definition is formalized as follows:

Definition 2.0.1 *Given a language L, if there exists a polynomial-time algorithm A that takes as input a string I, and produces as output a system of polynomial equations such that the system has a solution if and only if $I \in L$, then we say that the system of polynomial equations* encodes I.

We use the formal terminology of languages (see [59] for relevant background), instead of the more familiar vocabulary of *yes/no* decision problems to bypass any technical difficulties with the "encodings" of the instances themselves (such as adjacency lists vs. adjacency matrices of graphs). However, since *yes/no* decision questions are equivalent to languages, we will often use these terms interchangeably, especially when referring to NP-complete problems. We also note that, because Definition 2.0.1 stipulates that the algorithm A constructing the system of equations runs in *polynomial* time, this means that the system of equations can be written down in *polynomial* space. Intuitively, when converting a problem from the representation stipulated by the language (such as the adjacency matrix or adjacency list of a graph) to a system of polynomial equations, we must avoid introducing an exponential number of variables or equations, or introducing degrees or coefficients are exponentially long in terms of bit-length. However, given a polynomial $g(n)$ where n is the input size of the instance, by stipulating that the encoding can be constructed in $O(g(n))$ time, we can be certain of the following bounds: given a system of polynomial equations $f_1 = \cdots = f_s = 0$ that encodes an instance I, the number of equations is $O(g(n))$, the number of variables is $O(g(n))$, the bit-size of $\max\{\deg(f_1), \ldots, \deg(f_s)\}$ is $O(g(n))$, the

maximum number of monomials in any f_i is $O(g(n))$, and bit-size of the largest coefficient in any f_i is $O(g(n))$. This idea is formalized in the following definition, which we use extensively in Section 4.1:

Definition 2.0.2 *Given a language L and a polynomial $g(n)$, if the algorithm A for encoding a string I of length n runs in $O(g(n))$ time, then we say that L has an $O(g(n))$-*encoding.

In this chapter, we present encodings for independent set (Section 2.1), graph k-colorability (Section 2.2), Hamiltonian cycle (Section 2.3), graph planarity (Section 2.4), edge-chromatic number (Section 2.5), max-cut (Section 2.6), and finally, SAT (Section 2.7).

We note that by displaying an encoding for SAT, we can easily construct encodings for *all* NP-complete problems via polynomial reductions to SAT. However, this approach is not computationally practical because of the increase in the number of variables/equations in the systems. Furthermore, we will see later that encodings using constraints of the form $x_i(x_i - 1)$ seem to behave badly with algebraic methods such as **N**ul**LA** (see Section 4.2), but other type of constraints (e.g. root of unity constraints such as in graph 3-colorability) do behave better in practice (see Chapter 5).

Although the encodings presented in this chapter are not always very interesting in and of themselves, we explore their value in terms of providing insight into the underlying problem (Section 5.2.6), or in terms proving bounds on their associated identities (Section 4.2), or most importantly, in terms of being able to compute effectively on large combinatorial structures, such as graphs. The foremost question for us is the following: how can we best capture the combinatorial structure of an NP-complete problem with respect to Hilbert's Nullstellensatz?

Before we begin, we clarify our terms and notation: Adj(i) denotes the set of nodes adjacent to node i; a *zero-dimensional* system of equations is a system of equations with a finite number of solutions; every encoding presented is over $\mathbb{C}$ (the complex numbers) unless otherwise specified, and all graphs are assumed to be simple. For all other concepts and basic definitions from graph theory, we refer to [16].

2.1 Independent Set

Given a graph G, a *stable set* or *independent set* in G is a subset of vertices such that no two vertices in the subset are adjacent. The size of the largest independent set in G is called the *stability number*, or *independence number*, of G, and is denoted by $\alpha(G)$. The decision question of determining whether a given graph has an independent set of size k is NP-complete [21], and can be encoded as the following system of polynomial equations:

Lemma 2.1.1 (L. Lovász [41]) *A graph G has an independent set of size k if and only if the following zero-dimensional system of equations*

$$\begin{aligned} x_i^2 - x_i = 0 \ , &\quad \textit{for every node } i \in V(G) \ , \\ x_i x_j = 0 \ , &\quad \textit{for every edge } \{i,j\} \in E(G) \ , \\ \sum_{i=1}^{n} x_i - k = 0 \ , & \end{aligned}$$

has a solution. Moreover, the number of solutions equals the number of distinct independent sets of size k.

Proof: If the graph G has an independent set I of size k, we simply assign the variables associated with vertices in I to have value one, and all other variables to have value zero.

Then, the above system of equations is satisfied. If the above system of equations is satisfied, the equations $x_i^2 - x_i = 0$ force every variable to take on 0/1 values. The equations $x_i x_k = 0$ show that no two vertices assigned the value one are adjacent. The last equation shows that exactly k variables have the value one: thus, those variables correspond directly to vertices in an independent set of size k.

We now show that the number of solutions equals the number of distinct independent sets of size k. We have previously shown that independent sets correspond to solutions and solutions correspond to independent sets, but we now show that the map between the two sets is bijective. Given any independent set of size k, we simply assign every variable in the independent set to be one and all other variables to be zero. Thus, given two independent sets, I_1 and I_2, if they map to the same solution (the same 0/1 vector), then the same vertices must have been present in both sets and $I_1 = I_2$. Thus, the map is injective, or one-to-one. Since we have already shown that every solution maps to an independent set (the map is surjective or onto), we have shown that the map is bijective and the number of solutions is equal to the number of distinct independent sets of size k. □

In Section 4.2, we explicitly describe the Nullstellensatz certificates associated with this system of polynomial equations.

2.2 Graph k-Colorability

The problem of graph k-colorability is as follows: given a graph G and an integer k, can G be colored with k colors such that no two adjacent vertices have the same color?

This problem is known to be NP-complete [21]. In addition to being hard, it is also an interesting practical problem, having applications to fields as varying as compiler optimization and scheduling. In this section, we present several encodings for graph k-colorability: a degree-k encoding over $\mathbb{C}$, a degree-2 encoding over $\mathbb{C}$, and our most computationally successful encoding, a degree k-encoding over $\overline{\mathbb{F}_p}$ (the algebraic closure of the finite field with p elements).

We extensively studied the encodings described in this section, both from the theoretical perspective (Section 4.3), and from the computational perspective (Chapter 5). We begin with the following remark:

Lemma 2.2.1 *Given a graph G with n vertices, let I be the ideal*

$$I = \left\langle x_1^k - 1, \ldots, x_n^k - 1, \underbrace{\sum_{d=0}^{k-1} x_i^{k-1-d} x_j^d}_{\{i,j\} \in E(G)} \right\rangle .$$

Then I is a radical ideal. Moreover, $\dim(\mathbb{C}[x_1, \ldots, x_n]/I)$ *is equal to the number of points in variety(I).*

Proof: Recall that a "square-free" polynomial is a polynomial with no repeated factors (e.g., $(x+3)^2(x+6)$ is not square-free, whereas $(x+3)(x+6)$ is square-free). For every x_i, the ideal I contains the square-free, univariate polynomial $x_i^3 - 1$. Thus, I is a radical ideal by [12], pg. 39-41, Proposition 2.7. Therefore, by [13], pg. 232, Proposition 8 (ii),

$$\dim\left(\frac{\mathbb{C}[x_1, \ldots, x_n]}{I}\right) = |V(I)| .$$

□

We will see later that $|V(I)|$ is equal to the number of k-colorings of a graph multiplied by $k!$. Therefore, Lemma 2.2.1 allows us to compute the number of k-colorings of a graph by computing the dimension (as a vector space) of the quotient ring $\mathbb{C}[x_1, \ldots, x_n]/I$.

Our first encoding, degree-k over $\mathbb{C}$, is a generalization of an encoding proposed by Bayer for the case of 3-colorability.

Lemma 2.2.2 (Bayer [5]) *A graph G is k-colorable if and only if the following zero-dimensional system of equations*

$$\begin{aligned} x_i^k - 1 &= 0 \ , \qquad \textit{for every node } i \in V(G) \ , \\ \sum_{d=0}^{k-1} x_i^{k-1-d} x_j^d &= 0 \ , \qquad \textit{for every edge } \{i,j\} \in E(G) \ , \end{aligned}$$

has a solution. Moreover, the number of solutions equals the number of distinct k-colorings multiplied by $k!$.

Proof: Assume that G is k-colorable, and assign the colors to the k roots of unity. Clearly, the vertex equations ($x_i^k - 1 = 0$) are satisfied. Since the graph is k-colorable, there exists a coloring such that no two adjacent vertices have the same color. Therefore, given an edge $\{i,j\} \in E(G)$, the root of unity assigned to x_i does not equal the root of unity assigned to x_j, and we see that

$$\frac{x_i^k - x_j^k}{x_i - x_j} = \sum_{d=0}^{k-1} x_i^{k-1-d} x_j^d \quad \Longrightarrow \quad x_i^k - x_j^k = (x_i - x_j) \sum_{d=0}^{k-1} x_i^{k-1-d} x_j^d = 0 \ .$$

Since $x_i \neq x_j$, the factor $x_i - x_j \neq 0$. Thus, $\sum_{d=0}^{k-1} x_i^{k-1-d} x_j^d = 0$, and the edge equations are satisfied.

Conversely, assume there exists a solution to the above system of equations. The vertex equations ($x_i^k - 1 = 0$) force every variable to take on the value of one of the k-th

roots of unity. We will now show that no two adjacent vertices are assigned the same color. To prove this, assume the contrary: assume that for some edge $\{i, j\}$, x_i and x_j are both assigned the *same* root of unity, β. Then,

$$0 = \sum_{d=0}^{k-1} x_i^{k-1-d} x_j^d = k\beta^{k-1} \neq 0$$

Thus, every adjacent pair of vertices is assigned a different color.

We now show that the number of solutions equals number of distinct k-colorings multiplied by $k!$. We have previously shown that k-colorings correspond to solutions and solutions correspond to k-colorings, but we now show that the map between the two sets is bijective. We first explicitly map the k colors to the k roots of unity. For example, in the case of $k = 3$, we assign red $\rightarrow e^{2\pi i/3}$, green $\rightarrow e^{4\pi i/3}$, and blue $\rightarrow e^{2\pi i} = 1$. Under this map, if two colorings of the graph, C_1 and C_2, map to the same solution, then the two colorings are the same. Thus, the map is injective, or one-to-one. Since we have already shown that every solution maps to a coloring (the map is surjective or onto), we have shown that the map is bijective. Since there are $k!$ ways to assign k colors to the k roots of unity, the number of solutions is equal to the number of distinct k-colorings multiplied by $k!$. □

The following corollary concerning k-colorable subgraphs easily follows from this result.

Corollary 2.2.3 *A graph G has a k-colorable subgraph with R edges if and only if the following zero-dimensional system of equations has a solution:*

$$\sum_{\{i,j\} \in E(G)} y_{ij} - R = 0 \ .$$

For every vertex $i \in V(G)$:

$$x_i^k - 1 = 0 \ .$$

For every edge $\{i,j\} \in E(G)$:

$$y_{ij}^2 - y_{ij} = 0, \quad y_{ij}\big(x_i^{k-1} + x_i^{k-2}x_j + \cdots + x_j^{k-1}\big) = 0 \ .$$

Proof: For a k-colorable subgraph H with R edges, we set $y_{ij} = 1$ if edge $\{i,j\} \in E(H)$ or 0 otherwise. By Lemma 2.2.2, the resulting subsystem of equations has a solution. Conversely, from a solution, the subgraph H in question is described by $y_{ij} = 1$. By Lemma 2.2.2 above, a solution maps to a k-coloring. □

We now present a new degree two encoding of graph k-colorability, based on the idea of "partitioning" the graph into k disjoint sets. Thus, every vertex appears in exactly one partition, and the order of the partitions defines the order of the cycle.

Lemma 2.2.4 *Let $\mathbb{C}[x_{ip}]$, where $1 \le i \le n$ and $1 \le p \le k$, be a polynomial ring. A graph G is k-colorable if and only if the following zero-dimensional system of equations has a solution.*

For every node $i \in V(G)$:

$$\left(\sum_{p=1}^{k} x_{ip}\right) - 1 = 0 \ ,$$

For every edge $\{i,j\} \in E(G)$ and $p = 1, \ldots, k$:

$$x_{ip}x_{jp} = 0 \ .$$

For every node $i \in V(G)$ and $p = 1, \ldots, k$:

$$x_{ip}(x_{ip} - 1) = 0 \ .$$

Proof: If the graph G is k-colorable, then assign x_{ir} to be 1 if vertex i has the r-th color, and all other x_{ip} to be 0. Clearly, the first and third equations are satisfied. Furthermore, since no two adjacent vertices have the same color, no two adjacent vertices are in the same partition; thus, the second equation is satisfied.

Conversely, assume that there exists a solution to the above system of equations. By the first and third equations, clearly every vertex appears in only one partition. Furthermore, by the second equation, no two adjacent vertices are in the same partition. Thus, the vertex/partition mapping corresponds to a k-coloring. □

Finally, we come to our most computationally successful encoding: graph k-coloring over $\overline{\mathbb{F}_p}$, where k and p are relatively prime. Before we describe this encoding, we introduce a few well-known facts from algebra.

Lemma 2.2.5 *The equation $x^n - 1 = 0$ has n distinct roots over $\overline{\mathbb{F}_p}$, when p is relatively prime to n.*

Proof: The *discriminant* of a polynomial $f(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_0$ is defined to be

$$\mathrm{disc}(f) = \frac{(-1)^{n(n-1)/2}}{a_n}\mathrm{Res}(f, f')$$

When the discriminant is non-zero, f does not have multiple roots. In this case, $f(x) = x^n - 1$, and $f'(x) = nx^{n-1}$. The resultant is the determinant of the Sylvester matrix,

displayed below:

$$\mathrm{Syl}(f,f') = \left|\begin{array}{cccccccccc}
1 & 0 & \cdots & \cdots & 0 & -1 & 0 & \cdots & \cdots & 0 \\
0 & 1 & 0 & \cdots & \cdots & 0 & -1 & 0 & \cdots & 0 \\
\vdots & & \ddots & \ddots & & & \ddots & \ddots & & \vdots \\
\vdots & & & \ddots & \ddots & & & \ddots & \ddots & 0 \\
0 & \cdots & \cdots & 0 & 1 & 0 & \cdots & \cdots & 0 & -1 \\
n & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\
0 & n & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\
\vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\
0 & \cdots & \cdots & 0 & n & 0 & \cdots & \cdots & \cdots & 0
\end{array}\right| \begin{array}{l} \left.\begin{array}{l}\\\\\\\\\\\end{array}\right\} n-1 \text{ rows} \\ \left.\begin{array}{l}\\\\\\\\\end{array}\right\} n \text{ rows} \end{array}$$

Thus, the discriminant is $\pm n^n \mod p$. This is easy to see by expanding around the last n rows of the Sylvester matrix when taking the determinant. Since $\gcd(n,p) = 1$, $\pm n^n \not\equiv 0 \mod p$, and because the discriminant is non-zero, the roots of the equation $x^n - 1 = 0$ are distinct. □

Lemma 2.2.6 *A graph G is k-colorable if and only if the following zero-dimensional system of equations*

$$\begin{aligned} x_i^k - 1 &= 0 \ , \qquad \textit{for every node } i \in V(G) \ , \\ \sum_{d=0}^{k-1} x_i^{k-1-d} x_j^d &= 0 \ , \qquad \textit{for every edge } \{i,j\} \in E(G) \ , \end{aligned}$$

has a solution over $\overline{\mathbb{F}_p}$, where k and p are relatively prime. Moreover, the number of solutions equals the number of distinct k-colorings multiplied by $k!$.

Proof: This proof follows from Lemmas 2.2.2 and 2.2.5. Since the k distinct roots of unity (in this case, $\omega, \omega^2, \ldots, \omega^{k-1}, 1$) can be mapped to the k distinct colors, we see again that the number of solutions equals the number of distinct k-colorings multiplied by $k!$. □

We conclude with an observation that foreshadows a result from our experimental investigations: although the colorings of a given graph are completely described by a system of polynomial equations such as Lemma 2.2.6, it may sometimes be computationally useful to add *extra* equations to these systems (see Chapter 5, Section 5.1.2). These extra equations should capture extra combinatorial properties of the underlying graph, which allow us to simplify our computations. For example, when testing for 3-colorability, it is logical to search the graph for triangles as a means of reducing computation time. In our case, a triangle formed by the vertices $\{i, j, k\}$ forces the variables x_i, x_j, x_k to take on different colors/roots of unity. In order to formalize the idea of a *triangle equation*, we need the following well-known fact from algebra:

Proposition 2.2.7 ([17]) *If G is a finite group such that, for all positive integers n dividing its order, G contains at most n elements x satisfying $x^n - 1 = 0$, then G is cyclic.*

Therefore, the group formed by the roots of unity of $x^k - 1 = 0$ is cyclic, regardless of whether the field is $\mathbb{C}$ or $\overline{\mathbb{F}_p}$. The roots of unity over $\mathbb{C}$ are generated by powers of the primitive root $e^{2\pi i/k}$, while the roots of unity over $\overline{\mathbb{F}_p}$ can be described as powers of a primitive root ω:

$$\omega, \omega^2, \ldots, \omega^{k-1}, 1$$

This observation leads to the following lemma.

Lemma 2.2.8 *Given a graph G and an integer k, encoded as the system of polynomial equations from Lemma 2.2.2 or 2.2.6, if G contains a k-clique $\{x_{i_1}, x_{i_2}, \ldots, x_{i_k}\}$ as a subgraph and d is any integer such that $d \not| k$, then the following equation*

$$x_{i_1}^d + x_{i_2}^d + \cdots + x_{i_k}^d = 0 \tag{2.1}$$

can be added to the system of equations without changing the set of solutions.

Proof: If the system of equations from Lemma 2.2.2 or 2.2.6 has a solution, we must show that any such solution also satisfies Eq. 2.1. The following proof applies to the system of equations from either lemma. When the system of equations (either system) has a solution, no two adjacent vertices are assigned the same color. In particular, the vertices $\{i_1, i_2, \ldots, i_k\}$ corresponding to the k-clique are each assigned a different color. Therefore, the corresponding variables $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ represent the complete set of the k roots of unity. If the variables $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ are *each* raised to the d power, the values of the roots are simply permuted (since d is not a factor of k); therefore, $x_{i_1}^d, x_{i_2}^d, \ldots, x_{i_k}^d$ also represents the complete set of the k roots of unity. Furthermore, recall

$$x^k - 1 = (x-1)(x^{k-1} + x^{k-2} + \cdots + x + 1) = 0 \ .$$

Therefore, any primitive root satisfies $(x^{k-1} + x^{k-2} + \cdots + x + 1) = 0$, and the sum of the complete set of the k roots of unity is zero. Thus, if the system of equations has a solution, then the system of equations, *along with Eq. 2.1*, also has a solution. □

In Chapter 5, Section 5.1.2, we will see the computational advantage of including these types of subgraph equations, and the advantage of having flexibility in choosing the degree d.

2.3 Hamiltonian Cycle

A Hamiltonian cycle is a cycle that passes through every vertex exactly once. Given a graph G and an integer k, it is NP-complete to determine if G has a Hamiltonian cycle, or if G has a cycle of length k [21]. In this section, we describe an encoding for finding a cycle of length k in a graph, and then as a corollary, describe an encoding for Hamiltonian cycle. We also describe a graph-theoretic application that arises naturally from these encodings. We conclude by presenting an encoding of Hamiltonian cycle over $\overline{\mathbb{F}_p}$, and also demonstrating a degree two encoding similar to the encoding presented for graph k-colorability.

Lemma 2.3.1 *A simple graph G has a cycle of length L if and only if the following zero-dimensional system of polynomial equations has a solution:*

$$\left(\sum_{i=1}^{n} y_i\right) - L = 0 \ . \tag{2.2}$$

For every node $i = 1, \ldots, n$:

$$y_i(y_i - 1) = 0 \ , \qquad \prod_{s=1}^{n}(x_i - s) = 0 \ , \tag{2.3}$$

$$y_i \prod_{j \in Adj(i)} (x_i - y_j x_j + y_j)(x_i - y_j x_j - y_j(L-1)) = 0 \ . \tag{2.4}$$

Proof: Suppose that a cycle C of length L exists in the graph G. We set $y_i = 1$ or 0 depending on whether or not node i is in C. Next, starting the numbering at any node of C, we set $x_i = j$ if node i is the j-th node of C. It is easy to check that Eqs. 2.2 and 2.3 are satisfied. Note that if vertex i is not in C, x_i can be set to any value between 1 and n.

To verify Eq. 2.4, note that because C has length L, if vertex i is the j-th node of the cycle, then one of its neighbors, say k, must be the "follower", namely the $(j+1)$-th element of the cycle. If $j < L$, then the factor $(x_i - x_k + 1) = (x_i - (x_k - 1)) = 0$ appears in the product equation associated with the i-th vertex, and the product is zero. If $j = L$, then the factor $(x_i - x_k - (L-1)) = 0$ appears, and the product is again 0. For all other vertices x_i not in C, simply set x_i to any value between 1 and n (satisfying Eq. 2.3), and then "turn them off" by setting $y_i = 0$, which causes Eq. 2.4 to be automatically satisfied. Thus, every equation in the polynomial system is satisfied.

Conversely, from a solution of the system above, we see that L variables y_i are "turned on"; let these variable be the set C. Furthermore, we see that every variable takes on a value between 1 and n. We claim that the nodes $i \in C$ form a cycle. Because $y_i \neq 0$, the polynomial of Eq. 2.4 must vanish. Thus, for every $j \in C$,

$$\text{either} \quad (x_i - x_j + 1) = 0 \ , \quad \text{or} \quad (x_i - x_j - (L-1)) = 0 \ .$$

Note that Eq. 2.4 reduces to this form when y_i and a particular y_j equal one. Therefore, either vertex i is adjacent to a vertex j (with $y_j = 1$) such that x_j equals the *next integer value* ($x_i + 1 = x_j$), or $x_j = x_i - L + 1$. Consider the variable x_{i_1} which takes on the *smallest* value of any variable in C. For x_{i_1}, Eq. 2.4 cannot cancel by being adjacent to a variable $x_j = x_{i_1} - L + 1$ (because then x_{i_1} would not be the *smallest* value in C). Thus, for variable x_{i_1}, Eq. 2.4 must cancel because it is adjacent to a variable x_{i_2} which takes on the *next highest value*. This condition holds for the next $L - 2$ variables in C: Eq. 2.4 cancels because each variable is adjacent to a variable taking on the *next highest value*. Thus, the variables in C traverse a consecutive sequence of integers from x_{i_1}

to x_{i_L}. Therefore, x_{i_L} must take on the *largest* value of any variable in the set C: thus, it must be adjacent to a variable taking on the value $x_j = x_{i_L} + L - 1$. The only point in C satisfying that criteria is x_{i_1}. Thus, the points in C describe a cycle of length L in G. □

We have the following corollary.

Corollary 2.3.2 *A graph G has a Hamiltonian cycle if and only if the following zero-dimensional system of equations has a solution. For every node $i \in V(G)$, we have two equations:*

$$\prod_{s=1}^{n}(x_i - s) = 0 \ , \quad \textit{and} \quad \prod_{j \in Adj(i)} (x_i - x_j + 1)(x_i - x_j - (n-1)) = 0 \ .$$

The number of Hamiltonian cycles in the graph equals the number of solutions of the system divided by $2n$.

Proof: Clearly when $L = n$ we can just fix all y_i to 1, thus many of the equations simplify or become obsolete. We only have to check the last statement on the number of Hamiltonian cycles. For that, we remark that no solution appears with multiplicity because the ideal is radical. That the ideal is radical is implied by the fact that every variable appears as the only variable in a unique square-free polynomial (see page 246 of [31]). Finally, note for every cycle there are n ways to choose the initial node to be labeled as 1, and then two possible directions to continue the labeling. □

These results also apply to directed graphs; thus, we can also consider paths or cycles with orientation. We can also use the polynomials systems above to investigate the

distribution of cycle lengths in a graph (similarly for path lengths and cut sizes). This topic has several outstanding questions. For example, a still unresolved question of Erdös and Gyárfás [57] asks: If G is a graph with minimum-degree three, is it true that G always has a cycle having length that is a power of two? We define the *cycle-length polynomial* as the square-free univariate polynomial whose roots are the possible cycle lengths of a graph (same can be done for cuts). Considering L as a variable, the reduced lexicographic Gröbner basis (with L the last variable) computation provides us with a unique univariate polynomial on L that is divisible by the cycle-length polynomial of G.

Before we display our encoding over $\overline{\mathbb{F}_p}$, where p and n are relatively prime, we recall that there exists a primitive root of unity for $x^n - 1 = 0$, via Proposition 2.2.7 from Section 2.2.

Lemma 2.3.3 *Let G be a connected graph with n vertices. Then G has a Hamiltonian cycle if and only if the following zero-dimensional system of equations*

$$x_i^n + 1 = 0 \ , \quad \textit{for every node } i \in V(G) \ ,$$
$$\textstyle\prod_{j \in Adj(i)} (\omega x_i + x_j) = 0 \ , \quad \textit{for every node } i \in V(G) \ ,$$

has a solution over $\overline{\mathbb{F}_p}$, where p and n are relatively prime, and ω is an n-th primitive root of unity.

Proof: Suppose there exists a Hamiltonian cycle in the graph G. We can begin the numbering at an arbitrary vertex, and assign the vertices to be powers of ω; thus, a Hamiltonian cycle $v_{i_1}, v_{i_2}, \ldots, v_{i_n}$ implies $x_{i_1} = \omega, x_{i_2} = \omega^2$, and $x_{i_n} = \omega^n = 1$. Clearly, the vertex equations $x_i^n - 1 = 0$ are satisfied. The edge equations are also satisfied, because an variable x_i

is adjacent to the *next highest power* of ω in the cycle. Thus, $\omega x_{i_1} - x_{i_2} = \omega \cdot \omega - \omega^2 = 0$. Therefore, every edge equation is satisfied.

Conversely, from a solution to the system above, we see that every variable is assigned a root of unity. Consider the *lowest power* of ω assigned to any variable. Because the edge equations are satisfied, every variable must be adjacent to a variable taking on the value of the *next highest power* of ω. Since there are n variables, we must eventually come to a variable assigned the value $\omega^n = 1$. That variable must be adjacent to the variable taking on the value ω; thus, the *lowest power* of ω assigned to any variable is one, and the order of the powers of ω define a Hamiltonian cycle in G. □

We observe that this encoding, as is, does not necessarily lend itself to computation. In order to compute with this encoding, we would have to treat ω as an extra value in our field, and compute over the splitting field $\mathbb{F}_p \cup \omega$. This encoding is also easily transferable to an encoding over $\mathbb{C}$; however, the computational difficulty of computing over the splitting field $\mathbb{Q} \cup \omega$ is similar. However, by *adding* a few equations to the encoding, ω can be treated as a variable and not as a fixed primitive n-th root of unity. For example, if the equation $\omega^n - 1 = 0$, and the set of equations

$$y_k(\omega^k - 1) = 0 \ ,$$

where k is any factor dividing n, are added to the encoding of Lemma 2.3.3, then ω is simply a variable which can only take on the value of a *primitive* n-th root of unity, even if n is not a prime number. Another set of equations which ensures ω is a variable only taking on

the value of a *primitive* n-th root of unity is the following:

$$\omega^{k(n-1)} + \omega^{k(n-2)} + \cdots + \omega^k + 1 \ , \quad \text{for } 1 \leq k \leq n \ .$$

We can also use the *cyclotomic polynomial* [17], denoted by $\Phi(n)$, which is the polynomial whose roots are the primitive n-th roots of unity. For example, the first ten cyclotomic polynomials are as follows:

$$\begin{aligned}
&\Phi_1(x) = \omega - 1 \ , && \Phi_6(x) = \omega^2 - \omega + 1 \ , \\
&\Phi_2(x) = \omega + 1 \ , && \Phi_7(x) = \omega^6 + \omega^5 + \omega^4 + \omega^3 + \omega^2 + \omega + 1 \ , \\
&\Phi_3(x) = \omega^2 + \omega + 1 \ , && \Phi_8(x) = \omega^4 + 1 \ , \\
&\Phi_4(x) = \omega^2 + 1 \ , && \Phi_9(x) = \omega^6 + \omega^3 + 1 \ , \\
&\Phi_5(x) = \omega^4 + \omega^3 + \omega^2 + \omega + 1 \ , && \Phi_{10}(x) = \omega^4 - \omega^3 + \omega^2 - \omega + 1 \ .
\end{aligned} \tag{2.5}$$

We conclude by presenting a degree two encoding of Hamiltonian cycle. This encoding is similar to the one presented for graph k-colorability, and is again based on the idea of "partitioning" the graph into n disjoint sets. Thus, every vertex appears in exactly one partition, and the order of the partitions defines the order of the cycle.

Lemma 2.3.4 *Let $\mathbb{C}[x_{ip}]$, where $1 \leq i, p \leq n$, be a polynomial ring. Given a simple graph G, then G has a Hamiltonian cycle if and only if the following zero-dimensional system of equations has a solution.*

For every node $i \in V(G)$:

$$\left(\sum_{p=1}^{n} x_{ip} \right) - 1 = 0 \ , \quad \textit{and} \quad \sum_{j \in Adj(i)} \left(x_{in} x_{j1} + \sum_{p=1}^{n-1} x_{ip} x_{j(p+1)} \right) - 1 = 0 \ .$$

For every node $i \in V(G)$ *and* $p = 1, \ldots, n$*:*

$$x_{ip}(x_{ip} - 1) = 0 \ .$$

Proof: If the graph G has a Hamiltonian cycle, then start the labeling at an arbitrary vertex in the cycle, and assign x_{ir} to be 1 if vertex i is the r-th vertex on the cycle, and all other x_{ip} to be 0. Clearly, the first and third equations are satisfied. The second equation is satisfied since every vertex i is adjacent to a vertex j which is placed in the *next highest partition*, or vertex i is the n-th vertex on the cycle, in which case it is adjacent to the vertex placed in the first partition.

Conversely, assume that there exists a solution to the above system of equations. By the first and third equations, clearly every vertex appears in only one partition. Consider a vertex i in the *lowest possible partition*. That vertex cannot be in the n-th partition, because, in that case, the second equation is only satisfied if it is adjacent to a vertex j in the first partition (thus, contradicting the minimality of the partition of i). Thus, vertex i must be adjacent to a vertex j in the *next highest partition*. This condition holds for the n vertices in the graph. Finally, since the vertex in the n-th partition cannot be adjacent a vertex in a higher partition, it must be adjacent to a vertex in the first partition. Thus, the vertex in the lowest possible partition must be in the first partition, and the vertex/partition order defines a Hamiltonian cycle in the graph. □

We conclude by noting that the encodings presented in this section are not $O(g(n))$-encodings for any polynomial $g(n)$. For example, in Lemma 2.3.1, for any vertex i the equation $\prod_{s=1}^{n}(x_i - s) = 0$ has 2^n terms when expanded. Lemma 2.3.3 is also not an

$O(g(n))$-encoding for any polynomial $g(n)$, because, given a graph containing a vertex i with degree $n-1$, the equation $\prod_{j \in Adj(i)}(\omega x_i - x_j)$ has 2^{n-1} terms when expanded. However, it may yet be possible to compute using these encodings because the polynomials can be represented concisely as the product of linear factors. Algebraic methods exploiting these linear factors have yet to be developed or investigated. Despite these problems with the encoding size, we recall that the Hamiltonian cycle problem remains NP-complete even for k-regular graphs with $k \geq 3$ [54]. In this case, when restricted to k-regular graphs for a fixed k, Lemma 2.3.3 is an $O(n)$-encoding where n is the number of vertices in the graph.

2.4 Graph Planarity

A *planar* graph is any graph that can be embedded (or drawn) in the plane in such a way that no two edges cross (see [16] for details). Although graph planarity is solvable in polynomial-time, we present an encoding as a system of polynomial equations for two reasons: 1) we are interested in whether or not this system of polynomial equations is likewise solvable in polynomial time, and 2) can the techniques displayed in this encoding be used for other combinatorial problems. The encoding we present here for testing graph planarity is based on Schnyder's characterization of planarity in terms of the dimension of a *partially-ordered set* or *poset* [55]: For an n-element poset P, a *linear extension* is an order-preserving bijection $\sigma : P \to \{1, 2, \ldots, n\}$. The *poset dimension* of P is the smallest integer t for which there exists a family of t linear extensions $\sigma_1, \ldots, \sigma_t$ of P such that $x < y$ in P if and only if $\sigma_i(x) < \sigma_i(y)$ for all σ_i. The *incidence poset* $P(G)$ of a graph $G(V, E)$ is the partially-ordered set of height two on the union of nodes and edges, where we say $x < y$

if x is a node and y is an edge, and y is incident to x.

Example 2.4.1 (Posets and Planar Graphs) Let G be the planar square graph. Here we display the incidence poset $P(G)$ and its three corresponding linear extensions.

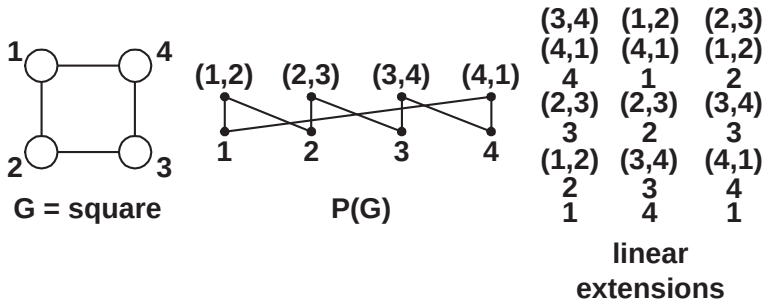

Figure 2.1: Via Schnyder's theorem, $P(G)$ has dimension at most three.

Lemma 2.4.2 (Schnyder's theorem [55]) *A graph G is planar if and only if the poset dimension of $P(G)$ is no more than three.*

We begin by encoding the decision question of poset dimension as a system of polynomial equations.

Lemma 2.4.3 *Let $P = (E, <)$ be a poset, and $\mathbb{C}[x_{\{i\}k}, \Delta_{\{ij\}k}, s_k]$ be a polynomial ring in $p|E|+(|E|^2-|E|)+p$ variables (where $i = 1, \ldots, |E|$, $j = 1, \ldots, |E|$, $j \neq i$, and $k = 1, \ldots, p$). Then P has poset dimension at most p if and only if the following system of equations has a solution:*

For $k = 1, \ldots, p$:

$$\prod_{s=1}^{|E|}(x_{\{i\}k} - s) = 0 \ , \quad \textit{for every } i \in \{1, \ldots, |E|\} \ , \quad \textit{and}$$

$$s_k \left(\prod_{1 \le i < j \le |E|} x_{\{i\}k} - x_{\{j\}k} \right) - 1 = 0 \ . \tag{2.6}$$

For $k = 1, \ldots, p$ *, and every ordered pair of comparable elements* $e_i > e_j$ *in* P *:*

$$x_{\{i\}k} - x_{\{j\}k} - \Delta_{\{ij\}k} = 0 \ . \tag{2.7}$$

For every ordered pair of incomparable elements of P *(i.e.,* $e_i \not> e_j$ *and* $e_j \not> e_i$*) :*

$$\prod_{k=1}^{p} \left(x_{\{i\}k} - x_{\{j\}k} - \Delta_{\{ij\}k} \right) = 0 \ , \qquad \prod_{k=1}^{p} \left(x_{\{j\}k} - x_{\{i\}k} - \Delta_{\{ji\}k} \right) = 0 \ . \tag{2.8}$$

For $k = 1, \ldots, p$ *, and for every pair* $\{i, j\} \in \{1, \ldots, |E|\}$*:*

$$\prod_{d=1}^{|E|-1} (\Delta_{\{ij\}k} - d) = 0 \ , \qquad \prod_{d=1}^{|E|-1} (\Delta_{\{ji\}k} - d) = 0 \ .$$

Proof: With Eqs. 2.6 and 2.7, we assign distinct numbers 1 through $|E|$ to the poset elements, such that the properties of a linear extension are satisfied. Eqs. 2.6 and 2.7 are repeated p times, so p linear extensions are created. If the intersection of these extensions is indeed equal to the original poset P, then for every incomparable pair of elements in P, at least one of the p linear extensions must detect the incomparability. But this is indeed the case for Eq. 2.8, which says that for the l-th linear extension, the values assigned to the incomparable pair e_i, e_j do not satisfy $x_{\{i\}l} < x_{\{j\}l}$, but instead satisfy $x_{\{j\}l} > x_{\{i\}l}$. □

We will now use the encoding of poset dimension as a system of polynomial equations to capture Schnyder's criterion for graph planarity.

Lemma 2.4.4 *Given a simple graph $G(V,E)$ with n vertices and m edges, let $\mathbb{C}[z_{ij}, x_{\{i\}k}, y_{\{ij\}k}, \Delta_{\{ij,i\},k}, \Delta_{\{ij,uv\},k}, s_k]$ be a polynomial ring in $\big(m+3(2m+m(m-1)+m+n+1)\big)$ variables (where $1 \le i \le n, \{i,j\} \in E(G), \{u,v\} \in E(G)$, and $1 \le k \le 3$). Then G has a planar subgraph with K edges if and only if the following zero-dimensional system of equations has a solution:*

For every edge $\{i,j\} \in E(G)$:

$$z_{ij}^2 - z_{ij} = 0, \quad \sum_{\{i,j\} \in E(G)} z_{ij} - K = 0 \ .$$

For $k = 1,2,3$, every node $i \in V(G)$ and every edge $\{i,j\} \in E(G)$:

$$\prod_{s=1}^{n+m} (x_{\{i\}k} - s) = 0, \qquad \prod_{s=1}^{n+m} (y_{\{ij\}k} - s) = 0 \ ,$$

$$s_k \left(\prod_{\substack{i,j \in V(G) \\ i<j}} \big(x_{\{i\}k} - x_{\{j\}k}\big) \prod_{\substack{i \in V(G), \\ \{u,v\} \in E(G)}} \big(x_{\{i\}k} - y_{\{uv\}k}\big) \prod_{\{i,j\},\{u,v\} \in E(G)} \big(y_{\{ij\}k} - y_{\{uv\}k}\big) \right) = 1 \ .$$

For $k = 1,2,3$, and for every pair of $i \in V(G)$ and incident edge $\{i,j\} \in E(G)$:

$$z_{ij}\big(y_{\{ij\}k} - x_{\{i\}k} - \Delta_{\{ij,i\}k}\big) = 0 \ .$$

For every pair of node $i \in V(G)$ and edge $\{u,v\} \in E(G)$ that is not *incident on i:*

$$z_{uv}\big(y_{\{uv\}1} - x_{\{i\}1} - \Delta_{\{uv,i\}1}\big)\big(y_{\{uv\}2} - x_{\{i\}2} - \Delta_{\{uv,i\}2}\big)\big(y_{\{uv\}3} - x_{\{i\}3} - \Delta_{\{uv,i\}3}\big) = 0 \ ,$$

$$z_{uv}\big(x_{\{i\}1} - y_{\{uv\}1} - \Delta_{\{i,uv\}1}\big)\big(x_{\{i\}2} - y_{\{uv\}2} - \Delta_{\{i,uv\}2}\big)\big(x_{\{i\}3} - y_{\{uv\}3} - \Delta_{\{i,uv\}3}\big) = 0 \ .$$

For every pair of edges $\{i,j\},\{u,v\} \in E(G)$ (regardless of whether or not they share an endpoint):

$$z_{ij}z_{uv}\big(y_{\{ij\}1} - y_{\{uv\}1} - \Delta_{\{ij,uv\}1}\big)\big(y_{\{ij\}2} - y_{\{uv\}2} - \Delta_{\{ij,uv\}2}\big)\big(y_{\{ij\}3} - y_{\{uv\}3} - \Delta_{\{ij,uv\}3}\big) = 0 \ ,$$

$$z_{ij}z_{uv}\big(y_{\{uv\}1}-y_{\{ij\}1}-\Delta_{\{uv,ij\}1}\big)\big(y_{\{uv\}2}-y_{\{ij\}2}-\Delta_{\{uv,ij\}2}\big)\big(y_{\{uv\}3}-y_{\{ij\}3}-\Delta_{\{uv,ij\}3}\big)=0\ .$$

For every pair of nodes $i,j\in V(G)$ (regardless of whether or not they are adjacent):

$$\big(x_{\{i\}1}-x_{\{j\}1}-\Delta_{\{i,j\}1}\big)\big(x_{\{i\}2}-x_{\{j\}2}-\Delta_{\{i,j\}2}\big)\big(x_{\{i\}3}-x_{\{j\}3}-\Delta_{\{i,j\}3}\big)=0\ ,$$

$$\big(x_{\{j\}1}-x_{\{i\}1}-\Delta_{\{j,i\}1}\big)\big(x_{\{j\}2}-x_{\{i\}2}-\Delta_{\{j,i\}2}\big)\big(x_{\{j\}3}-x_{\{i\}3}-\Delta_{\{j,i\}3}\big)=0\ .$$

For every Δ_{index} (e.g., $\Delta_{\{ij,uv\}k},\Delta_{\{ij,i\}k}$, etc.) variable appearing in the above system:

$$\prod_{d=1}^{n+m-1}\big(\Delta_{index}-d\big)=0\ .$$

Proof: We simply apply Lemma 2.4.3 to the particular pairs of order relations of the incidence poset of the graph. Note that in the formulation, we have added variables z_{ij} that have the effect of "turning on or off" an edge of the input graph. □

2.5 Edge-Chromatic Number

The *edge-chromatic number* of a graph, denoted by $\chi'(G)$, is the minimum number of colors necessary to color every edge such that no two edges incident on the same vertex are the same color. It is easy to see that the edge-chromatic number is bounded from below by $\Delta(G)$, the largest vertex degree in the graph. Also, by Vizing's theorem [16], we know that any graph can be edge-colored with $\Delta(G)+1$ colors, and therefore, we see

$$\Delta(G)\le\chi'(G)\le\Delta(G)+1\ .$$

Thus, the question of determining edge-chromatic simplifies to determining whether $\chi'(G)$ is $\Delta(G)$ or $\Delta(G)+1$. This problem is NP-complete [25], and we encode it as the following system of polynomial equations.

Lemma 2.5.1 *Let G be a simple graph with maximum vertex degree Δ. The graph G has edge-chromatic number Δ if and only if the following zero-dimensional system of polynomials has a solution:*

For every edge $\{i,j\} \in E(G)$:

$$x_{ij}^{\Delta} - 1 = 0 \ . \tag{2.9}$$

For every node $i \in V(G)$:

$$s_i \left(\prod_{\substack{j,k \in Adj(i) \\ j<k}} (x_{ij} - x_{ik}) \right) - 1 = 0 \ . \tag{2.10}$$

Proof: If the system of equations has a solution, then Eq. 2.9 insures that all variables x_{ij} are assigned Δ roots of unity. Eq. 2.10 insures that no node is incident on two edges of the same color. Because the graph contains a vertex of degree Δ, the graph cannot have an edge-chromatic number less than Δ, and because the solution corresponds to an edge-Δ-coloring, this implies that the graph has edge-chromatic number exactly Δ. Conversely, if the graph has an edge-Δ-coloring, simply map the coloring to the Δ roots of unity and all equations are satisfied. Because Vizing's theorem states that any graph with maximum vertex degree Δ can be edge-colored with at most $\Delta+1$ colors, if there is no solution, then the graph must have an edge-chromatic number of $\Delta+1$. □

2.6 Max-Cut

A *cut* in an undirected graph $G(V, E)$ is a partition of the vertices into two nonempty sets, S and $V - S$. The *size* of the cut $(S, V - S)$ is the number of edges crossing the cut. The problem of finding the *smallest* cut in the graph is well-known to be solvable in polynomial-time via network flow algorithms, but the problem of finding the *largest* cut in the graph is NP-hard [52]. We represent the max-cut problem as a system of polynomial equations as follows:

Lemma 2.6.1 *Given a graph G, there is a cut of size K in G if and only if the following zero-dimensional system of equations has a solution:*

$$x_i^2 - 1 = 0 \ , \qquad \textit{for every } i \in V(G) \ ,$$
$$\big(|E| - 2K\big) - \left(\sum_{\{i,j\} \in E} x_i x_j \right) = 0 \ .$$

Proof: If G has a cut of size K, we assign every vertex in S to have the value 1 and every vertex in $V - S$ to have the value -1. Clearly, the vertex equations $x_i^2 - 1 = 0$ are satisfied. For the second equation, we note that every edge in the graph is contained in the sum: edges that do *not* span the cut are $1 \cdot 1 = 1$ or $(-1) \cdot (-1) = 1$. However, edges that span the cut are $1 \cdot (-1) = -1$. Thus, the second equation becomes $\big(|E| - 2K\big) - \big(|E| - K - K\big) = 0$.

Conversely, suppose there exists a solution to the system of polynomial equations. The vertex equations force every variable to be ± 1. By the second equation, there are exactly K edges with one endpoint assigned 1 and one endpoint assigned -1. Thus, a solution to the system corresponds directly to a cut of size K in the graph. □

2.7 SAT

A Boolean expression is *satisfiable* if there exists a truth assignment to the variables such that the expression evaluates to *true*. A Boolean expression is in *conjunctive normal form* if the clauses are separated by ANDs, and every clause is a sequence of literals separated by ORs. The *Boolean satisfiability* problem (SAT) is the problem of determining whether there exists a truth assignment such that a given Boolean expression ϕ in conjunctive normal form evaluates to *true*. SAT was the first decision problem to be proven NP-complete, via the Cook-Levin theorem of 1971 [21]. As such, encodings of SAT and their theoretical complexity have been a rich area of research, which we will summarize in Section 4.4.

The following encoding is most commonly used in research in logic and complexity (see [8, 26] and references therein).

Lemma 2.7.1 *A Boolean expression ϕ in conjunctive normal form is satisfiable if and only if the following zero-dimensional system of equations has a solution:*

$$x_i(x_i - 1) = 0 \ , \qquad \textit{for every variable } i \in \phi \ ,$$

$$\prod_{x_i \in C} (x_i - 1) \prod_{\neg x_j \in C} x_j = 0 \ , \qquad \textit{for every clause } C \textit{ in } \phi \ .$$

Proof: If ϕ is satisfiable, there must exist a truth assignment such that every clause evaluates to *true*. Let every variable in the system corresponding to a *true* variable in ϕ equal 1, and every variable in the system corresponding to a *false* variable in ϕ equal 0. Clearly, this assignment of values to variables causes the first equation to be satisfied. Now consider a clause C in ϕ. A clause evaluates to *true* if it contains at least one positive literal

x_i that is *true*, or it contains at least one negative literal $\neg x_i$ that is *false*. Therefore, either the literal appears in positive form, and $x_i = 1$, in which case $x_i - 1 = 0$, or the literal appears in negative form, and $\neg x_i = true$, in which case $x_i = 0$. In either case, the clause equations are satisfied.

Conversely, suppose there exists a solution to the system of polynomial equations. The first equation clearly forces every variable to be 0 or 1. The clause equations force every clause to have either a positive literal that is *true*, or a negative literal that is *false* (recall that when x_i is *false*, $\neg x_i$ is *true*). Thus, a solution to the system of equations corresponds directly to a satisfying truth assignment. □

There has also been extensive work on representing SAT as a system of inequalities and solving as a semidefinite program (see [3] and references therein).

Chapter 3

Nullstellensatz Linear Algebra Algorithm (NulLA)

Who among us would not be glad to lift the veil behind which the future lies hidden; to cast a glance at the next advances of our science and at the secrets of its development during future centuries? What particular goals will there be toward which the leading mathematical spirits of coming generations will strive?

–David Hilbert, 1900

The Nullstellensatz Linear Algebra Algorithm (**N**ul**LA**) takes an *input* a system of polynomial equations, and produces as *output* either **yes** or **no**. To be precise, **N**ul**LA** outputs either **no**, the system of polynomial equations has *no* solution, along with a certificate of *infeasibility*, or **yes**, the system of polynomial equations has a solution. In Section 3.1, we

will state and prove Hilbert's Nullstellensatz, borrowing the proof from [4]. In Section 3.2, we will thoroughly describe **N**ul**LA**, providing examples, pseudocode, and briefly discussing known upper bounds on its running time.

3.1 Hilbert's Nullstellensatz

David Hilbert (1862 – 1943) proved the *Nullstellensatz* or *theorem of zeros* in 1893 [23]. For our purposes, the Nullstellensatz is most conveniently stated in the following way: Given a system of polynomial equations $f_1(x) = \cdots = f_s(x) = 0$, where $f_i \in \mathbb{K}[x_1, \ldots, x_n]$ and $\mathbb{K}$ is an algebraically closed field, the system has *no* solution in $\mathbb{K}^n$ if and only if there exist polynomials $\beta_1, \ldots, \beta_s \in \mathbb{K}[x_1, \ldots, x_n]$ such that $1 = \sum \beta_i f_i$ [13]. The proof we display here, borrowed in its entirety from [4], is known for being almost completely self-contained: it relies on Noether's normalization lemma, and very little other external mathematical background.

Lemma 3.1.1 (Noether's normalization lemma) *If $\mathbb{K}$ is an infinite field and f is a nonconstant polynomial in $\mathbb{K}[x_1, \ldots, x_n]$ with $n \geq 2$, then it is possible to find $\lambda_1, \ldots, \lambda_{n-1}$ in $\mathbb{K}$ such that in*

$$f(x_1 + \lambda_1 x_n, \ldots, x_{n-1} + \lambda_{n-1} x_n, x_n) \ ,$$

the coefficient of x_n^d (where d is the total degree of f) is nonzero.

Proof: Let f_d be the homogeneous component of f of degree d. In other words, f can be written as $f = f_d + f_{\text{rest}}$. Consider the coefficient of x_n^d in $f(x_1 + \lambda_1 x_n, \ldots, x_{n-1} + \lambda_{n-1} x_n, x_n)$. Let $x_{i_1} x_{i_2} \cdots x_{i_d}$ represent any monomial of degree d in f_d, where $i_1, \ldots, i_d$ are

not necessarily distinct. Thus, when f is evaluated at $\{x_1 + \lambda_1 x_n, \ldots, x_{n-1} + \lambda_{n-1} x_n, x_n\}$, we see

$$x_{i_1} x_{i_2} \cdots x_{i_d} \Longrightarrow (x_{i_1} + \lambda_{i_1} x_n)(x_{i_2} + \lambda_{i_2} x_n) \cdots (x_{i_d} + \lambda_{i_d} x_n)$$
$$\Longrightarrow \text{the coefficient of } x_n^d \text{ is } \lambda_{i_1} \lambda_{i_2} \cdots \lambda_{i_d} \ .$$

Since all terms of degree d are collected in f_d, the coefficient of x_n^d in $f(x_1 + \lambda_1 x_n, \ldots, x_{n-1} + \lambda_{n-1} x_n, x_n)$ is

$$f_d(\lambda_1, \ldots, \lambda_{n-1}, 1) \ .$$

By induction on n, we can establish that $f_d(x_1, \ldots, x_{n-1}, 1)$ is a nonzero polynomial in $\mathbb{K}[x_1, \ldots, x_{n-1}]$, and since $\mathbb{K}$ is infinite, there is some point at which it does not vanish. Let this point be $(\lambda_1, \ldots, \lambda_{n-1})$. Therefore, we can be certain that $f_d(\lambda_1, \ldots, \lambda_{n-1}, 1)$ is nonzero. Therefore, the coefficient of x_n^d in $f(x_1 + \lambda_1 x_n, \ldots, x_{n-1} + \lambda_{n-1} x_n, x_n)$ is likewise nonzero. □

Thus, what we have shown here is that, given *any* non-constant polynomial f of degree d in *any* polynomial ring $\mathbb{K}[x_1, \ldots, x_n]$ (with $\mathbb{K}$ infinite and $n \geq 2$), we can translate the polynomial in such a way that the coefficient for a particular variable of degree d (for example, x_n^d) is nonzero. This lemma turns out to be critical to the subsequent proof of Hilbert's Nullstellensatz.

Theorem 3.1.2 (Hilbert's Nullstellensatz) *Let I be a proper ideal of $\mathbb{K}[x_1, \ldots, x_n]$. If $\mathbb{K}$ is algebraically closed, then there exists $(a_1, \ldots, a_n)$ in $\mathbb{K}^n$ such that $f(a_1, \ldots, a_n) = 0$ for all $f \in I$.*

Proof: Let us assume that $I \neq 0$, since otherwise the result is trivial (the only polynomial in I is the zero polynomial, and it vanishes on all points). We will prove the Nullstellensatz by induction on n.

In the case $n = 1$, any nonzero proper ideal $I \in \mathbb{K}[x]$ is a principal ideal since the univariate polynomial ring is a principal ideal domain. Therefore, I is generated by a single nonconstant polynomial, which must therefore have at least one root $a \in \mathbb{K}$, since $\mathbb{K}$ is algebraically closed. Thus, $f(a) = 0$ for all $f \in I$.

Now assume that the theorem holds for all proper ideals in $\mathbb{K}[x_1, \ldots, x_{n-1}]$. We will prove the theorem for a proper ideal $I \in \mathbb{K}[x_1, \ldots, x_n]$.

From Noether's normalization lemma (Lemma 3.1.1), we know that for *any* nonconstant $f \in I$, we can translate it in such a way that the coefficient x_n^d is nonzero. Therefore, by scaling accordingly, we can find a $g \in I$ that is *monic* in x_n.

Let I' be a proper ideal in $\mathbb{K}[x_1, \ldots, x_{n-1}]$ consisting of all of the polynomials in I that do not contain the variable x_n. Therefore, by the induction hypothesis, there exists a point $(a_1, \ldots, a_{n-1})$ such that $f(a_1, \ldots, a_{n-1}) = 0$ for all $f \in I'$. Now, consider the following claim:

- **Claim:** The set $J = \{f(a_1, \ldots, a_n, x_n) \mid f \in I\}$ is a proper ideal of $\mathbb{K}[x_n]$.
- **Proof:** Suppose, for the purpose of deriving a contradiction, there exists an $f \in I$ such that $f(a_1, \ldots, a_{n-1}, x_n) = 1$. Then, we can write f as

$$f = f_0 + f_1 x_n + \ldots + f_d x_n^d$$

where $f_i \in \mathbb{K}[x_1, \ldots, x_{n-1}]$ and

$$f_1(a_1, \ldots, a_{n-1}) = \cdots = f_d(a_1, \ldots, a_{n-1}) = 0, \quad f_0(a_1, \ldots, a_{n-1}) = 1 \ .$$

In addition, we can likewise express the monic polynomial g as

$$g = g_0 + g_1 x_n + \ldots + g_{e-1} x_n^{e-1} + x_n^e$$

with $g_j \in \mathbb{K}[x_1, \ldots, x_{n-1}]$ for $j = 0, \ldots, (e-1)$. Now, consider the *resultant* of f and g with respect to the variable x_n (see [22] for background exposition on resultants). In other words, the resultant R is the polynomial in $\mathbb{K}[x_1, \ldots, x_{n-1}]$ given by the determinant of the following $(e+d) \times (e+d)$ square matrix:

$$R = \left| \begin{array}{cccccccccc}
f_0 & f_1 & \cdots & \cdots & \cdots & f_d & 0 & \cdots & \cdots & 0 \\
0 & f_0 & f_1 & \cdots & \cdots & f_{d-1} & f_d & 0 & \cdots & 0 \\
\vdots & & \ddots & \ddots & & & \ddots & \ddots & & \vdots \\
\vdots & & & \ddots & \ddots & & & \ddots & \ddots & 0 \\
0 & \cdots & \cdots & 0 & f_0 & f_1 & \cdots & \cdots & f_{d-1} & f_d \\
g_0 & g_1 & \cdots & \cdots & g_{e-1} & 1 & 0 & \cdots & \cdots & 0 \\
0 & g_0 & g_1 & \cdots & \cdots & g_{e-1} & 1 & 0 & \cdots & 0 \\
\vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\
\vdots & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\
0 & \cdots & \cdots & \cdots & 0 & g_0 & g_1 & \cdots & g_{e-1} & 1
\end{array} \right| \begin{array}{l} \\ \\ \\ \left. \begin{array}{l} \\ \\ \end{array} \right\} e \text{ rows} \\ \left. \begin{array}{l} \\ \\ \\ \end{array} \right\} d \text{ rows} \\ \\ \\ \end{array}$$

It is easy to see that $R \in I$ by adding the second column times x_n to the first column, and then the third column times x_n^2 to the first column, etc, repeated x_n^{d+e-1} times. Since we have performed only column operations, the value of the determinant is

unchanged, but now we can see that when the determinant is expanded around the first column, the resulting polynomial is a linear combination of f and g, which are both in I. Thus, $R \in I$. Furthermore, $R \in \mathbb{K}[x_1, \ldots, x_{n-1}]$, and therefore R is also in I' (since I' was defined to contain all polynomials in I that do not contain the variable x_n). But $R(a_1, \ldots, a_{n-1})$ is equivalent to evaluating all of the polynomials at $(a_1, \ldots, a_{n-1})$ and then taking the determinant, and that yields a lower triangular matrix, whose diagonal values are all 1. Therefore,

$$R(a_1, \ldots, a_{n-1}) = 1$$

But, according to our induction hypothesis, all polynomials in I' vanish on the point $(a_1, \ldots, a_n)$. Thus, $R \notin I'$, and we have reached a contradiction. Therefore, there cannot exist a polynomial $f \in J$ that is identically 1. Therefore, $J = \{f(a_1, \ldots, a_n, x_n) | f \in I\}$ is a proper ideal of $\mathbb{K}[x_n]$.

But since J is a proper ideal of $\mathbb{K}[x_n]$, it is generated by a single nonconstant polynomial $h(x_n)$ or $h = 0$. In the former case, since $\mathbb{K}$ is algebraically closed, $h(x_n)$ has a single root a_n. In either case, $f(a_1, \ldots, a_{n-1}, a_n) = 0$ for all $f \in I$. Thus, for any proper ideal $I \in \mathbb{K}[x_1, \ldots, x_n]$, there exists a point $(a_1, \ldots, a_n)$ such that $f(a_1, \ldots, a_n)$ for all $f \in I$. This concludes the proof of Hilbert's weak Nullstellensatz. □

It is easy to translate Hilbert's Nullstellensatz into a form more accessible for computation. Consider the following corollary:

Corollary 3.1.3 *Let $\mathbb{K}$ be an algebraically-closed field, and let $f_1, \ldots, f_s$ be polynomials in $\mathbb{K}[x_1, \ldots, x_n]$. Then the system of equations $f_1 = f_2 = \cdots = f_s = 0$ has* no *solution if and*

only if there exist polynomials $\beta_1, \ldots, \beta_s \in \mathbb{K}[x_1, \ldots, x_n]$ *such that*

$$1 = \sum \beta_i f_i \ . \tag{3.1}$$

Proof: Let $I = \langle f_1, \ldots, f_s \rangle$. By Hilbert's Nullstellensatz, if the ideal I is *proper*, there exists a point $(a_1, \ldots, a_n)$ in $\mathbb{K}^n$ such that $f(a_1, \ldots, a_n) = 0$ for all $f \in I$. In other words, the system of polynomial equations $f_1 = \cdots = f_s = 0$ has a solution. If the ideal I is *not* proper, then *no* such point exists, and the system of polynomial equations has *no* solution. In this case, by the definition of an improper ideal, there exist polynomials $\beta_1, \ldots, \beta_s \in \mathbb{K}[x_1, \ldots, x_n]$ such that $1 = \sum \beta_i f_i$. □

We refer to Eq. 3.1 as a *Nullstellensatz certificate*, because it is a certificate that the system of polynomial equations $f_1 = \cdots = f_s = 0$ is *infeasible*.

3.2 NulLA: Examples, Pseudocode and Running Time

Hilbert's Nullstellensatz states that a system of polynomial equations $f_1(x) = \cdots = f_s(x) = 0$, where $f_i \in \mathbb{K}[x_1, \ldots, x_n]$ and $\mathbb{K}$ is an algebraically closed field, has *no* solution in $\mathbb{K}^n$ if and only if there exist polynomials $\beta_1, \ldots, \beta_s \in \mathbb{K}[x_1, \ldots, x_n]$ such that $1 = \sum_{i=1}^{s} \beta_i f_i$ [13]. The polynomial identity $1 = \sum_{i=1}^{s} \beta_i f_i$ is called a *Nullstellensatz certificate*, and we say that a given certificate has degree d if $\max_{1 \leq i \leq s}\{\deg(\beta_i)\} = d$.

The Nullstellensatz Linear Algebra (**N**ul**LA**) algorithm takes as *input* a system of polynomial equations and produces as *output* either **yes**, if the system of polynomial equations has a solution, or **no**, along with a Nullstellensatz certificate of infeasibility, if the system has *no* solution. Before stating the algorithm in pseudocode, we clarify the

connection between certificates and linear algebra computations. Suppose that an input system of polynomial equations has no solution over $\mathbb{K}$, and suppose further that an oracle has told us the degree of the corresponding Nullstellensatz certificate. Thus, we know the polynomial identity $1 = \sum_{i=1}^{s} \beta_i f_i$ exists and has degree d, but we do not know the structure and form of the individual β_i. If we expand the certificate into monomials, the coefficients of a given monomial are linear expressions in the coefficients of the β_i. Since two polynomials over a field are identical precisely when the coefficients of corresponding monomials are identical, from the existence of the degree d certificate $1 = \sum \beta_i f_i$, we can extract a system of *linear* equations whose variables are the coefficients of the β_i. Here is an example:

Example 3.2.1 Consider the input system of polynomial equations $x_1^2 - 1 = 0, x_1 + x_2 = 0, x_1 + x_3 = 0, x_2 + x_3 = 0$. This system has *no* solution. Therefore, it has an associated Nullstellensatz certificate. We begin by assuming the certificate has degree one (the smallest non-trivial degree), and we construct the most generalized Nullstellensatz certificate of degree one possible, with unknowns for coefficients.

$$1 = \underbrace{(c_0x_1 + c_1x_2 + c_2x_3 + c_3)}_{\beta_1}\underbrace{(x_1^2 - 1)}_{f_1} + \underbrace{(c_4x_1 + c_5x_2 + c_6x_3 + c_7)}_{\beta_2}\underbrace{(x_1 + x_2)}_{f_2}$$
$$+ \underbrace{(c_8x_1 + c_9x_2 + c_{10}x_3 + c_{11})}_{\beta_3}\underbrace{(x_1 + x_3)}_{f_3} + \underbrace{(c_{12}x_1 + c_{13}x_2 + c_{14}x_3 + c_{15})}_{\beta_4}\underbrace{(x_2 + x_3)}_{f_4}.$$

Expanding the tentative Nullstellensatz certificate into monomials and grouping like terms, we arrive at the following polynomial equation:

$$1 = c_0x_1^3 + c_1x_1^2x_2 + c_2x_1^2x_3 + (c_3 + c_4 + c_8)x_1^2 - c_3 + (c_{10} + c_{14})x_3^2 + (c_4 + c_5 + c_9 + c_{12})x_1x_2$$
$$+ (c_5 + c_{13})x_2^2 + (c_6 + c_8 + c_{10} + c_{12})x_1x_3 + (c_6 + c_9 + c_{13} + c_{14})x_2x_3 + (c_7 + c_{15} - c_1)x_2$$
$$+ (c_{11} + c_{15} - c_2)x_3 + (c_7 + c_{11} - c_0)x_1 \ .$$

From this, we extract a system of *linear* equations. Since a Nullstellensatz certificate is identically one, all monomials except the constant term must be equal to zero; namely:

$$c_0 = 0 \ , \ c_1 = 0 \ , \ c_2 = 0 \ , \ c_3 + c_4 + c_8 = 0 \ , \ \ldots \ , c_7 + c_{11} - c_0 = 0 \ , \ -c_3 = 1 \ .$$

After solving this system of linear equations, if it is *consistent*, we can reconstruct the Nullstellensatz certificate from the solution. In this case,

$$1 = \frac{1}{2}x_1(x_1 + x_2) - \frac{1}{2}x_1(x_2 + x_3) + \frac{1}{2}x_1(x_1 + x_3) - (x_1^2 - 1) \ .$$

□

In general, the degree of a Nullstellensatz certificate will not be known in advance. Our approach is to start with the lowest possible degree (almost always one), and construct the corresponding linear system. If the linear system has a solution, then the solution corresponds directly to a Nullstellensatz certificate, and we have found a *proof* that the original input system of polynomial equations does *not* have a solution (or a common root). Otherwise, we continue to increment the degree, and construct and solve the corresponding linear systems, until we have either found a *consistent* linear system (and thus a Nullstellensatz certificate), or tested enough degrees that we can be certain no Nullstellensatz certificate exists. The number of iterations of this incremental approach determines the running time of **N**ul**LA**.

There are well-known upper bounds on the degrees of Nullstellensatz certificates. For example, consider the following:

Lemma 3.2.2 (Kollár [30]) *Given polynomials* $f_1, \ldots, f_s \in \mathbb{K}[x_1, \ldots, x_n]$ *where* $\mathbb{K}$ *is an algebraically-closed field and* $d = \max\{\deg(f_i)\}$*, if* $f_1, \ldots, f_s$ *have no common zeros, then*

$1 = \sum_{i=1}^{s} \beta_i f_i$ *where*

$$\deg(\beta_i) \leq \max\{3, d\}^n \ .$$

Proof: This follows directly from Definitions 1.3 and 1.4 and Theorem 1.5 of [30]. □

Beyond the very general (and sharp) bounds of Kollár for the Nullstellensatz, we can still hope for less extreme (e.g., subexponential) bounds for our combinatorial ideals. Indeed, we are in luck: The solution of *linear* systems of equations with *polynomial* coefficients is an important topic that has received attention, both by algebraic geometers as well as computer algebraists, and we can profit here from a fundamental result by D. Lazard [35] that provides ideals like ours with a *linear* bound.

Lemma 3.2.3 (Lazard [35]) *Let $f_1, \ldots, f_k$ be homogeneous polynomials of $\mathbb{K}[x_0, \ldots, x_n]$ that generate an ideal I, let d_i be the degree of f_i and assume that $d_1 \geq d_2 \geq \cdots \geq d_k \geq 1$ and $k \geq n+1$. Then the following conditions are equivalent:*

1) *The k projective hypersurfaces defined by $f_1, \ldots, f_k$ have no point in common over the algebraic closure of $\mathbb{K}$ (in particular, they have no point in common at infinity).*

2) *The ideal I contains a power of the maximal ideal $M = \langle x_0, x_1, \ldots, x_n \rangle$; namely, for some power p, $x_i^p \in I$ for all x_i.*

3) *$M^p \subset I$ with $p = d_1 + d_2 + \cdots + d_{n+1} - n \leq (n+1)(\max_{1 \leq i \leq n+1}\{d_i\} - 1) + 1$.*

4) *The map $\phi : (\beta_1, \ldots, \beta_k) \to \sum \beta_i f_i$ is surjective among all polynomials of degree p, when, for all i, β_i is a homogeneous polynomial of degree $p - d_i$.*

The proof of Lemma 3.2.3 relies on advanced techniques in commutative and homological algebra, and is presented in [35], pg. 169. As a consequence of Lemma 3.2.3, when given polynomials $f_i \in \mathbb{K}[x_1, \ldots, x_n]$, we can consider their homogenization $\bar{f}_i$, using an extra variable x_0 (e.g., $x^2 - x$ can be homogenized to $x^2 - xx_0$). If we are able to find a "projective" Nullstellensatz of the form

$$x_0^p = \sum \beta_i \bar{f}_i \ ,$$

then we can substitute $x_0 = 1$ in the above equation and obtain the form of the Nullstellensatz that is more desirable for computation (e.g., $1 = \sum \beta_i' f_i$). Furthermore, the degree of β_i' is less than or equal to the degree of β_i.

We can summarize the Lazard lemma as follows (see Brownawell [7]):

Corollary 3.2.4 *Given polynomials $f_1, \ldots, f_s \in \mathbb{K}[x_1, \ldots, x_n]$ where $\mathbb{K}$ is an algebraically-closed field and $d = \max\{\deg(f_i)\}$, if $f_1, \ldots, f_s$ have no common zeros and $f_1, \ldots, f_s$ have no common zeros at infinity, then $1 = \sum_{i=1}^{s} \beta_i f_i$ where*

$$\deg(\beta_i) \leq n(d-1) \ .$$

Therefore, the bound on the Nullstellensatz obtained for the combinatorial ideals described by Lemmas 2.2.2 and 2.1.1 (the coloring and independent set ideals, respectively) is $2n$ and n, respectively. In other words, our combinatorial ideals have a *linear* bound, which is a considerable improvement on the exponential bound predicted by Kollár.

Although this linear bound is an improvement, it is still far from being computationally practical. However, we have observed in practice that the degree growth of polynomial systems for combinatorial problems is often *very* slow — slow enough to deal with large graphs or other combinatorial structures.

We conclude by presenting **Nul**$\mathbf{LA}$ in pseudocode:

****

ALGORITHM: **Null**stellensatz **L**inear **A**lgebra Algorithm (**Nul**$\mathbf{LA}$)
INPUT: A system of polynomial equations $F = \{f_1(x) = 0, \ldots, f_s(x) = 0\}$
OUTPUT: YES, if F has solution, else NO, along with a Nullstellensatz certificate of infeasibility.

$d \leftarrow 1$.
$K \leftarrow$ known upper bounds on degree of Nullstellensatz for F (see [30], [7], [35])
while $d \leq K$ **do**
  CERT $\leftarrow \sum_{i=1}^{s} \beta_i f_i$ (where β_i are polynomials of degree d, with unknowns for their coefficients).
  Extract a system of linear equations from CERT with columns corresponding to unknowns, and rows corresponding to monomials.
  Solve the linear system.
  if the linear system is consistent **then**
    CERT $\leftarrow \sum_{i=1}^{s} \beta_i f_i$ (with unknowns in β_i replaced with linear system solution values.)
    print "The system of equations F is infeasible."
    return NO, along with CERT.
  end if
  $d \leftarrow d + 1$.
end while
print "The system of equations F is feasible."
return YES.

****

Chapter 4

Theoretical Complexity of NulLA

Into the Valley of Death
Rode the six hundred.

- Alfred, Lord Tennyson
Charge of the Light Brigade

4.1 P, NP and the Nullstellensatz

There is a fascinating connection between Hilbert's Nullstellensatz and computational complexity. If the system of polynomial equations given as input to **N**ul**LA** encodes an NP-complete problem, the output certificate of infeasibility is a witness certificate for the *complement* of the NP-complete problem. For example, when the input system of polynomial equations encodes the independent set problem, the output Nullstellensatz certificate demonstrates that *no* independent set of size k exists. When the input system of polynomial equations encodes graph k-colorability, the output certificate demonstrates that the underlying graph is non-k-colorable. Since the belief in certain containments within complexity classes such as P and NP is so ubiquitous as to be considered fact (in addition to being

supported by decades of computational evidence), it is logical to characterize growth in the degree of Nullstellensatz certificates in terms of these expected containments.

In this section, we begin by characterizing Nullstellensatz certificate degree growth in terms of P $\neq$ NP, then we discuss the growth in terms of the proper containment of NP within EXPTIME, and finally, we conclude with observations on the *density* of the certificates and the relationship between NP and coNP. Throughout this section, we rely heavily on definitions 2.0.1 and 2.0.2 from Chapter 2.

In terms of the notation regarding $O(g(n))$-encodings, recall that $O(n^k)$ can be bounded above by $O(n^{k+d})$ where d is any non-negative integer. In the proofs that follow, we will often use the largest order of growth to bound every quantity in the system of polynomial equations. For example, if an encoding has $O(n)$ variables, but $O(n^2)$ equations, we will refer to the encoding as an $O(n^2)$-encoding, and treat the system of equations as though it has $O(n^2)$ variables. This is obviously true, even if a large overestimate.

Lemma 4.1.1 *Let L be an NP-complete language and $g(n)$ a polynomial such that L has a $O(g(n))$-encoding. Then, if $P \neq NP$, $\forall d \in \mathbb{Z}_{\geq 0}$, there must exist an instance I of the language L with $I \notin L$ such that the minimum-degree of the associated Nullstellensatz certificate is strictly greater than d.*

Proof: Our proof is by contradiction with the hypothesis P $\neq$ NP. Assume that, $\forall\ I \notin L$, the minimum-degree of the associated Nullstellensatz certificate has $\deg(\beta_i) < d$ for some constant d. We will show that P = NP by demonstrating a polynomial-time algorithm for deciding L: (1) Given an instance I of L, encode it (using the given $O(g(n))$-encoding) as the system of equations $f_1 = \cdots = f_s = 0$, (2) Construct and solve the **NulLA** linear system

obtained by assuming the degree of the certificate is d, (3) If the system has a solution, a Nullstellensatz certificate exists, and $I \notin L$: return **no**, (4) If the system does *not* have a solution, then there does *not* exist a Nullstellensatz certificate, and $I \in L$: return **yes**.

Now we analyze the running time of this algorithm. In Step 1, since L has an $O(g(n))$-encoding, we encode I in $O(g(n))$ time.

For Step 2, we note that by Corollary 3.2b of [56], if a system of linear equations $Ax = b$ has a solution, then it has a solution polynomially-bounded by the bit-sizes of the matrix A and the vector b (see [56] for a definition of bit-size). In this case, the vector b contains only zeros and ones. To calculate the bit-size of A, we recall our assumption that, for every β_i, $\deg(\beta_i) < d$ for some constant d. Therefore, an upper bound on the number of terms in each β_i is the total number of monomials in $O(g(n))$ variables of degree less than or equal to d. Therefore, the number of terms in each β_i is

$$\binom{O(g(n))+d-1}{O(g(n))-1} + \binom{O(g(n))+d-2}{O(g(n))-1} + \cdots + \binom{O(g(n))-1}{O(g(n))-1}$$

$$= O\Big(\big(O(g(n))\big)^d\Big) + O\Big(\big(O(g(n))\big)^{d-1}\Big) + \cdots + O(1) = O\Big(n^{d\cdot\deg(g)}\Big) .$$

Because there are $O(g(n))$ equations, there are $O(g(n)) \cdot O\Big(n^{d\cdot\deg(g)}\Big) = O\Big(n^{(d+1)\deg(g)}\Big)$ unknowns in the linear system, and thus, $O\Big(n^{(d+1)\deg(g)}\Big)$ columns in A. Because each of the $O(g(n))$ equations has $O(g(n))$ terms, there are $O(g(n)) \cdot O(g(n)) \cdot O\Big(n^{d\cdot\deg(g)}\Big) = O\Big(n^{(d+2)\deg(g)}\Big)$ terms in the *expanded* Nullstellensatz certificate, and thus $O\Big(n^{(d+2)\deg(g)}\Big)$ rows in A. Furthermore, since the coefficient bit-size in every equation is $O(g(n))$, the matrix A contains only entries of bit-size $O(g(n))$. Therefore, the bit-sizes of both A and b are polynomially-bounded in n, and by Theorem 3.3 of [56], the linear system can be solved

in polynomial-time.

Therefore, we have demonstrated a polynomial-time algorithm for deciding the language L, and because L is NP-complete, this implies P = NP, which contradicts our hypothesis. Therefore, $\forall d \in \mathbb{Z}_{\geq 0}$, there must exist an instance I of L with $I \notin L$ such that the minimum-degree of the associated Nullstellensatz certificate is strictly greater than d.□

Lemma 4.1.1 and our encodings from Section 2 immediately give rise to the following corollaries for the NP-complete problems of independent set, graph k-colorability and planar graph k-colorability.

Corollary 4.1.2 *If $P \neq NP$, then there must exist an infinite family of non-k-colorable graphs such that, $\forall d \in \mathbb{Z}_{\geq 0}$, there is a graph in the family where the minimum-degree of the associated Lemma 2.2.2 Nullstellensatz certificate has degree strictly greater than d.*

Proof: Graph k-coloring is NP-complete [21] for $k \geq 3$, and is encoded in Lemma 2.2.2 by the following system of polynomial equations: $x_i^k - 1 = 0$ for $i \in V(G)$, and $\sum_{l=0}^{k-1} x_i^l x_j^{k-1-\ l} = 0$ for $\{i, j\} \in E(G)$. Since the vertex polynomials $x_i^k - 1$ have two terms, and the edge polynomials $\sum_{l=0}^{k-1} x_i^l x_j^{k-1-\ l}$ have k terms (with $k \leq n$), each equation has $O(n)$ terms. Furthermore, the coefficients within the equations are ± 1, and the degree is at most k. Since there are $O(n^2)$ edges in a graph, there are $O(n + n^2) = O(n^2)$ equations in the system. Therefore, this is an $O(n^2)$-encoding and, by Lemma 4.1.1, $\forall d \in \mathbb{Z}_{\geq 0}$, there exists a non-$k$-colorable graph such that the minimum-degree of the associated Nullstellensatz certificate is strictly greater than d. This defines the infinite family of graphs. □

Corollary 4.1.3 *If $P \neq NP$, then there must exist an infinite family of non-k-colorable* planar *graphs such that, $\forall d \in \mathbb{Z}_{\geq 0}$, there is a graph in the family where the minimum-degree of the associated Lemma 2.2.2 Nullstellensatz certificate has degree strictly greater than d.*

Proof: Planar graph k-coloring is also NP-complete [21] for $k \geq 3$. Thus, the proof follows from the proof of Corollary 4.1.2. □

Corollary 4.1.4 *If $P \neq NP$, then there must exist an infinite family of graphs such that, $\forall d \in \mathbb{Z}_{\geq 0}$, there is a graph in the family where the minimum-degree of the associated Lemma 2.1.1 Nullstellensatz certificate proving the non-existence of an independent set of size greater than $\alpha(G)$ has degree strictly greater than d.*

Proof: The independent set decision problem is NP-complete [21], and is encoded by the following system of polynomial equations: $x_i^2 - x_i = 0$ for $i \in V(G)$, $x_i x_j = 0$ for $\{i, j\} \in E(G)$, and $-(\alpha(G) + r) + \sum_{i=1}^{n} x_i = 0$ (Lemma 2.1.1). Every equation has $O(n)$ terms, and since there are at most $O(n^2)$ edges in a graph, there are at most $O(n^2)$ equations in the system. Furthermore, the coefficients within the equations are ± 1, and the degree is at most two. Therefore, Lemma 2.1.1 provides an $O(n^2)$-encoding, and the existence of an infinite family of graphs follows from Lemma 4.1.1. □

In Lemma 4.1.1, we describe growth in the minimum-degree of a Nullstellensatz certificate in terms of the well-respected belief that P $\neq$ NP. However, we did not characterize the growth as logarithmic, linear or exponential, beyond the observation that the minimum-degree cannot be bounded from above by a constant. It is logical to expect that different *rates* of growth might again be related to certain containments within the tower

of complexity class inclusions. From complexity theory, we know the following:

$$\mathrm{P} \subseteq \mathrm{NP} \subseteq \mathrm{PSPACE} \subseteq \mathrm{EXPTIME}\ .$$

It is known via the time hierarchy theorem [52] that P is a *proper* subset of EXPTIME. Thus, at least one of these containments must be proper, but it is not known which one; indeed, it is widely believed that *all* inclusions are proper [52]. In the following corollary, we observe that if the minimum-degree grows *logarithmically*, then NP is a *proper* subset of EXPTIME, which is widely believed to be true.

Lemma 4.1.5 *Let L be an NP-complete language and $g(n)$ a polynomial such that L has an $O(g(n))$-encoding. If the degree of a minimum-degree Nullstellensatz certificate is $O(\log(n))$, then $NP \subsetneq EXPTIME$.*

Proof: By the algorithm described in the proof of Lemma 4.1.1, we know that the time required to find a Nullstellensatz certificate is at most the time required to solve the associated linear system. We note that, as in Lemma 4.1.1, because the bit-size of every entry in the matrix has bit-size $O(g(n))$, the solution will be polynomially-bounded in the bit-size of the matrix. Thus, within a polynomial factor, the time required to solve a linear system is dependent on the size of the linear system, and the size of the linear system is dependent on the minimum-degree of the Nullstellensatz certificate. Since our encoding is an $O(g(n))$-encoding, if the minimum-degree is $O(\log(n))$, the number of rows and columns in the linear system is given by the number of monomials per equation times the number

of equations, as shown in Lemma 4.1.1. Thus, we see that the size of the linear system is

$$\underbrace{\Big(O\Big(g(n)^{\log(n)}\Big)\cdot O(g(n))\Big)}_{\text{rows}} \times \underbrace{\Big(O\Big(g(n)^{\log(n)}\Big)\cdot O(g(n))\Big)}_{\text{columns}} ,$$

$$\underbrace{O\Big(n^{\deg(g)(\log(n)+1)}\Big)}_{\text{rows}} \times \underbrace{O\Big(n^{\deg(g)(\log(n)+1)}\Big)}_{\text{columns}} .$$

For an $N \times N$ matrix, Gaussian Elimination is $O(N^3)$ (possibly with another polynomial factor for the arithmetic). Thus, the time required to find the Nullstellensatz certificate is

$$\begin{aligned} O\Big(\big(n^{\deg(g)(\log(n)+1)}\big)^3\Big) &= O\Big(n^{3\deg(g)(\log(n)+1)}\Big) \\ &= O\Big(n^{3\deg(g)(\log(n)+\log(n))}\Big) = O\Big(n^{6\deg(g)\log(n)}\Big) \\ &= O\Big(\big(2^{\log(n)}\big)^{6\deg(g)\log(n)}\Big) = O\Big(2^{6\deg(g)\log^2(n)}\Big) \\ &= O\Big(2^{\log^2(n)}\Big) = O\Big(n^{\log(n)}\Big) . \end{aligned}$$

Recall the definition:

$$\text{EXPTIME} = \bigcup_{k\in\mathbb{N}} \text{DTIME}\big(2^{n^k}\big) .$$

Thus, the running time required to find a Nullstellensatz certificate is *superpolynomial but subexponential*. Since L is an NP-complete problem, all other problems in NP are polynomially-reducible to L. Therefore, there exists a *superpolynomial but subexponential* time algorithm for *every* problem in NP. Therefore, NP $\subsetneq$ EXPTIME. □

To summarize Lemma 4.1.5, we note that if the minimum-degree of Nullstellensatz certificates associated with an NP-complete problem grow *linearly* with respect to input size, then the running time of **N**ul**LA** is $O(n^n) = O\big(2^{n\log(n)}\big)$: exponential. In other words, if

we demonstrate that the minimum-degree of Nullstellensatz certificates associated with an NP-complete grow *linearly* in the worst case, then we have simply demonstrated yet another exponential-time algorithm for an NP-complete problem. However, if the minimum-degree of Nullstellensatz certificates associated with an NP-complete problem grow *logarithmically* with respect to input size, then the running time of **N**ul**LA** is $O\big(n^{\log(n)}\big) = O\big(2^{\log^2(n)}\big)$: superpolynomial but subexponential. In other words, demonstrating logarithmic growth in the minimum-degree of Nullstellensatz certificates associated with an NP-complete problem in the worst case would be a new result; we would have described a superpolynomial but subexponential time algorithm for an NP-complete problem, and unexpectedly proved that NP is a proper subset of EXPTIME. The curious point is that the rate of growth in the minimum-degree of Nullstellensatz certificates is a fixed, algebraic property of the encoding; we can at times slow down the rate of growth (see Section 5.1.2), but in general, the rate of growth is simply a property of the encoding. Thus, there is hope that if a particular encoding captures the combinatorial properties of an NP-complete problem elegantly enough with respect to the Nullstellensatz, then the rate of growth might be such that a new complexity result simply reduces to accurately describing an algebraic property of a system of polynomial equations.

Thus far in this section, we have characterized growth in the degree of the Nullstellensatz in terms of the commonly-held view of P and NP. However, as observed earlier, when a system of equations encodes an NP-complete problem, the resulting Nullstellensatz certificate is a certificate for the *complement* of the NP-complete problem, or a witness certificate for a problem in coNP. (Recall that a problem is in coNP if its complement is in

NP). Thus, it is also logical to characterize Nullstellensatz certificates in terms of the relationship between NP and coNP. In other words, we are no longer concerned with *finding* a Nullstellensatz certificate; we are only concerned with *verifying* a Nullstellensatz certificate. These certificates are, after all, polynomial identities that simplify to one. Thus, given a Nullstellensatz certificate, not only must its degree grow with respect to the input size, but the number of multiplications/additions/monomial comparisons required to simplify the certificate must be large; in other words, the certificates must be *dense* with respect to the number of monomials contained in the coefficients.

The following lemma was first proposed by Lovász in [41] in terms of specifically-defined graph identities: we generalize it here to any NP-complete problem with an $O(g(n))$-encoding.

Lemma 4.1.6 *Let L be an NP-complete language and $g(n)$ a polynomial such that L has an $O(g(n))$-encoding. Furthermore, let every instance I of L with $I \notin L$ have an associated Nullstellensatz certificate of the form*

$$1 = \sum_{i=1}^{s} \beta_i f_i \ , \tag{4.1}$$

where the bit-size of $\max\{\deg(\beta_1), \ldots, \deg(\beta_s)\}$ *is $O(g(n))$. Then, if NP $\neq$ coNP, there must exist an infinite family of instances I of L with $I \notin L$ such that the associated Nullstellensatz certificates contain at least one β_i with a superpolynomial number of terms.*

Proof: Our proof is by contradiction with the hypothesis NP $\neq$ coNP. Consider an instance I of L with $I \notin L$, and let Eq. 4.1 be its associated Nullstellensatz certificate. Let $h(n)$ be a polynomial, and assume that the number of terms in any β_i is $O(h(n))$. Without loss

of generality, assume $h(n) = O(g(n))$; otherwise, $g(n) = O(h(n))$, and we can simply use $O(h(n))$ to bound the encoding. We will show that NP = coNP by showing that Eq. 4.1 can be *simplified* or *verified* in polynomial-time.

Consider the number of operations required to simplify the certificate. We must expand each individual $\beta_i f_i$, sort the resulting monomials and then add/subtract the coefficients. This is a lower bound on the number of operations required to simplify the certificate, since we must determine the final coefficient of each monomial produced by the product $\beta_i f_i$ in the expanded certificate. Since $f_1 = \cdots = f_s = 0$ is an $O(g(n))$-encoding, every f_i contains $O(g(n))$ terms, and by assumption, every β_i contains $h(n) = O(g(n))$ terms. Thus, the product $\beta_i f_i$ requires $O(g(n)) \cdot O(g(n)) = O\Big(n^{2\deg(g)}\Big)$ multiplications. Since the number of equations is $O(g(n))$, the total number of multiplications required to expand the entire certificate is $O(g(n)) \cdot O\Big(n^{2\deg(g)}\Big) = O\Big(n^{3\deg(g)}\Big)$. Since the bit-size of every coefficient in any f_i is $O(g(n))$, and because the coefficients in any β_i correspond to the solution of a polynomially-bounded linear system (as shown in the proof of Lemma 4.1.1), each of those multiplications can be performed in polynomial-time.

Next, we sort the terms appearing in the expanded certificate, group the similar monomials together, and simplify the coefficients. In any given monomial, the number of variables is $O(g(n))$ and the bit-size of the degree is $O(g(n))$. Thus, comparing two monomials takes polynomial time. Since the number of terms in the expanded certificate is $O\Big(n^{3\deg(g)}\Big)$, sorting the list also takes polynomial time. Finally, since the bit-size of the coefficients is polynomial (either $O(g(n))$ or polynomial-bounded via the linear system solution), calculating the coefficient of any given monomial takes polynomial-time.

Thus, we have shown that a Nullstellensatz certificate such as Eq. 4.1, where $f_1 = \cdots f_s = 0$ represents an $O(g(n))$-encoding of L, and where each β_i has $O(g(n))$ terms, can be simplified or *verified* in polynomial time. But recall the following well-known result [21], where $\overline{L}$ denotes the complement of L:

Let L be any NP-complete language. If $\overline{L}$ is in NP, then NP = coNP.

By showing that a Nullstellensatz certificate like Eq. 4.1 can be *simplified* in polynomial time, we have shown that an instance I of L with $I \notin L$ can be *verified* in polynomial-time. In other words, we have demonstrated that $\overline{L}$ is in NP. Thus, NP = coNP, which contradicts our hypothesis. Therefore, the number of terms in every β_i cannot be $O(h(n))$ for any polynomial $h(n)$: there must exist at least one β_i with a *superpolynomial* number of terms. □

The Schwartz-Zippel lemma (see [51], pg. 165) is a probability result which is commonly used as the basis of a random algorithm for verifying proposed polynomial identities. Suppose we are given a polynomial $f \in \mathbb{K}[x_1, \ldots, x_n]$, and we suspect f is identically zero, but the act of deterministically expanding and simplifying f takes exponential time (in the number of variables). Therefore, we are interested in a method for *randomly* verifying f in polynomial-time (in the number of variables), and determining whether f is identically zero within some user-specified, error bound. Note that, since a Nullstellensatz certificate is identically one, subtracting one from the certificate creates an identically zero polynomial. The Schwartz-Zippel lemma is as follows:

Lemma 4.1.7 (Schwartz-Zippel) *Let $f \in \mathbb{K}[x_1, \ldots, x_n]$ be a non-zero polynomial of degree $d \geq 0$. Let S be a finite subset of $\mathbb{K}$ and let $r_1, \ldots, r_n$ be selected randomly from S.*

Then

$$Pr[f(r_1, r_2, \ldots, r_n) = 0] \leq \frac{d}{|S|} \ .$$

The Schwartz-Zippel random polynomial identity verification algorithm is as follows. We define a "trial" to consist of 1) choosing n values arbitrarily from the set S to create a point $\{r_1, \ldots, r_n\}$, and 2) evaluating $f(r_1, \ldots, r_n)$. If $f(r_1, \ldots, r_n) = 0$, then the trial is considered a *success*. If $f(r_1, \ldots, r_n) \neq 0$, then we can be certain f is not identically zero. Thus, if $|S| \geq 2d$ and we conduct t successful trials, then the probability that f is identically zero is $1 - (1/2)^t$. However, if $f(r_1, \ldots, r_n)$ fails on even one point, then we are certain f is not identically zero. Therefore, the Schwartz-Zippel random polynomial identity verification algorithm is a *no-biased Monte Carlo* algorithm: a "no" answer is always correct, but a "yes" answer may be incorrect, within boundable error probability. Thus, if $f(r_1, \ldots, r_n)$ can be evaluated in polynomial-time, then randomly verifying a polynomial identify to within a very small error bound can be done in polynomial-time. Therefore, Lemma 4.1.6 provides the following corollary:

Corollary 4.1.8 *Let L be an NP-complete language and $g(n)$ a polynomial such that L has an $O(g(n))$-encoding. Furthermore, let every instance I of L with $I \notin L$ have an associated Nullstellensatz certificate of the form*

$$1 = \sum_{i=1}^{s} \beta_i f_i \ ,$$

where the bit-size of $\max\{\deg(\beta_1), \ldots, \deg(\beta_s)\}$ *is $O(g(n))$. Then, if NP $\neq$ coNP, a given trial of the Schwartz-Zippel polynomial identify verification algorithm does not run in polynomial-time in the length of I.*

4.2 Independent Set and the Nullstellensatz

In this section, we explore the Nullstellensatz certificates associated with the independent set encoding described by Lovász in Lemma 2.1.1. Without using any assumptions about complexity theory such as P $\neq$ NP or NP $\neq$ coNP, we prove a clear and direct relationship between the Nullstellensatz certificates and the underlying graphs: the minimum-degree is the *stability number* $\alpha(G)$, or the size of the largest independent set in G, and there is at least one monomial per independent set in G. In this case, not only do we demonstrate *linear* growth in the minimum-degree of the Nullstellensatz certificates, but we also prove that they are *dense*. In other words, we arrive at a result *predicted* by assuming P $\neq$ NP and NP $\neq$ coNP without any assumptions on complexity class containments. In [41], Lovász proposed the challenge of explicitly finding a family of graphs with growth in the minimum-degree of their Nullstellensatz certificates. As an unexpected byproduct of investigating the theoretical complexity of **N**ul**LA**, we answer his open question.

This section is structured as follows. We first introduce the idea of a *reduced* certificate, which we use throughout this section. Next, we prove that for every graph, there *exists* a Nullstellensatz certificate of degree $\alpha(G)$. Then, we prove that $\alpha(G)$ is indeed the minimum degree. As a corollary that arises from the structure of these proofs, we demonstrate that there is one monomial per independent set in G, and conclude with remarks on the computational implications of these results.

Throughout this section, when we say a monomial is *supported on the independent sets of* G, we mean that the variables in the monomial correspond to vertices in the graph that form an independent set. Furthermore, A, Q_i, Q_{ij}, etc. always denote polynomials in

$\mathbb{C}[x_1, \ldots, x_n]$.

Lemma 4.2.1 *For any graph G and a Nullstellensatz certificate*

$$1 = A\Big(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i \Big) + \sum_{i \in V(G)} Q_i(x_i^2 - x_i) + \sum_{\{i,j\} \in E(G)} Q_{ij}(x_i x_j) \tag{4.2}$$

certifying that G has no stable set of size $(\alpha(G)+r)$ (with $r \geq 1$), we can construct a "reduced" Nullstellensatz certificate

$$1 = A'\Big(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i \Big) + \sum_{i \in V(G)} Q'_i(x_i^2 - x_i) + \sum_{\{i,j\} \in E(G)} Q'_{ij}(x_i x_j) \ ,$$

such that

1. *The coefficient A' multiplying $-(\alpha(G)+r) + \sum_{i=1}^{n} x_i$ has only square-free monomials supported on independent sets of G, and thus* $\deg(A') \leq \alpha(G)$.

2. $\max\{deg(A), deg(Q_i), deg(Q_{ij})\} = \max\{deg(A'), deg(Q'_i), deg(Q'_{ij})\}$. *Thus, if the original Nullstellensatz certificate has minimum-degree, the "reduced" certificate also has minimum-degree.*

Proof: Let I be the ideal generated by $x_i^2 - x_i$ (for every node $i \in V(G)$), and $x_i x_j$ (for every edge $\{i,j\} \in E(G)$). Furthermore, let B equal

$$1 = \underbrace{A\Big(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i \Big)}_{B} + \sum_{i \in V(G)} Q_i(x_i^2 - x_i) + \sum_{\{i,j\} \in E(G)} Q_{ij}(x_i x_j) \ ,$$

We apply reductions modulo I to Eq. 4.2. If a non-square-free monomial appears in polynomial A, say $x_{i_1}^{\alpha_1} x_{i_2}^{\alpha_2} \cdots x_{i_k}^{\alpha_k}$ with at least one $\alpha_j > 1$, then we can subtract the polynomial $x_{i_1}^{\alpha_1} x_{i_2}^{\alpha_2} \cdots, x_{i_j}^{\alpha_j - 2} x_{i_k}^{\alpha_k} B(x_{i_j}^2 - x_{i_j})$ from AB and simultaneously add it to $\sum_{i=1}^{n} Q_i(x_i^2 - x_i)$. Thus, eventually we obtain a new certificate that has only square-free monomials in A'.

Furthermore, if Q'_{i_j} has new monomials, they are of degree less than or equal to what was originally in A.

Similarly, if $x_{i_1}x_{i_2}\cdots x_{i_k}$ appears in A, but $x_{i_1}x_{i_2}\cdots x_{i_k}$ contains an edge $\{i,j\} \in E(G)$ (if x_ix_j divides $x_{i_1}x_{i_2}\cdots x_{i_k}$), then we can again subtract $B(x_{i_1}x_{i_2}\cdots x_{i_k}/x_ix_j)(x_ix_j)$ from AB, and, at the same time, add it to $\sum_{\{i,j\}\in E(G)} Q_{ij}x_ix_j$. Furthermore, the degree is maintained, and we have reached the form we claim exists for A'. □

We now show that, for *every* graph, there exists an explicit Nullstellensatz certificate of degree $\alpha(G)$. In order to prove this claim, we introduce the following notation. Let S_i be the set of all independent sets of size i in G. For any independent set $I \in S_i$, if I consists of the vertices $\{c_1, c_2, \ldots, c_i\}$, then $x_I := x_{c_1}x_{c_2}\cdots x_{c_i}$, and we refer to the monomial x_I as a "independent set". We define $S_0 := \emptyset$, and $x_\emptyset = 1$. If we say $I \cup k \in S_{i+1}$, we explicitly mean that $I \cap k = \emptyset$, and that x_Ix_k is a square-free independent set monomial of degree $i+1$. If $I \cup k \notin S_{i+1}$, we explicitly mean that $I \cap k = \emptyset$ but $I \cup k$ contains at least one edge $\{k, c_j\}$. In other words, x_Ix_k is a square-free non-independent set monomial of degree $i+1$. In this case, let $\min_k(I)$ denote the *smallest* $c_j \in I$ such that $\{k, c_j\} \in E(G)$. Finally, let

$$P_i := \sum_{I\in S_i} x_I \ , \quad \text{with} \quad P_0 := 1 \ , \quad \text{and} \quad L_i := \frac{iL_{i-1}}{\alpha(G)+r-i} \ , \quad \text{with} \quad L_0 := \frac{1}{\alpha(G)+r} \ .$$

Theorem 4.2.2 *Given a graph G, there exists a Nullstellensatz certificate of degree $\alpha(G)$ certifying the non-existence of an independent set of size $\alpha(G)+r$ (for $r \geq 1$) such that*

$$1 = A\Big(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i\Big) + \sum_{\{u,v\}\in E(G)} Q_{uv}x_ux_v + \sum_{k=1}^{n} Q_k(x_k^2 - x_k) \ , \tag{4.3}$$

where

$$A = -\sum_{i=0}^{\alpha(G)} L_i P_i \ , \qquad Q_{uv} = \sum_{i=1}^{\alpha(G)} \left(\sum_{\substack{I \in S_i : I \cup v \notin S_{i+1} \text{ and} \\ \min_v(I) = u}} L_{i+1} x_{I \setminus u} \right) \quad \textit{and}$$

$$Q_k = \sum_{i=0}^{\alpha(G)} \left(\sum_{I \in S_i : I \cup k \in S_{i+1}} L_{i+1} x_I \right) .$$

Proof: Our proof is the direct verification of Eq. 4.3. Let B, C and D equal

$$1 = A \underbrace{\left(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i \right)}_{B} + \underbrace{\sum_{\{u,v\} \in E(G)} Q_{uv} x_u x_v}_{C} + \underbrace{\sum_{k=1}^{n} Q_k (x_k^2 - x_k)}_{D} \ ,$$

It is easy to see that

$$-L_0 P_0 \big(-(\alpha(G)+r) \big) = -\frac{1}{\alpha(G)+r} \left(-(\alpha(G)+r) \right) = 1 \ .$$

We will now show that the coefficient for every other monomial in Eq. 4.3 simplifies to zero. We begin by observing that every monomial in A, Q_k or Q_{uv} is an independent set, and furthermore, that the independent set monomials in Q_k do not contain the variable x_k, and the independent set monomials in Q_{uv} contain neither x_u nor x_v. Therefore, in the expanded certificate $AB + C + D$, only three types of monomials appear: square-free independent set monomials, square-free non-independent set monomials, and independent set monomials with exactly one variable squared.

- **square-free independent set:** Let $I = \{c_1, c_2, \ldots, c_m\}$ be any independent set of size m. The monomial x_I is created in AB in two ways: $x_{I \setminus c_k} x_{c_k}$ (formed m times, one for each c_k), or $x_I \big(-(\alpha(G)+r) \big)$. Thus, the coefficient for x_I in AB is

$$-mL_{m-1} - L_m \big(-(\alpha(G)+r) \big) = -m \frac{L_m(\alpha(G)+r-m)}{m} + L_m(\alpha(G)+r) = mL_m \ .$$

The monomial x_I does not appear in C, because x_I is an independent set monomial. However, the monomial x_I is produced by $x_{I\setminus c_k}(-x_{c_k})$ in D (formed m times, one for each c_k), and the coefficient for x_I in D is $-mL_m$. Therefore, we see that

$$\underbrace{mL_m}_{\text{from } AB}\ \underbrace{-mL_m}_{\text{from } D} = 0 \ .$$

- **square-free non-independent set:** Let $I = \{c_1, c_2, \ldots, c_{m-1}, u\}$ be any independent set of size m, and consider the monomial $x_I x_v$ where $u = \min_v I$ and $\{u, v\} \in E(G)$. Now, consider all $\binom{m+1}{m}$ subsets of $\{c_1, c_2, \ldots, c_{m-1}, u, v\}$, and let M be the number of independent sets among those $\binom{m+1}{m}$ subsets. Each of those M subsets appears as an independent set monomial in A. Therefore, the monomial $x_I x_v$ is created M times in AB, and the coefficient for $x_I x_v$ in AB is $-ML_m$. The monomial $x_I x_v$ does not appear in D, because it is a non-independent set monomial, and it appears exactly M times in C. Therefore, the coefficient for $x_I x_v$ in C is ML_m, and we see that

$$\underbrace{-ML_m}_{\text{from } AB} + \underbrace{ML_m}_{\text{from } C} = 0 \ .$$

- **independent set with one variable squared:** Let $I = \{c_1, c_2, \ldots, c_{m-1}, k\}$ be any independent set of size m, and consider the monomial $x_{I\setminus k}x_k^2$. This monomial is created in AB by the direct product $x_I x_k$, and the coefficient is $-L_m$. This monomial is not created in C, because it contains no edges, and it is created in D by $x_{I\setminus k}x_k^2$. Thus, the coefficient for $x_I x_k$ in D is L_m, and we see that

$$\underbrace{-L_m}_{\text{from } AB} + \underbrace{L_m}_{\text{from } D} = 0 \ .$$

Since the constant term in $AB + C + D$ is one, and the coefficient for every other monomial is zero, Eq. 4.3 is a Nullstellensatz certificate of degree $\alpha(G)$. □

Example 4.2.3 Figure 4.1 depicts the $T(5,3)$ Turán graph. It is clear that $\alpha(T(5,3)) = 2$. Therefore, we construct a certificate via Theorem 4.2.2 verifying the non-existence of an independent set of size 3.

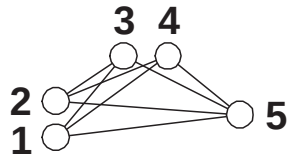

Figure 4.1: Turán graph $T(5,3)$

$$
\begin{aligned}
1 = {} & \left(\frac{1}{3}x_4 + \frac{1}{3}x_2 + \frac{1}{3}\right)x_1x_3 + \left(\frac{1}{3}x_2 + \frac{1}{3}\right)x_1x_4 + \left(\frac{1}{3}x_2 + \frac{1}{3}\right)x_1x_5 + \left(\frac{1}{3}x_4 + \frac{1}{3}\right)x_2x_3 + \\
& \left(\frac{1}{3}\right)x_2x_4 + \left(\frac{1}{3}\right)x_2x_5 + \left(\frac{1}{3}x_4 + \frac{1}{3}\right)x_3x_5 + \left(\frac{1}{3}\right)x_4x_5 + \left(\frac{1}{3}x_2 + \frac{1}{6}\right)(x_1^2 - x_1) + \\
& \left(\frac{1}{3}x_1 + \frac{1}{6}\right)(x_2^2 - x_2) + \left(\frac{1}{3}x_4 + \frac{1}{6}\right)(x_3^2 - x_3) + \left(\frac{1}{3}x_3 + \frac{1}{6}\right)(x_4^2 - x_4) + \left(\frac{1}{6}\right)(x_5^2 - x_5) + \\
& \underbrace{\left(-\frac{1}{3}(x_1x_2 + x_3x_4) - \frac{1}{6}(x_1 + x_2 + x_3 + x_4 + x_5) - \frac{1}{3}\right)}_{A}(x_1 + x_2 + x_3 + x_4 + x_5 - 3)\ .
\end{aligned}
$$

In this example, note that A, Q_i, Q_{ij} are polynomials in $\mathbb{Q}[x_1, \ldots, x_n]$, and furthermore, note that A contains one monomial for every independent set in $T(5,3)$. For example, note that the term $-\frac{1}{3}x_1x_2$ corresponds to the independent set formed by vertices 1 and 2 in $T(5,3)$. Additionally, every monomial in every coefficient is also an independent set in $T(5,3)$. □

We will now prove that the stability number $\alpha(G)$ is the minimum-degree for any Nullstellensatz certificate for the non-existence of an independent set of size greater than

$\alpha(G)$. To prove this, we rely on two lemmas, which represent the base case and the inductive step in the final inductive proof.

Lemma 4.2.4 *Let G be a graph, and let*

$$1 = A'\Big(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i \Big) + \sum_{\{i,j\}\in E(G)} Q'_{ij} x_i x_j + \sum_{i=1}^{n} Q'_i (x_i^2 - x_i) \ , \qquad (4.4)$$

be a reduced *(via Lemma 4.2.1) Nullstellensatz certificate proving the non-existence of a independent set of size $\alpha(G)+r$ (for $r \geq 1$). Then the constant term in A' is $-L_0$, and the coefficient for x_i in A' is $-L_1$.*

Proof: The certificate presented in Eq. 4.4 must simplify to one. We begin by letting B, C and D equal

$$1 = A' \underbrace{\Big(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i \Big)}_{B} + \underbrace{\sum_{\{i,j\}\in E(G)} Q'_{ij} x_i x_j}_{C} + \underbrace{\sum_{i=1}^{n} Q'_i (x_i^2 - x_i)}_{D} \ ,$$

A constant only appears in the expanded certificate $A'B + C + D$ via the product of a constant term in A' and the constant term in B. Therefore, letting β_0 be the constant term in A', we see

$$-(\alpha(G)+r)\beta_0 = 1 \ , \quad \Longrightarrow \quad \beta_0 = -\frac{1}{\alpha(G)+r} = -L_0 \ .$$

Now let β_i be the coefficient of x_i in A' and let $D = \deg(Q'_i)$. Therefore,

$$Q'_i = M_D x_i^D + M_{D-1} x_i^{D-1} + \cdots + M_1 x_i + M_0$$
$$+ \text{ other terms in } Q'_i \text{ that are not powers of } x_i \ .$$

Now consider the coefficients for x_i, x_i^2 in the expanded certificate $A'B+C+D$, which must simplify to zero:

$$\mathbf{x_i}: \quad 0 = \beta_i\big(-(\alpha(G)+r)\big) - M_0 - L_0 \ ,$$

$$\mathbf{x_i^2}: \quad 0 = \beta_i + M_0 - M_1 \ .$$

Now consider the coefficients for the monomials $x_i^{D+2}, x_i^{D+1}, \ldots, x_i^3$ in the expanded certificate $A'B + C + D$, which are $M_D, -M_D + M_{D-1}, -M_{D-1} + M_{D-2}, \ldots, -M_2 + M_1$. Each of these coefficients must simplify to zero, which implies each of these equations is equal to zero. When the coefficients for $x_i^{D+2}, x_i^{D+1}, \ldots, x_i^3, x_i^2$ (note that x_i is excluded) are summed together in *one* equation, the sum telescopes and the terms cancel yielding $\beta_i + M_0 = 0$. Therefore, the equation for x_i becomes:

$$\beta_i\big(-(\alpha(G)+r)\big) + \beta_i - L_0 = 0 \ ,$$

$$\beta_i(\alpha(G)+r) - \beta_i = -L_0 \ ,$$

$$\beta_i = -\frac{L_0}{\alpha(G)+r-1} \ ,$$

$$\beta_i = -L_1 \ .$$

Thus, we see that coefficient of x_i in A' is equal to $-L_1$. □

Lemma 4.2.5 *Let G be a graph, and let*

$$1 = A'\left(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i\right) + \sum_{\{i,j\}\in E(G)} Q'_{ij}x_ix_j + \sum_{i=1}^{n} Q'_i(x_i^2 - x_i) \ , \qquad (4.5)$$

be a reduced *(via Lemma 4.2.1) Nullstellensatz certificate proving the non-existence of an independent set of size $\alpha(G)+r$ (for $r \geq 1$). Let $I = \{c_1, c_2, \ldots, c_{m+1}\}$ be an independent set in G. If the coefficient for $x_{I\setminus c_i}$ in A' is $-L_m$, then the coefficient for x_I in A' is $-L_{m+1}$.*

Proof: As before, let B, C and D equal

$$1 = A' \underbrace{\Big(-(\alpha(G)+r) + \sum_{i=1}^{n} x_i \Big)}_{B} + \underbrace{\sum_{\{i,j\}\in E(G)} Q'_{ij} x_i x_j}_{C} + \underbrace{\sum_{i=1}^{n} Q'_i (x_i^2 - x_i)}_{D} \ ,$$

Let β_I be the coefficient for x_I in A', denote $x_I^\gamma := x_{c_1}^{\gamma_1} \cdots x_{c_{m+1}}^{\gamma_{m+1}}$ by x_I^γ, set $N = \max\{\deg(Q'_{c_1}), \ldots, \deg(Q'_{c_{m+1}})\}$, and let N_γ be the set of $\{\gamma_1, \ldots, \gamma_{m+1}\}$-tuples such that $\gamma_i \geq 0$ and $\sum_{i=1}^{m+1} \gamma_i \leq N$. Therefore, let

$$Q'_{c_i} = \sum_{\gamma \in N_\gamma} M_{I^\gamma}^{c_i} x_I^\gamma + \text{other terms in } Q'_{c_i} \ .$$

Now consider the coefficients for $x_I, x_{I\setminus c_i} x_{c_i}^2$ in the expanded certificate $A'B + C + D$:

- $\mathbf{x_I}$: This monomial is formed in two ways in $A'B$, $x_I\big(-(\alpha(G)+r)\big)$, or $x_{I\setminus c_i} x_{c_i}$ (formed $m+1$ times, once for each c_i), and formed in one way in D, $x_{I\setminus c_i}(-x_{c_i})$ (formed $m+1$ times, once for each c_i), yielding

$$\beta_I\big(-(\alpha(G)+r)\big) - (m+1)L_m - \underbrace{\sum_{i=1}^{m+1} M_{I\setminus c_i}^{c_i}}_{E} = 0 \ . \tag{4.6}$$

- $\mathbf{x_{I\setminus c_i} x_{c_i}^2}$: This monomial is formed in one way in $A'B$, $x_I x_{c_i}$, and formed in three ways in D, $x_I(-x_{c_i})$, $x_{I\setminus c_i} x_{c_i}^2$, or $x_{c_i}^2 x_{I\setminus\{c_i\cup c_j\}}(-x_{c_j})$ (formed m times, once for each c_j with $j \neq i$), yielding

$$\beta_I - M_I^{c_i} + M_{I\setminus c_i}^{c_i} - \sum_{\substack{j=1 \\ j\neq i}}^{m+1} M_{I\setminus(c_i\cup c_j)}^{c_j} = 0 \ . \tag{4.7}$$

Now we will consider Eq. 4.7 for *each individual* c_i, with $i = 1, \ldots, m+1$, and sum those $m+1$ equations. This yields

$$(m+1)\beta_I - \sum_{i=1}^{m+1} M_I^{c_i} + \underbrace{\sum_{i=1}^{m+1} M_{I\setminus c_i}^{c_i}}_{E} - \sum_{i=1}^{m+1} \sum_{\substack{j=1 \\ j\neq i}}^{m+1} M_{I\setminus(c_i\cup c_j)}^{c_j} = 0 \ . \tag{4.8}$$

Notice that part E in Eq. 4.8 is equal to part E in Eq. 4.6. Now, as in Lemma 4.2.4, we sum Eq. 4.8 with the equations for the coefficients of *every other monomial* x_I^γ in Q'_{c_i}, excluding x_I (and thus, Eq. 4.6). As before, every $M_{I^\gamma}^{c_i}$ appears in exactly two equations, once with a positive sign and once with a negative sign, (corresponding to the multiplication $x_{c_i}^2$ and $-x_{c_i}$, respectively). Thus, when Eq. 4.8 is summed with the equations corresponding to every other monomial *excluding* x_I, the sum will telescope and every $M_{I^\gamma}^{c_i}$ *excluding part* E will cancel. The negative component for part E is contained in Eq. 4.6, which is *not* included in this sum, which is why part E does *not* cancel. Thus, we see

$$(m+1)\beta_I = -\underbrace{\sum_{i=1}^{m+1} M_{I\setminus c_i}^{c_i}}_{E} \ . \tag{4.9}$$

Substituting Eq. 4.9 into Eq. 4.6, we see

$$\begin{aligned}
\beta_I\big(-(\alpha(G)+r)\big) - (m+1)L_m + (m+1)\beta_I &= 0 \ , \\
\beta_I(\alpha(G)+r) - (m+1)\beta_I &= -(m+1)L_m \ , \\
\beta_I &= -\frac{(m+1)L_m}{\alpha(G)+r-(m+1)} \ , \\
\beta_I &= -L_{m+1} \ .
\end{aligned}$$

Thus, the coefficient of x_I in A' is equal to $-L_m$. □

Using Lemmas 4.2.4 and 4.2.5, we can now prove the main theorem of this section.

Theorem 4.2.6 *Given a graph G, a Nullstellensatz certificate (associated with the Lovász encoding of Lemma 2.1.1) for the non-existence of an independent set of size greater than $\alpha(G)$ has degree at least $\alpha(G)$.*

Proof: Our proof is by contradiction. Let

$$1 = A\Big(- (\alpha(G) + r) + \sum_{i=1}^{n} x_i \Big) + \sum_{\{i,j\} \in E(G)} Q_{ij} x_i x_j + \sum_{i=1}^{n} Q_i (x_i^2 - x_i)$$

be any Nullstellensatz certificate for the non-existence of an independent set of size $\alpha(G)+r$, with $r \geq 1$, such that $\deg(A), \deg(Q_i), \deg(Q_{ij}) < \alpha(G)$, and let

$$1 = A' \underbrace{\Big(- (\alpha(G) + r) + \sum_{i=1}^{n} x_i \Big)}_{B} + \underbrace{\sum_{\{i,j\} \in E(G)} Q'_{ij} x_i x_j}_{C} + \underbrace{\sum_{i=1}^{n} Q'_i (x_i^2 - x_i)}_{D} \tag{4.10}$$

be the reduced certificate via Lemma 4.2.1. The proof of Lemma 4.2.1 implies $\deg(A') \leq \deg(A) < \alpha(G)$. Let $M = \{c_1, c_2, \ldots, c_{\alpha(G)}\}$ be any maximum independent set in G. Via Lemma 4.2.4, we know that x_{c_1} appears in A' with coefficient $-L_1$, which implies (via Lemma 4.2.5) that $x_{c_1} x_{c_2}$ appears in A' with coefficient $-L_2$, which implies that $x_{c_1} x_{c_2} x_{c_3}$ appears in A' with coefficient $-L_3$ and so on. In particular, $x_{c_1} x_{c_2} \cdots x_{c_{\alpha(G)}}$ appears in A' with coefficient $-L_{\alpha(G)}$. This contradicts our assumption that $\deg(A') < \alpha(G)$. Therefore, there can be no Nullstellensatz certificate with $\deg(A) < \alpha(G)$; thus, the degree of *any* Nullstellensatz certificate is at least $\alpha(G)$. □

Lemmas 4.2.4 and 4.2.5 also give rise to the following corollary.

Corollary 4.2.7 *Given a graph G, a Nullstellensatz certificate (associated with the Lovász encoding of Lemma 2.1.1) for the non-existence of a independent set of size greater than $\alpha(G)$ contains at least one monomial for every independent set in G.*

Proof: Given any Nullstellensatz certificate associated with the Lovász encoding of Lemma 2.1.1, we can create the reduced certificate via Lemma 4.2.1. The proof of the Lemma 4.2.1

implies that the number of terms in A is equal to the number of terms in A'. Via Lemmas 4.2.4 and 4.2.5, A' contains one monomial for every independent set in G. Therefore, A also contains one monomial for every independent set in G. □

This brings us to the last theorem of this section.

Theorem 4.2.8 *Given a graph G, a minimum-degree Nullstellensatz certificate (associated with the Lovász encoding of Lemma 2.1.1) for the non-existence of an independent set of size greater than $\alpha(G)$ has degree equal to $\alpha(G)$ and contains at least one term for every independent set in G.*

Proof: This theorem follows directly from Theorems 4.2.2, 4.2.6, and Corollary 4.2.7. □

Our results establish new lower bounds for the degree and number of terms in Nullstellensatz certificates. In earlier work, researchers in logic and complexity showed both logarithmic and linear growth in the degree of Nullstellensatz certificates over finite fields or for special instances, e.g. Nullstellensatz related to the pigeonhole principle (see [8], [26] and references therein). Our main complexity result below settles the open question posed by Lovász [41]:

Corollary 4.2.9 *There exist infinite families of graphs G_n, on n vertices, such that the degree of a minimum-degree Nullstellensatz certificate (associated with the Lovász encoding of Lemma 2.1.1) grows linearly in n and, at the same time, the number of terms in the coefficient polynomials of the Nullstellensatz certificate is exponential in n.*

Proof: We describe two infinite families explicitly. First, the disjoint union of $n/3$ triangles has exactly $4^{n/3} - 1$ independent sets and the minimum-degree of the Nullstellensatz certificates is $\alpha(G) = n/3$. Second, graphs with no edges have $\alpha(G) = n$, and the number of independent sets is 2^n. □

In this section, we have provided a thorough description of the independent set coNP certificates provided by **N**ul**LA**. Although in the process we have shed no new light on complexity class inclusions, it is somewhat surprising that the polynomial identities provided by Hilbert's Nullstellensatz so clearly and directly represent combinatorial properties of the underlying graph. Furthermore, given a graph, the Nullstellensatz certificate required to verify a "no" instance of the independent set decision problem contains all of the information about every "yes" instance for the graph.

From a computational perspective, the density of these certificates represents a serious obstacle. In this case, we have demonstrated that computing Hilbert's Nullstellensatz is at least as hard as counting all possible independent sets in a graph, which is known to be $\#P$-complete, even for graphs with low-degree vertices [18]. Furthermore, we strongly expect, based on the structure of our proof and the resulting telescopic sums, that *any* polynomial system containing 0/1 equations ($x_i^2 - x_i = 0$) will exhibit similar computational difficulties. This has ramifications for Gröbner bases computations, showing that any such computation can generate exponentially many intermediate monomials. On the other hand, it suggests that the natural binary encodings of combinatorial problems are perhaps not the most desirable for **N**ul**LA**.

Finally, we note that, since the degrees of the polynomials from the Lemma 2.1.1 encoding of independent set are less than or equal to two, and because we have demonstrated *linear* growth in the minimum-degree of the associated Nullstellensatz certificates, we have shown that the Lazard bound on projective Nullstellensatz certificates presented in Corollary 3.2.4 is tight.

4.3 Graph 3-Coloring and the Nullstellensatz

In this section, we explore Nullstellensatz certificates for the NP-complete problem of graph 3-colorability. We rely on two different encodings, described in Lemmas 2.2.2 and 2.2.6, which are over $\mathbb{C}$ and $\overline{\mathbb{F}_2}$ respectively. Although we do not yet have an encompassing theorem for graph 3-colorability that is comparable to Theorem 4.2.8 for independent set, there is strong computational and theoretical evidence that non-k-colorability certificates capture combinatorial properties of underlying non-k-colorable subgraphs. In Subsection 4.3.1, we describe the structure and combinatorial meaning of non-2-colorability certificates. In Subsection 4.3.2, we explore minimum-degree non-3-colorability certificates, touching on their growth and structure, and proving lower bounds on their minimum degree (different for each encoding). In Subsection 4.3.3, we explore two common non-3-colorable subgraphs (cliques and odd-wheels), and illustrate the somewhat surprising result that as the underlying graphs grow, the degree of the Nullstellensatz certificates remains fixed and the coefficients remain sparse. We conclude with a few remarks on the theoretical applications of a deeper understanding of the combinatorial meaning of these certificates.

4.3.1 NulLA 2-colorability is in P

As we continue to explore the theoretical complexity of **Nul**$\mathbf{LA}$, it is natural to ask about its performance on systems of polynomial equations representing problems known to be in P. Unlike graph 3-colorability, graph 2-colorability is easily solvable in linear time. In this subsection, we show that the Nullstellensatz certificates for non-2-colorability are likewise sparse and of fixed-degree.

A bipartite graph is any graph whose vertices can be partitioned into two sets, where no two vertices in a set are adjacent. Two well-known facts from graph theory are: 1) A graph is 2-colorable if and only if the graph is bipartite, and 2) A graph is bipartite if and only if it does *not* contain an odd-cycle. Based on this link between odd-cycles and non-2-colorable graphs, we have the following proposition:

Proposition 4.3.1 *Given a non-2-colorable graph containing the odd-cycle* $\{v_{i_1}, v_{i_2}, \ldots, v_{i_k}\}$*, there exists a Nullstellensatz certificate for non-2-colorability of the following form:*

$$1 = -(x_{i_1}^2 - 1) + \frac{1}{2}x_{i_1}(x_{i_1} + x_{i_2}) - \frac{1}{2}x_{i_1}(x_{i_2} + x_{i_3}) + \cdots - \frac{1}{2}x_{i_1}(x_{k-1} + x_k) + \frac{1}{2}x_{i_1}(x_{i_k} + x_{i_1})$$

Proof: By inspection, the certificate simplifies to one because the sum telescopes as it "follows" the odd-cycle. □

It is interesting to note that non-2-colorability certificates capture the central, relevant combinatorial property of the underlying graph. We also note that $\mathbb{C}$ is the algebraically closed field of this encoding, because $(x_i^2 + 1) = 0$ does not have two roots of unity over $\overline{\mathbb{F}_2}$.

4.3.2 Minimum-degree non-3-colorability Nullstellensatz certificates

In this subsection, we explore the coefficients and degree of non-3-colorability certificates.

Growth and structure

We begin by bounding the number of *necessary* monomials appearing in the non-3-colorability certificates. Although these lemmas do not help with understanding the combinatorial meaning of the certificates, they are invaluable from a computational perspective. We conclude by proving that the minimum-degree is a number of the form $3q+1$ with $q \in \mathbb{Z}_{\geq 0}$. We note that the following results are easily generalizable to k-colorability (often as simple as changing 3 to k), and apply regardless of which encoding (Lemma 2.2.2 or 2.2.6) is used, although each result is only proven for 3-colorability over $\mathbb{C}$.

Lemma 4.3.2 *Given a non-3-colorable graph G, the minimum-degree Nullstellensatz certificate associated with the Lemma 2.2.2 encoding has degree $3q$ or $3q+1$, where $q \in \mathbb{Z}_{\geq 0}$.*

Proof: Since G is non-3-colorable, there exists a minimum-degree Nullstellensatz certificate of degree D for some D. Let $D = 3q + r$ for $q \in \mathbb{Z}_{\geq 0}$ and remainder $r \in \{0, 1, 2\}$, and write the Nullstellensatz certificate in the following form:

$$1 = \sum_{i=1}^{n} \beta_i (x_i^3 - 1) + \sum_{\{i,j\} \in E(G)} \beta_{ij} (x_i^2 + x_i x_j + x_j^2) + \sum_{i=1}^{n} \beta_i' (x_i^3 - 1)$$
$$+ \sum_{\{i,j\} \in E(G)} \beta_{ij}' (x_i^2 + x_i x_j + x_j^2) \ ,$$

where the $\beta_i, \beta_{ij}, \beta_i', \beta_{ij}'$ have the following properties:

1. β_i: contains only monomials of degrees that are 0 mod 3 .

2. β'_i: contains only monomials of degrees that are *not* 0 mod 3 .

3. β_{ij}: contains only monomials of degrees that are 1 mod 3 .

4. β'_{ij}: contains only monomials of degrees that are *not* 1 mod 3 .

Thus, when expanded, $\beta_i(x_i^3 - 1)$ and $\beta_{ij}(x_i^2 + x_i x_j + x_j^2)$ only yield monomials of degrees $0, 3, 6, \ldots, 3q + 3$, while $\beta'_i(x_i^3 - 1)$ and $\beta'_{ij}(x_i^2 + x_i x_j + x_j^2)$ only yield monomials *not* of degrees $0, 3, 6, \ldots, 3q + 3$. Thus, we can see

$$1 = \underbrace{\sum_{i=1}^{n} \beta_i(x_i^3 - 1) + \sum_{\{i,j\} \in E(G)} \beta_{ij}(x_i^2 + x_i x_j + x_j^2)}_{\text{simplifies to 1}} + \underbrace{\sum_{i=1}^{n} \beta'_i(x_i^3 - 1) + \sum_{\{i,j\} \in E(G)} \beta'_{ij}(x_i^2 + x_i x_j + x_j^2)}_{\text{simplifies to 0}} .$$

Therefore, we can simply set $\beta'_i = \beta'_{ij} = 0$ for all β'_i, β'_{ij}, and the degree of the Nullstellensatz certificate is $\max\{\deg(\beta_i), \deg(\beta_{ij})\}$, which is $3q$ or $3q + 1$ (depending on whether the remainder $r = 0$, or $r \geq 1$). □

Lemma 4.3.3 *Given a non-3-colorable graph G and an integer $q \in \mathbb{Z}_{\geq 0}$, there does not exist a Nullstellensatz certificate of degree $3q$ associated with the Lemma 2.2.2 encoding.*

Proof: Our proof is by contradiction. Assume there exists a Nullstellensatz certificate of degree $3q$ with $q \geq 1$ ($q = 0$ is the trivial case). By the proof of Lemma 4.3.2, we can write the certificate in the following form:

$$1 = \sum_{i=1}^{n} \beta_i(x_i^3 - 1) + \sum_{\{i,j\} \in E(G)} \beta_{ij}(x_i^2 + x_i x_j + x_j^2) ,$$

where the β_i, β_{ij} coefficients have the following properties:

1. β_i: contains only monomials of degrees that are 0 mod 3 .

2. β_{ij}: contains only monomials of degrees that are 1 mod 3 .

We will show that every monomial of degree $3q$ in β_i must have coefficient zero. We start by defining a *reduced* monomial as any monomial in β_i where the exponent of every variable x_j, with $j < i$, is ≤ 2.

Consider a given monomial mx_j^d in β_i. If $j < i$ and $d > 2$, the monomial is *not* reduced. We will transfer every non-reduced variable x_j with exponent > 2 and $j < i$ from β_i into β_j, using the following algebraic relation, or *syzygy*:

$$mx_j^3(x_i^3 - 1) = (mx_i^3 - m)(x_j^3 - 1) + m(x_i^3 - 1) \ .$$

Note that this syzygy *replaces* monomials of higher degree (mx_j^3) with monomials of *lower* degree (m), by transferring them from one coefficient (β_i) to another (β_j).

We begin by applying this syzygy to β_n, removing variables $x_{n-1}, \ldots, x_1$ with exponents > 2, and then to β_{n-1}, removing variables $x_{n-2}, \ldots, x_1$ with exponents > 2, and continue until every monomial in β_i is reduced.

Now we will argue that all monomials of degree $3q$ cannot cancel in the expanded certificate, and therefore must have coefficient zero.

Consider a reduced monomial m of degree $3q$ in β_i. The product $m(x_i^3 - 1)$ produces a monomial mx_i^3 where $\deg(mx_i^3) = 3q + 3$. This monomial must cancel with another monomial of degree $3q + 3$ in the expanded certificate. However, every other monomial of degree $3q + 3$ is formed by some $m'(x_j^3 - 1)$. Either the exponent of x_j is < 2 in m, or the exponent of x_i is < 2 in m', because both m and m' are reduced. Thus, mx_i^3 cannot cancel

in the expanded certificate, and the coefficient must be zero. Since this argument applies to every reduced monomial of degree $3q$ in β_i, and every monomial is reduced, there are no monomials of degree $3q$ in β_i. Therefore, there cannot exist a certificate of degree $3q$. □

The previous two lemmas together describe the coefficients and degree-growth in minimum-degree Nullstellensatz certificates. To summarize these lemmas clearly, they imply that a minimum-degree Nullstellensatz certificate is a number of the form $3q+1$ with $q \in \mathbb{Z}_{\geq 0}$.

The next lemma allows us to further simplify the form of the certificate, in the special case when the underlying graph is connected. This "special" case is actually the standard case, because the colorability of a graph is determined by the colorability of its connected subgraphs.

Lemma 4.3.4 *Given a connected non-3-colorable graph, there exists a minimum-degree Nullstellensatz certificate of degree* $3q+1$*, with* $q \in \mathbb{Z}_{\geq 0}$*, of the form*

$$1 = \beta_r(x_r^3 - 1) + \sum_{\{i,j\} \in E(G)} \beta_{ij}(x_i^2 + x_i x_j + x_j^2) \ ,$$

where r *is an arbitrary vertex in the graph.*

Proof: Via Lemma 4.3.3, we know that there exists a Nullstellensatz certificate of the following form

$$1 = \sum_{i=1}^{n} \beta_i(x_i^3 - 1) + \sum_{\{i,j\} \in E(G)} \beta_{ij}(x_i^2 + x_i x_j + x_j^2) \ , \tag{4.11}$$

where β_i, β_{ij} contain only monomials of degree that are 0 mod 3, and 1 mod 3, respectively. Our proof relies on repeated substitutions of the following algebraic relation, or *syzygy*, into

Eq. 4.11:

$$(x_j^3 - 1) = (x_i^3 - 1) + (x_j - x_i)(x_i^2 + x_i x_j + x_j^2) \ .$$

Note that this syzygy holds over $\mathbb{F}_2$, as well as being generalizable to the case of k-colorability. Thus, given the vertex polynomial $(x_j^3 - 1)$, we can substitute the syzygy $(x_i^3 - 1) + (x_j - x_i)(x_i^2 + x_i x_j + x_j^2)$ as long as *there exists an edge from i to j*. Therefore, since G is a connected graph, we can remove every vertex polynomial except one (arbitrarily $x_r^3 - 1$) by tracing paths from some arbitrarily chosen "root" vertex r to every other vertex, and then substituting the appropriate syzygy working backwards from the end of the path. For example, given a path $\{v_r, v_{r_2}, v_{r_3}, \ldots, v_{r_k}\}$, we can remove the vertex polynomial $(x_{r_k}^3 - 1)$ with the substitution $(x_{r_{k-1}}^3 - 1) + (x_{r_k} - x_{r_{k-1}})(x_{r_k}^2 + x_{r_k} x_{r_{k-1}} + x_{r_{k-1}}^2)$, etc. As long as we are careful about the *order* in which we remove vertex polynomials, since G is connected, we can remove every vertex polynomial (except $(x_r^3 - 1)$), with similar substitutions. In terms of degree, when a substitution occurs,

$$\beta_j(x_j^3 - 1) \Longrightarrow \beta_j(x_i^3 - 1) + \beta_j(x_j - x_i)(x_i^2 + x_i x_j + x_j^2) \ .$$

Since $\deg(\beta_j) \leq 3q$, the new coefficient β_i' for vertex polynomial $(x_i^3 - 1)$ becomes $\beta_i + \beta_j$, and $\deg(\beta_i')$ is still $\leq 3q$. The new coefficient β_{ij}' for edge polynomial $(x_i^2 + x_i x_j + x_j^2)$ becomes $\beta_i(x_j - x_i) + \beta_{ij}$. Thus, $\deg(\beta_{ij}') \leq 3q + 1$. Furthermore, since β_i only contains monomials of degrees that are 0 mod 3, $\beta_i(x_j - x_i)$ only contains monomials of degrees that are 1 mod 3. □

The previous lemma allows us to delete vertex polynomials until only one (arbitrarily $x_r^3 - 1$) remains. We summarize all of the previous lemmas in a final theorem.

Theorem 4.3.5 *A minimum-degree Nullstellensatz certificate associated with the Lemma 2.2.2 encoding for non-3-colorability has degree* $3q + 1$ *with* $q \in \mathbb{Z}_{\geq 0}$. *Furthermore, there exists a minimum-degree Nullstellensatz certificate of the form*

$$1 = \beta_r(x_r^3 - 1) + \sum_{\{i,j\} \in E(G)} \beta_{ij}(x_i^2 + x_i x_j + x_j^2) \ ,$$

with the following properties: 1) β_r *only contains monomials of degrees* $0 \bmod 3$, *2)* β_{ij} *only contains monomials of degrees* $1 \bmod 3$, *and 3)* r *is an arbitrary vertex in a connected, non-3-colorable subgraph.*

The significance of Theorem 4.3.5 with respect to practical computation is immediate. In the generalized description of **N**ul**LA** (Section 3), if there is no certificate of degree d, the degree is incremented to $d + 1$, and the certificate construction process is repeated. However, Theorem 4.3.5 suggests that a specialized version of **N**ul**LA**, specifically devoted to graph 3-colorability, could skip certain degrees such as $2, 3, 5, 6, \ldots$, etc.. Furthermore, this specialized version of **N**ul**LA** could only construct Nullstellensatz certificates with a specific subset of monomials. This significantly reduces the size of the linear system, which in turn significantly reduces computation time.

The following table represents the size difference in linear systems before and after applying Theorem 4.3.5. We choose a graph of 40 vertices and 80 edges, because sparse graphs are considered more interesting for graph 3-colorability, since denser graphs are more

likely to contain 4-cliques as a subgraph, and thus be trivially non-3-colorable.

# of vertices	# of edges	deg	# of unknowns	# of unknowns *after* simplification	% savings
40	80	1	4,920	3,201	35%
40	80	4	16,290,120	9,872,802	40%
40	80	7	9,485,504,160	4,281,974,403	55%

We note that the above table displays the savings in terms of the sizes of the linear systems only; since solving a linear system is cubic in the size of the linear system, the percentage of savings in terms of computation time will be even more significant.

$\mathbb{C}$ vs. $\overline{\mathbb{F}_2}$

In this subsection, we compare the minimum-degree non-3-colorability Nullstellensatz certificate over $\mathbb{C}$ to the minimum-degree non-3-colorability Nullstellensatz certificate over $\overline{\mathbb{F}_2}$. In particular, we discover that the minimum-degree non-3-colorability Nullstellensatz certificate over $\mathbb{C}$ is at least four, while the minimum-degree non-3-colorability Nullstellensatz certificate over $\overline{\mathbb{F}_2}$ is at least one.

Lemma 4.3.6 *Using the encoding over* $\mathbb{C}$ *presented in Lemma 2.2.2, every Nullstellensatz certificate for non-3-colorability has degree* at least *four.*

Proof: Our proof is by contradiction. Suppose there exists a Nullstellensatz certificate of degree three or less. Such a certificate has the following form

$$1 = \sum_{i=1}^{n} P_{\{i\}}(x_i^3 - 1) + \sum_{\{i,j\} \in E} P_{\{ij\}}(x_i^2 + x_i x_j + x_j^2) \ , \tag{4.12}$$

where $P_{\{i\}}$ and $P_{\{ij\}}$ represent general polynomials of degree less than or equal to three. To be precise,

$$P_{\{i\}} = \sum_{s=1}^{n} a_{\{i\}s} x_s^3 + \sum_{s=1}^{n} \sum_{\substack{t=1 \\ t\neq s}}^{n} b_{\{i\}st} x_s^2 x_t + \sum_{s=1}^{n} \sum_{t=s+1}^{n} \sum_{u=t+1}^{n} c_{\{i\}stu} x_s x_t x_u + \sum_{s=1}^{n} \sum_{t=1}^{n} d_{\{i\}st} x_s x_t + \sum_{s=1}^{n} e_{\{i\}s} x_s + f_{\{i\}} \ ,$$

and

$$P_{\{ij\}} = \sum_{s=1}^{n} a_{\{ij\}s} x_s^3 + \sum_{s=1}^{n} \sum_{\substack{t=1 \\ t\neq s}}^{n} b_{\{ij\}st} x_s^2 x_t + \sum_{s=1}^{n} \sum_{t=s+1}^{n} \sum_{u=t+1}^{n} c_{\{ij\}stu} x_s x_t x_u + \sum_{s=1}^{n} \sum_{t=1}^{n} d_{\{ij\}st} x_s x_t + \sum_{s=1}^{n} e_{\{ij\}s} x_s + f_{\{ij\}} \ .$$

Because we work with undirected graphs, note that $a_{\{ij\}s} = a_{\{ji\}s}$, and this fact applies to all coefficients a through f. Note also that when $\{i, j\}$ is *not* an edge of the graph, $P_{ij} = 0$ and thus $a_{\{ij\}s} = 0$. Again, this fact holds for all coefficients a through f.

When $P_{\{i\}}$ multiplies $(x_i^3 - 1)$, this generates cross-terms of the form $P_{\{i\}} x_i^3$ and $-P_{\{i\}}$. In particular, this generates monomials of degree six or less. Notice that $P_{\{ij\}}(x_i^2 + x_i x_j + x_j^2)$ does *not* generate monomials of degree six, only monomials of degree five or less. We begin the process of deriving a contradiction from Eq. 4.12 by considering all monomials of the form $x_s^3 x_i^3$ that appear in the expanded Nullstellensatz certificate. These monomials are formed in only *two* ways: Either (1) $x_s^3(x_i^3 - 1)$, or (2) $x_i^3(x_s^3 - 1)$. Since the certificate must simplify to zero, the n^2 equations for $x_s^3 x_i^3$ are either $a_{\{i\}i} = 0$ for x_i^6, or $a_{\{s\}i} + a_{\{i\}s} = 0$ for $x_s^3 x_i^3$. Summing these equations, we see

$$0 = \sum_{i=1}^{n} \sum_{s=1}^{n} a_{\{i\}s} \ . \tag{4.13}$$

Let us now consider monomials of the form $x_s^2 x_t x_i^3$ (with $s \neq t$). These monomials are formed in only *one* way: by multiplying $b_{\{i\}st} x_s^2 x_t$ by x_i^3. Therefore, because the coefficient for $x_s^2 x_t x_i^3$ must simplify to zero in the expanded Nullstellensatz certificate, $b_{\{i\}st} = 0$ for all $b_{\{i\}}$. When we consider monomials of the form $x_s x_t x_u x_i^3$ (with $s < t < u$), we see that $c_{\{i\}stu} = 0$ for all $c_{\{i\}}$, for the same reasons as above.

As we continue toward our contradiction, we now consider monomials of degree three in the expanded Nullstellensatz certificate. In particular, we consider the coefficient for x_s^3. The monomial x_s^3 is generated in three ways: (1) $f_{\{s\}}(x_s^3 - 1)$, (2) $a_{\{i\}s} x_s^3 (x_i^3 - 1)$ (from the vertex polynomials), and (3) $e_{\{st\}s} x_s (x_s^2 + x_s x_t + x_t^2)$ (from the edge polynomials). The n equations for x_s^3 are of the following form:

$$0 = f_{\{s\}} - \sum_{i=1}^{n} a_{\{i\}s} + \sum_{t \in \mathrm{Adj}(s)} e_{\{st\}s} \ .$$

Summing these equations, we see

$$0 = \sum_{i=1}^{n} f_{\{i\}} - \left(\sum_{i=1}^{n} \sum_{s=1}^{n} a_{\{i\}s} \right) + \sum_{s=1}^{n} \sum_{t \in \mathrm{Adj}(s)}^{n} e_{\{st\}s} \ . \tag{4.14}$$

Because the degree three or less Nullstellensatz certificate (Eq. 4.12) is identically one, the constant terms must sum to one. Therefore, we know $\sum_{i=1}^{n} f_{\{i\}} = -1$. Furthermore, recall that $e_{\{st\}s} = 0$ if the undirected edge $\{s, t\}$ does not exist in the graph. Therefore, applying Eq. 4.13 to Eq. 4.14, we have the following equation

$$1 = \sum_{s=1}^{n} \sum_{\substack{t=1, \\ s \neq t}}^{n} e_{\{st\}s} \ . \tag{4.15}$$

To give some intuition for our overall proof strategy, the equations to come will ultimately show that the right-hand side of Eq. 4.15 also equals zero, which is a contradiction.

Now we will consider monomials of the form $x_s^2 x_t$ (with $s \neq t$). We recall that $b_{\{i\}st} = 0$ for all $b_{\{i\}}$ (where $b_{\{i\}st}$ is the coefficient for $x_s^2 x_t$ in the i-th vertex polynomial). Therefore, we do *not* need to consider $b_{\{i\}st}$ in the equation for the coefficient of monomial $x_s^2 x_t$. In other words, we only need to consider the edge polynomials, which can generate this monomial in two ways: (1) $e_{\{st\}s} x_s \cdot x_s x_t$, and (2) $e_{\{si\}t} x_t \cdot x_s^2$. The $2\binom{n}{2}$ equations for these coefficients are of the following form:

$$0 = e_{\{st\}s} + \sum_{i \in \mathrm{Adj}(s)} e_{\{si\}t} \ .$$

Summing these equations, we see

$$\sum_{s=1}^{n} \sum_{\substack{t=1,\\ t\neq s}}^{n} e_{\{st\}s} + \underbrace{\left(\sum_{s=1}^{n} \sum_{t \in Adj(s)} e_{\{st\}t} \right)}_{\text{partial sum A}} + \underbrace{\left(\sum_{s=1}^{n} \sum_{t \in Adj(s)} \sum_{\substack{u=1,\\ u\neq s,t}}^{n} e_{\{st\}u} \right)}_{\text{partial sum B}} = 0 \ . \tag{4.16}$$

However, recall that $e_{\{st\}u} = 0$ when $\{s,t\}$ does not exist in the graph, and also that $e_{\{st\}t} = e_{\{ts\}t}$. Thus, we can rewrite partial sum A from Eq. 4.16 as

$$\sum_{s=1}^{n} \sum_{t \in Adj(s)} e_{\{st\}t} = \sum_{s=1}^{n} \sum_{\substack{t=1,\\ t\neq s}}^{n} e_{\{st\}t} = \sum_{s=1}^{n} \sum_{\substack{t=1,\\ t\neq s}}^{n} e_{\{ts\}t} = \sum_{s=1}^{n} \sum_{\substack{t=1,\\ t\neq s}}^{n} e_{\{st\}s} \ .$$

Substituting the above into Eq. 4.16 yields

$$2 \sum_{s=1}^{n} \sum_{\substack{t=1,\\ t\neq s}}^{n} e_{\{st\}s} + \underbrace{\left(\sum_{s=1}^{n} \sum_{t \in Adj(s)} \sum_{\substack{u=1,\\ u\neq s,t}}^{n} e_{\{st\}u} \right)}_{\text{partial sum B}} = 0 \ . \tag{4.17}$$

Finally, we consider the monomial $x_s x_t x_u$ (with $s < t < u$). We have already argued that $c_{\{i\}stu} = 0$ for all $c_{\{i\}}$ (where $c_{\{i\}stu}$ is the coefficient for $x_s x_t x_u$ in the i-th vertex polynomial). Therefore, as before, we need only consider the edge polynomials, which can generate this monomial in three ways: (1) $e_{\{st\}u} x_u \cdot x_s x_t$, (2) $e_{\{su\}t} x_t \cdot x_s x_u$, and

(3) $e_{\{tu\}s}x_s \cdot x_t x_u$. As before, these coefficients must cancel in the expanded certificate, which yields $\binom{n}{3}$ equations of the following form:

$$0 = e_{\{st\}u} + e_{\{su\}t} + e_{\{tu\}s} \ .$$

Summing these equations, we see

$$\sum_{s=1}^{n-2} \sum_{t=s+1}^{n-1} \sum_{u=t+1}^{n} \Big(e_{\{st\}u} + e_{\{su\}t} + e_{\{tu\}s} \Big) = 0 \ . \tag{4.18}$$

Now we come to the critical argument of the proof. We claim that the following equation holds:

$$\Big(\sum_{s=1}^{n} \sum_{t \in Adj(s)} \sum_{\substack{u=1, \\ u \neq s,t}}^{n} e_{\{st\}u} \Big) = 2 \Big(\sum_{s=1}^{n-2} \sum_{t=s+1}^{n-1} \sum_{u=t+1}^{n} \Big(e_{\{st\}u} + e_{\{su\}t} + e_{\{tu\}s} \Big) \Big) \ . \tag{4.19}$$

Notice that the left-hand and right-hand sides of this equation consist only of coefficients $e_{\{st\}u}$ with s, t, u distinct. Consider any such coefficient $e_{\{st\}u}$. Notice that $e_{\{st\}u}$ appears exactly *once* on the right side of the equation. Furthermore, either $e_{\{st\}u}$ appears exactly *twice* on the left side of this equation (because $s \in \mathrm{Adj}(t)$ implies $t \in \mathrm{Adj}(s)$), or $e_{\{st\}u} = 0$ (because the edge $\{s, t\}$ does not exist in the graph). Therefore, Eq. 4.19 is valid. Applying this result (and Eq. 4.18) to Eq. 4.17 gives us the following:

$$\sum_{s=1}^{n} \sum_{\substack{t=1, \\ t \neq s}}^{n} e_{\{st\}s} = 0 \ . \tag{4.20}$$

But Eq. 4.20 contradicts Eq. 4.15 ($1 = 0$); thus there can be no certificate of degree less than four. □

It is important to note that when we try to construct certificates of degree four or greater, the equations for the degree six monomials become considerably more complicated. In this case, the edge polynomials *do* contribute monomials of degree six, which causes the above argument to break. It is also important to note that this argument does not hold over $\overline{\mathbb{F}_2}$; in $\mathbb{C}$, if $x + x = 2x = 0$, then $x = 0$. However, over $\overline{\mathbb{F}_2}$, $x + x = 2x$ is always zero, regardless of whether $x = 1$ or $x = 0$. Thus, we have the following theorem:

Theorem 4.3.7 *Using the encoding over* $\mathbb{C}$ *presented in Lemma 2.2.2, every Nullstellensatz certificate for non-3-colorability has degree* at least *four, and using the encoding over* $\overline{\mathbb{F}_2}$ *presented in Lemma 2.2.6, every Nullstellensatz certificate for non-3-colorability has degree* at least *one.*

Curiously enough, as we will see in the chapter detailing our experimental results (Chapter 5), the minimum-degree possible (e.g., four with respect to $\mathbb{C}$ and one with respect to $\overline{\mathbb{F}_p}$) is almost always the degree of the certificates returned by **N**ul**LA**.

4.3.3 Subgraphs

The complexity of determining the non-3-colorablity of a given graph is determined in part by the complexity of its non-3-colorable subgraphs. For example, if the entire graph is 4-edge-critical (meaning that the graph has chromatic number four, but if any edge is removed, the chromatic number drops to three), then every edge must appear in the Nullstellensatz certificate. However, if the graph contains a less complicated non-3-colorable subgraph, such as a 4-clique (a 4-complete graph) or an odd-wheel, then we would expect the complexity of the certificate to reflect the existence of polynomial-time algorithm for

finding such subgraphs. In this subsection, we describe the relationship between subgraphs and Nullstellensatz certificate degree complexity (Lemma 4.3.8), and then compare and contrast the certificates for common non-3-colorable subgraphs (mainly cliques and odd-wheels) produced via the $\mathbb{C}$ and $\overline{\mathbb{F}_2}$ encodings of Lemmas 2.2.2 and 2.2.6, respectively.

We begin with a lemma relating subgraphs to Nullstellensatz certificate degree.

Lemma 4.3.8

1. *If H is a subgraph of G, and H has a minimum-degree non-3-colorability Nullstellensatz certificate of degree k, then G also has a minimum-degree non-3-colorability Nullstellensatz certificate of degree k.*

2. *Suppose that a non-3-colorable graph G can be transformed to a non-3-colorable graph H via a sequence of merges of non-adjacent nodes of G. If a minimum-degree non-3-colorability Nullstellensatz certificate for H has degree k, then a minimum-degree non-3-colorable Nullstellensatz certificate for G has degree at least k.*

Proof of 1: Since H is a subgraph of G, then any Nullstellensatz certificate for non-3-colorability of H is also a Nullstellensatz certificate for non-3-colorability of G. □

Proof of 2: Assume that a minimum-degree Nullstellensatz certificate for H has degree k, but G has a Nullstellensatz certificate for non-3-colorability of degree less than k. The certificate has the form $1 = \sum \beta_i(x_i^3 - 1) + \sum \beta_{ij}(x_i^2 + x_i x_j + x_j^2)$ where both β_i and β_{ij} denote polynomials of degree less than k. Since this certificate is an identity, the identity must hold for all values of the variables. In particular, it must hold for every variable substitution $x_i = x_j$ when the nodes are non-adjacent. In this case, the variable reassignment

(pictorially represented in Figure 4.2) yields a Nullstellensatz certificate of degree less than k for the transformed graph H. But this is in contradiction with the assumed degree of a

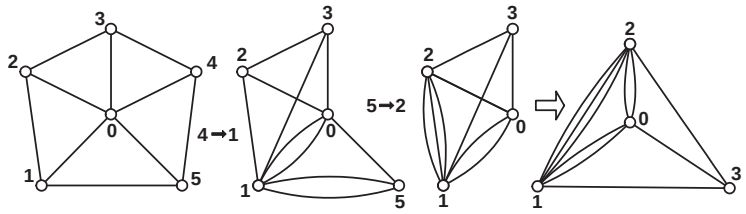

Figure 4.2: Converting G (the 5-odd-wheel) to H (the 3-odd-wheel) via node merges.

minimum-degree certificate for H. Therefore, any certificate for G must have degree at least k. Note that the parallel edges displayed in Figure 4.2 are irrelevant to our considerations. □

The Hadwiger conjecture of 1943 (see [16] and exposition therein) states that, given an undirected graph G with chromatic number greater than k, then G contains k disjoint connected subgraphs such that if each subgraph is contracted to a single supervertex, the vertices form the complete graph K_k. In light of Hadwiger's conjecture, it might be interesting to propose a "Nullstellensatz calculus", and explore the form and structure of certificates produced via edge contractions or node merges, or even the reverse; to explore certificates expanded from minors to larger graphs via pre-computed syzygies.

Continuing with our subgraph investigations, if a given graph G contains a 4-clique as a subgraph, we can simply investigate every $\binom{n}{4}$ subgraphs in four vertices, and check whether those vertices form a 4-clique. Since this algorithm is $O(n^4)$ (polynomial-time), we would expect that the certificates associated with these graphs are likewise sparse and of fixed degree.

Lemma 4.3.9 *Using the encoding over $\mathbb{C}$ presented in Lemma 2.2.2, the complete graph K_n with $n \geq 4$ has a minimum-degree Nullstellensatz certificate for non-3-colorability of degree four.*

Proof: Since K_4 is a subgraph of K_5, which is a subgraph of K_6, etc., then via Lemma 4.3.8, part 1, if there exists a Nullstellensatz certificate of degree four for K_4, then there exists a Nullstellensatz certificate of degree four for K_n ($n \geq 5$). We conclude the proof by displaying the following degree four certificate for K_4:

$$\begin{aligned}
1 = &\left(\frac{4}{9}x_1^4 - \frac{5}{9}x_1^3x_2 - \frac{2}{9}x_1^3x_3 - \frac{4}{9}x_1^3x_0 + \frac{2}{9}x_1^2x_2x_0 + \frac{2}{9}x_1^2x_3x_0\right)(x_1^2 + x_2x_1 + x_2^2) + \\
&\left(\frac{1}{9}x_1^4 + \frac{2}{9}x_1^3x_2 - \frac{1}{9}x_1^3x_0 - \frac{2}{9}x_1^2x_2x_0\right)(x_2^2 + x_3x_2 + x_3^2) + \frac{1}{3}x_1^3x_2(x_2^2 + x_0x_2 + x_0^2) + \\
&\left(\frac{2}{9}x_1^4 + \frac{1}{9}x_1^3x_2 + \frac{1}{9}x_1^3x_0 + \frac{2}{9}x_1^2x_2x_0\right)(x_1^2 + x_3x_1 + x_3^2) + \frac{1}{3}x_1^4(x_1^2 + x_0x_1 + x_0^2) + \\
&\left(-\frac{1}{3}x_1^4 - \frac{1}{3}x_1^3x_2\right)(x_3^2 + x_0x_3 + x_0^2) + (-x_1^3 - 1)(x_1^3 - 1)\ . \qquad (4.21)
\end{aligned}$$

□

Lemma 4.3.10 *Using the encoding over $\overline{\mathbb{F}_2}$ presented in Lemma 2.2.6, K_n with $n \geq 4$ has a minimum-degree Nullstellensatz certificate for non-3-colorability of degree one.*

Proof: Via the proof of Lemma 4.3.9, we conclude by displaying the following degree one certificate for K_4:

$$\begin{aligned}
1 = &(x_1^3 + 1) + (x_2^3 + 1) + (x_3^3 + 1) + x_1(x_1^2 + x_1x_2 + x_2^2) + x_2(x_2^2 + x_2x_3 + x_3^2) \\
&+ x_3(x_3^2 + x_3x_1 + x_1^2) + (x_2 + x_3)(x_0^2 + x_0x_1 + x_1^2) + (x_1 + x_3)(x_0^2 + x_0x_2 + x_2^2) \\
&+ (x_1 + x_2)(x_0^2 + x_0x_3 + x_3^2)\ .
\end{aligned}$$

□

The 4-clique is the smallest, most trivial non-3-colorable graph. By demonstrating that the associated Nullstellensatz certificates, regardless of the encoding used, are sparse and of *minimum* degree, we are demonstrating our first correlation between complexity of certificates and hardness of underlying graphs. We will return to this theme in Section 5.2.6 where we describe our experimental investigations on supposed "hard" instances of 3-colorability. For our next result, we demonstrate that there exist infinite families of non-3-colorable graphs where the certificates remain sparse and of *minimum* degree, regardless of the encoding used, even as the underlying graphs grow infinitely large. To do this, we rely on another canonical example from graph 3-colorability: the odd-wheel.

The odd-wheels consist of an odd-cycle rim, with a center vertex connected to all other vertices. The $(2k+1)$-odd-wheel refers to a rim of length $2k+1$, which implies that the actual graph contains $2k+2$ vertices, and $4k+2$ edges. It is easy to see that the odd-wheels form an infinite family of non-3-colorable graphs, by simply attempting to color the graph and arriving at a contradiction with the edge $\{1, 2k+1\}$.

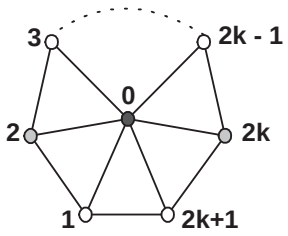

Figure 4.3: The odd-wheels are non-3-colorable.

Lemma 4.3.11 *Using the encoding over $\mathbb{C}$ presented in Lemma 2.2.2, the $(2k+1)$-odd-wheel has a minimum-degree Nullstellensatz certificate for non-3-colorability of degree four.*

Proof: Our proof is by induction on k. We will show that for every k, we can construct a certificate of degree four with very particular properties. By Theorem 4.3.7, any certificate of degree four is minimal. Our base case is $k = 1$. The 3-odd-wheel is isomorphic to K_4 (the 4-complete graph), and a certificate of degree four was previously displayed in Eq. 4.21. Based on that equation, we denote the non-3-colorability certificate for the 3-odd-wheel as follows:

$$1 = \alpha_1 v_1 + \alpha_{\{12\}} e_{\{12\}} + \alpha_{\{23\}} e_{\{23\}} + \widetilde{\alpha} e_{\{13\}} + \alpha_{\{20\}} e_{\{20\}} + \alpha_{\{10\}} e_{\{10\}} + \alpha_{\{30\}} e_{\{30\}} \ ,$$

where $v_1 = x_1^3 - 1$, and $e_{\{ij\}} = x_i^2 + x_i x_j + x_j^2$ and $\alpha_1, \alpha_{\{ij\}}$ and $\widetilde{\alpha}$ denote polynomials of degree four in $\mathbb{C}[x_0, x_1, x_2, x_3]$. In particular, via Eq. 4.21, we see

$$\widetilde{\alpha} = \frac{2}{9} x_1^4 + \frac{1}{9} x_1^3 x_2 + \frac{1}{9} x_1^3 x_0 + \frac{2}{9} x_1^2 x_2 x_0 \ . \tag{4.22}$$

For our induction hypothesis, we assume that there exists a degree four certificate for the $(2k+1)$-odd-wheel of the following specific form:

$$\begin{aligned} 1 = \gamma_1 v_1 + \gamma_{\{12\}} e_{\{12\}} + \cdots + \gamma_{\{2k,2k+1\}} e_{\{2k,2k+1\}} + \widetilde{\alpha} e_{\{1,2k+1\}} + \gamma_{\{10\}} e_{\{10\}} \\ + \cdots + \gamma_{\{0,2k+1\}} e_{\{0,2k+1\}} \ , \end{aligned} \tag{4.23}$$

where $\gamma_1, \gamma_{\{ij\}}$ denote polynomials of degree four in $\mathbb{C}[x_0, x_1, \ldots, x_{2k+1}]$. Note in particular that the coefficient for the edge $\{1, 2k+1\}$ in the $(2k+1)$-odd-wheel certificate is exactly the same as the coefficient for the $\{1, 3\}$ edge in the 3-odd-wheel certificate: both are equal to $\widetilde{\alpha}$.

Now, we will show that there exists a degree four certificate for the $\big(2(k+1)+1\big)$-odd-wheel such that the coefficient for the $\{1, 2(k+1)+1\}$ edge is still $\widetilde{\alpha}$. In Figure 4.4, we

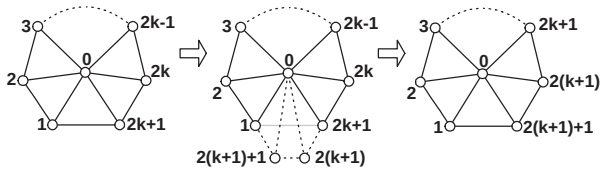

Figure 4.4: The $(2k+1)$-odd-wheel to the $(2(k+1)+1)$-odd-wheel.

can see that the topological difference between the $(2k+1)$-odd-wheel and the $\big(2(k+1)+1\big)$-odd-wheel is that the edge $\{1, 2k+1\}$ is lost, and the $2(k+1), 2(k+1)+1$ vertices are gained, along with associated edges

$$\Big\{\big(2k+1, 2(k+1)\big), \big(2(k+1), 2(k+1)+1\big), \big(1, 2(k+1)+1\big), \big(0, 2(k+1)\big), \big(0, 2(k+1)+1\big)\Big\}\,.$$

Suppose there exists an algebraic relation or syzygy of the specific form

$$\begin{aligned}\widetilde{\alpha}e_{\{1,2k+1\}} = \widetilde{\alpha}e_{\{1,2(k+1)+1\}} &+ \beta_{\{2k+1,2(k+1)\}}e_{\{2k+1,2(k+1)\}} \\ &+ \beta_{\{2(k+1),2(k+1)+1\}}e_{\{2(k+1),2(k+1)+1\}} + \beta_{\{01\}}e_{\{01\}} + \beta_{\{0,2k+1\}}e_{\{0,2k+1\}} \\ &+ \beta_{\{0,2(k+1)\}}e_{\{0,2k+1\}} + \beta_{\{0,2(k+1)+1\}}e_{\{0,2(k+1)+1\}}\ , \end{aligned} \tag{4.24}$$

where $\beta_{\{ij\}} \in \mathbb{C}[x_0, x_1, x_2, x_{2k+1}, x_{2(k+1)}, x_{2(k+1)+1}]$ and $\deg(\beta_{\{ij\}}) = 4$. Note that the coefficients for $e_{\{1,2k+1\}}$ and $e_{\{1,2(k+1)+1\}}$ are the same: both are equal to $\widetilde{\alpha}$. Therefore, in order to construct a degree four certificate for the $\big(2(k+1)+1\big)$-odd-wheel, we can simply substitute Eq. 4.24 for the $\widetilde{\alpha}e_{\{1,2k+1\}}$ term in Eq. 4.23. Thus, demonstrating the existence of a syzygy such as Eq. 4.24 will conclude our proof.

This special syzygy was indeed found explicitly via computer and it is listed below for the 3-odd-wheel to the 5-odd-wheel. For space considerations we do not list it for

general k, however it can be easily generalized to match the indices of Eq. 4.24 via the following variable substitutions: $x_3 \to x_{2k+1}$, $x_4 \to x_{2(k+1)}$, $x_5 \to x_{2(k+1)+1}$. Notice that $\widetilde{\alpha} \in \mathbb{C}[x_0, x_1, x_2]$. Therefore, $\widetilde{\alpha}$ is invariant under this substitution.

$$
\begin{aligned}
0 = &- \underbrace{\left(\frac{2}{9}x_1^4 + \frac{1}{9}x_1^3x_2 + \frac{1}{9}x_1^3x_0 + \frac{2}{9}x_1^2x_2x_0\right)}_{\widetilde{\alpha}} \underbrace{(x_1^2 + x_3x_1 + x_3^2)}_{e_{\{13\}}} \\
&+ \underbrace{\left(\frac{2}{9}x_1^4 + \frac{1}{9}x_1^3x_2 + \frac{1}{9}x_1^3x_0 + \frac{2}{9}x_1^2x_2x_0\right)}_{\widetilde{\alpha}} \underbrace{(x_1^2 + x_5x_1 + x_5^2)}_{e_{\{15\}}} \\
&+ \Big(\frac{2}{9}x_1^3x_0 + \frac{1}{9}x_1x_2x_0x_5 - \frac{1}{9}x_1x_2x_4x_5 - \frac{1}{9}x_1x_3x_0^2 - \frac{2}{9}x_1x_3x_0x_4 - \frac{2}{9}x_2x_0^3 \\
&- \frac{1}{9}x_2x_0^2x_4 + \frac{1}{9}x_4^4\Big) \underbrace{(x_3^2 + x_3x_4 + x_4^2)}_{e_{\{34\}}}
\end{aligned}
$$

$$
\begin{aligned}
&+ \Big(- \frac{2}{9}x_1^4 - \frac{2}{9}x_1^2x_2x_0 - \frac{1}{9}x_1^2x_2x_4 + \frac{1}{9}x_1^2x_0x_4 - \frac{1}{9}x_1x_2x_3x_0 + \frac{1}{9}x_1x_2x_3x_4 - \frac{1}{9}x_1x_2x_0^2 \\
&+ \frac{1}{9}x_1x_2x_4^2 - \frac{2}{9}x_0^4 + \frac{1}{9}x_0^3x_4 - \frac{1}{9}x_4^4 + \frac{1}{9}x_4^3x_5 - \frac{1}{9}x_4x_5^3\Big) \underbrace{(x_4^2 + x_4x_5 + x_5^2)}_{e_{\{45\}}}
\end{aligned}
$$

$$
\begin{aligned}
&+ \Big(- \frac{1}{3}x_1x_3x_0^2 - \frac{2}{9}x_3x_0x_4^2 - \frac{5}{9}x_1x_3^2x_0 - \frac{1}{3}x_1^2x_3x_0 + \frac{2}{9}x_1^2x_4x_5 + \frac{2}{9}x_0^2x_4x_5 \\
&+ \frac{1}{9}x_1^2x_2x_3 - \frac{1}{9}x_1^2x_2x_5 + \frac{2}{9}x_1^3x_3 - \frac{2}{9}x_1^3x_5 + \frac{1}{9}x_1^2x_0x_5 - \frac{2}{9}x_1^2x_0^2 + \frac{2}{9}x_1^2x_4^2 - \frac{4}{9}x_1x_3^2x_4 \\
&- \frac{5}{9}x_1x_0^2x_4 - \frac{4}{9}x_1x_0x_4^2 - \frac{1}{9}x_1x_0x_5^2 - \frac{1}{9}x_1x_4^2x_5 - \frac{2}{9}x_1x_0^3 + \frac{2}{9}x_2x_3^2x_0 + \frac{1}{9}x_2x_3^2x_4 \\
&- \frac{1}{9}x_1x_4x_5^2 + \frac{2}{9}x_3^2x_0x_4 + \frac{2}{9}x_2x_3x_4^2 - \frac{2}{3}x_1x_3x_0x_4 - \frac{4}{9}x_1x_0x_4x_5 - \frac{1}{9}x_2x_3x_5^2 \\
&+ \frac{2}{9}x_2x_0x_4^2 + \frac{1}{3}x_2x_3x_0x_4 - \frac{1}{9}x_2x_3x_0x_5 + \frac{1}{9}x_2x_4^3 \\
&- \frac{4}{9}x_3^3x_0 - \frac{1}{3}x_3^4 - \frac{1}{9}x_3^3x_4 + \frac{2}{9}x_3^2x_4^2 + \frac{2}{9}x_0^2x_5^2 - \frac{1}{9}x_0x_4^3\Big) \underbrace{(x_0^2 + x_0x_1 + x_1^2)}_{e_{\{01\}}}
\end{aligned}
$$

$$
\begin{aligned}
&+\Big(\frac{2}{9}x_1^4+\frac{1}{9}x_1^3x_2+\frac{4}{9}x_1^3x_0+\frac{4}{9}x_1^3x_4-\frac{1}{9}x_1^2x_2x_4+\frac{1}{3}x_1^2x_3^2+\frac{1}{9}x_1^2x_3x_0+\frac{1}{9}x_1^2x_3x_4+\frac{5}{9}x_1^2x_0^2\\
&-\frac{2}{9}x_1x_2x_0^2-\frac{1}{9}x_1x_2x_0x_4-\frac{1}{9}x_1x_2x_0x_5+\frac{1}{9}x_1x_2x_4x_5+\frac{1}{3}x_1x_3^2x_0+\frac{2}{9}x_1x_3x_0^2+\frac{1}{3}x_1x_3x_0x_4\\
&+\frac{1}{3}x_3^2x_0^2+\frac{5}{9}x_1^2x_0x_4+\frac{2}{9}x_1^2x_4^2-\frac{1}{9}x_3x_0^3-\frac{1}{9}x_3x_0^2x_4-\frac{2}{9}x_3x_0x_4^2-\frac{2}{9}x_0^4-\frac{2}{9}x_0^3x_4\Big)\underbrace{(x_0^2+x_0x_3+x_3^2)}_{e_{\{03\}}}\\
&+\Big(\frac{1}{9}x_1^3x_5-\frac{2}{9}x_1^2x_2x_3+\frac{1}{9}x_1^2x_2x_5-\frac{4}{9}x_1^2x_3^2-\frac{1}{9}x_1x_2x_3x_4+\frac{1}{9}x_1x_2x_0^2-\frac{1}{9}x_1x_2x_4^2+\frac{1}{9}x_1x_3x_0^2\\
&+\frac{1}{3}x_1x_0^3+\frac{1}{9}x_1x_0^2x_4+\frac{1}{9}x_1x_0^2x_5+\frac{1}{9}x_2x_3x_0x_5+\frac{1}{9}x_2x_3x_5^2+\frac{2}{9}x_3^2x_0+\frac{1}{9}x_3^2x_0x_4-\frac{1}{9}x_3^2x_4^2\\
&+\frac{2}{9}x_1x_3x_0x_4+\frac{1}{3}x_3x_0^3+\frac{1}{9}x_3x_0x_4^2-\frac{1}{9}x_3x_4^3+\frac{2}{9}x_0^4\Big)\underbrace{(x_0^2+x_0x_4+x_4^2)}_{e_{\{04\}}}\\
&+\Big(-\frac{1}{9}x_1^3x_2+\frac{1}{9}x_1^3x_4+\frac{1}{9}x_1^2x_2x_3+\frac{1}{9}x_1^2x_2x_4-\frac{1}{9}x_1^2x_0^2+\frac{2}{9}x_1x_2x_3x_0-\frac{1}{9}x_1x_2x_3x_4\\
&+\frac{1}{9}x_1x_2x_0^2-\frac{1}{9}x_1x_2x_4^2-\frac{1}{9}x_1x_0^3+\frac{1}{9}x_1x_0^2x_4-\frac{1}{9}x_2x_3x_0x_4\\
&-\frac{1}{9}x_2x_3x_4^2-\frac{1}{9}x_0x_4^2x_5-\frac{1}{9}x_0x_4x_5^2+\frac{1}{9}x_4^2x_5^2+\frac{1}{9}x_4x_5^3\Big)\underbrace{(x_0^2+x_0x_5+x_5^2)}_{e_{\{05\}}}\ .
\end{aligned}
$$

□

Lemma 4.3.12 *Using the encoding over $\overline{\mathbb{F}_2}$ presented in Lemma 2.2.6, the $(2k+1)$-odd-wheel has a minimum-degree Nullstellensatz certificate for non-3-colorability of degree one.*

Proof: We claim that the $(2k+1)$-odd-wheel has a Nullstellensatz certificate of the following form:.

$$
\begin{aligned}
1=&(x_1^3+1)+(x_2^3+1)+\cdots+(x_{2k+1}^3+1)\\
&+x_1(x_1^2+x_1x_2+x_2^2)+x_2(x_2^2+x_2x_3+x_3^2)+\cdots+x_{2k+1}(x_{2k+1}^2+x_{2k+1}x_1+x_1^2)\\
&+(x_2+x_{2k+1})(x_1^2+x_1x_0+x_0^2)+(x_1+x_3)(x_2^2+x_2x_0+x_0^2)\\
&+\cdots+(x_{2k}+x_1)(x_{2k+1}^2+x_{2k+1}x_0+x_0^2)\ . \qquad (4.25)
\end{aligned}
$$

Our proof is by direct verification. Note that this certificate has degree one, and is therefore minimal by Theorem 4.3.7. We begin our direct verification by observing that there are an odd number of polynomials of the form $(x_i^3 + 1)$: therefore, the sum of these "vertex" polynomials becomes $1 + x_1^3 + x_2^3 + \cdots + x_{2k+1}^3$. The "rim" polynomials tracing the odd-length cycle yield monomials of the form $x_i^3, x_i^2 x_{i+1}$ and $x_i x_{i+1}^2$, except for the last edge, which yields $x_{2k+1}^3, x_{2k+1}^2 x_1$ and $x_{2k+1} x_1^2$. Thus, the monomials x_i^3 have already cancelled. Finally, the "spokes" of the odd-wheel yield the following monomials: $x_i x_0^2$ (formed twice), $x_0 x_i x_{i+1}$ (formed twice), and $x_i^2 x_{i+1}, x_{i-1} x_i^2$ (each formed once). The spokes joining the center and vertices 1 and $2k + 1$ are slightly different: they also generate monomials $x_1^2 x_{2k+1}$ and $x_1 x_{2k+1}^2$, respectively. Thus, every monomial except the constant term cancels in the expanded certificate, and Eq. 4.25 simplifies to one. □

Lemmas 4.3.9, 4.3.10, 4.3.11 and 4.3.12 can be distilled into the following theorem:

Theorem 4.3.13 *Using the encodings presented in Lemmas 2.2.2 and 2.2.2 (over $\mathbb{C}$ and $\overline{\mathbb{F}_2}$, respectively), K_n with $n \geq 4$ has a minimum-degree Nullstellensatz certificate for non-3-colorability of degree four or one, respectively, and the $(2k+1)$-odd-wheel has a minimum-degree Nullstellensatz certificate for non-3-colorability of degree four or one, respectively.*

The importance of Theorem 4.3.13, in terms of computation, is to demonstrate that subgraphs with short, concise proofs of non-3-colorability can yield sparse, low-degree Nullstellensatz certificates. Although we do not understand the structure of non-3-colorability certificates in terms of a clear and direct relationship between the degree/coefficients and combinatorial properties of the underlying graph, as in the case with independent set (see

Theorem 4.2.8), we *conjecture*, based on our experimental results (see Table 5.3), that the certificates produced by **N**ul**LA** are certificates for somehow "minimal" 4-edge-critical subgraphs. Thus, we believe that the complexity of **N**ul**LA** for graph 3-colorability is defined by the complexity of the *most trivial* 4-edge-critical subgraph within a graph.

We also do not understand the combinatorial differences between non-3-colorable graphs with Nullstellensatz certificate degrees $1, 4$ or 7. The question of whether "hard" instances of graph 3-colorability have specific, identifiable, and systematically reproducible properties is an area of active research ([48, 15, 37, 60, 10]). We believe that a deeper understanding of these certificates may lead to results exposing concrete combinatorial properties that are linked to the "hardness" of 3-colorability. Being able to identify a property linked to hardness may lead to algorithms for creating hard instances of 3-colorability.

The difference between a Nullstellensatz certificate of degree one, and a Nullstellensatz certificate of degree four is stark and clear, whereas other measurements of hardness, such as running time or memory usage, may be more subtly ambiguous and less sharply delineated. This view of certificate degree as an indicator of the hardness of a 4-edge-critical subgraph is explored further in Section 5.2.6.

4.4 SAT and the Nullstellensatz

The polynomial encoding for SAT presented in Section 2.7 is the basis for a rich area of research on propositional proof systems (see [8, 26] and references therein). The seminal result from this area concerns a very simple Boolean formula known as the "induction"

principle:

$$\mathrm{IND}_n = x_1 \wedge (\neg x_1 \vee x_2) \wedge (\neg x_2 \vee x_3) \wedge \cdots \wedge (\neg x_{n-1} \vee x_n) \wedge \neg x_n \ .$$

By inspection, we can see that this Boolean formula is *not* satisfiable. It is known as the "induction" principle because x_1 is *true* and x_n is *false*; therefore, there must exist some point x_i where the variables change from *true* to *false*.

Theorem 4.4.1 (Buss and Pitassi [8]) *Given the Boolean formula IND_n encoded a system of polynomial equations via Lemma 2.7.1, the minimum degree d of its Nullstellensatz certificate is*

$$\lfloor \log_2(n) \rfloor - 1 \leq d \leq \lceil \log_2(n-1) \rceil \ .$$

From a computational perspective, this theorem is further evidence that "binary" encodings (encodings containing the equations $x_i(x_i - 1) = 0$ for all x_i) perform poorly with respect to computation with Hilbert's Nullstellensatz.

Chapter 5

Experimental Results

"Where a new invention promises to be
useful, it ought to be tried."
–Thomas Jefferson,
1743–1826 .

5.1 Four Mathematical Ideas to Optimize NulLA

In this section, we explore refinements and variations of **N**ul**LA** to improve performance on practical computational problems.

The main computational component of **N**ul**LA** is to construct and solve linear systems associated with Nullstellensatz certificates of increasing degree. These linear systems typically have millions of rows and columns, even for reasonably-sized problems with certificate degrees as low as four (see Section 5.2). Furthermore, the sizes of the linear systems increase dramatically with the degree of the certificate. In particular, the number of variables in a linear system associated with a Nullstellensatz certificate of degree d is precisely $s\binom{n+d}{d}$ where n is the number of variables in the polynomial system and s is the

number of polynomials. Note that $\binom{n+d}{d}$ is the number of possible monomials of degree d or less. Also, the number of non-zero entries in the constraint matrix is precisely $M\binom{n+d}{d}$ where M is the sum over the number of monomials in each polynomial of the system.

In this section, we explore mathematical approaches for solving the linear systems more efficiently, for decreasing the sizes of the linear systems associated with given degree, and for reducing the minimum degree required to produce a Nullstellensatz certificate.

It is certainly possible to significantly decrease the sizes of the linear systems by preprocessing, using the methods and techniques outlined in Section 4.3. However, those methods are specific to graph k-colorability, and cannot necessarily be extended to an arbitrary polynomial system. Furthermore, preprocessing alone is often not sufficient to enable us to solve some of the larger polynomial systems.

The mathematical ideas we explain in the rest of this section can be applied to arbitrary polynomial systems, but in some cases, a careful study of the structure of the polynomial system is required.

5.1.1 $\mathbb{C}$ vs. $\mathbb{Q}$ and $\overline{\mathbb{F}_p}$ vs. $\mathbb{F}_p$

In terms of computation and practical implementation, **N**ul**LA** over $\mathbb{C}$ or $\overline{\mathbb{F}_p}$ is obviously far more difficult to implement and far more time and memory-intensive to run than **N**ul**LA** over $\mathbb{Q}$ or $\mathbb{F}_p$. Fortunately, even though every encoding from Chapter 2 is proven with respect to the algebraically closed fields of $\mathbb{C}$ or $\overline{\mathbb{F}_p}$, the following lemma allows us compute over $\mathbb{Q}$ or $\mathbb{F}_p$.

Lemma 5.1.1 *Let $\mathbb{K}$ be a field and $\overline{\mathbb{K}}$ its algebraic closure. Given $f_1, \ldots, f_s \in \mathbb{K}[x_1, \ldots, x_n]$,*

there exists a Nullstellensatz certificate $1 = \sum \beta_i f_i$ *where* $\beta_i \in \overline{\mathbb{K}}[x_1, \ldots, x_n]$ *if and only if there exists a Nullstellensatz certificate* $1 = \sum \beta_i' f_i$ *where* $\beta_i' \in \mathbb{K}[x_1, \ldots, x_n]$.

Proof: If there exists a Nullstellensatz certificate $1 = \sum \beta_i f_i$ where $\beta_i \in \overline{\mathbb{K}}[x_1, \ldots, x_n]$, then, via **N**ul**LA**, we can construct the associated linear system of degree $\max\{\deg(\beta_i)\}$ and solve. Since $f_i \in \mathbb{K}[x_1, \ldots, x_n]$, the coefficients in the linear system will consist only of values in $\mathbb{K}$. Thus, solving the linear system relies only on computations in $\mathbb{K}$. When the free variables are chosen from $\mathbb{K}$ instead of $\overline{\mathbb{K}}$, the resulting Nullstellensatz certificate $1 = \sum \beta_i' f_i$ has $\beta_i' \in \mathbb{K}[x_1, \ldots, x_n]$.

Conversely, if there exists a Nullstellensatz certificate $1 = \sum \beta_i' f_i$ where $\beta_i' \in \mathbb{K}[x_1, \ldots, x_n]$, since $\mathbb{K} \subseteq \overline{\mathbb{K}}$, the same certificate is a certificate $1 = \sum \beta_i f_i$ where $\beta_i \in \overline{\mathbb{K}}[x_1, \ldots, x_n]$. □

Therefore, we have the following corollary:

Corollary 5.1.2 *A graph* G *is* non-*3-colorable if and only if there exists a Nullstellensatz certificate* $1 = \sum \beta_i f_i$ *where* $\beta_i \in \mathbb{F}_2[x_1, \ldots, x_n]$ *and the polynomials* $f_i \in \mathbb{F}_2[x_1, \ldots, x_n]$ *are as defined in Lemma 2.2.6.*

This corollary enables us to compute non-3-colorability over $\mathbb{F}_2$, which is extremely fast in practice (see Subsections 5.2.4 and 5.2.5).

5.1.2 Subgraph Equations as Degree-cutters

In Section 2.2, Lemma 2.2.8, we described a family of "subgraph equations" that can be added to the system of polynomial equations encoding graph k-colorability without

changing the set of solutions. In our experimental investigations, we discovered that *appending* these valid but redundant polynomial equations to the original system can *reduce* the degree of a minimum-degree Nullstellensatz certificate. A valid but redundant polynomial equation is any polynomial equation $g(x) = 0$ that is true for all the zeros of the polynomial system $f_1 = \cdots = f_s = 0$, i.e., $g \in \sqrt{I}$, the radical ideal of I, where I is the ideal generated by $f_1, ..., f_s$. In fact, we only really require that $g(x) = 0$ holds for at least one of zeros of the polynomial system $f_1 = \cdots = f_s = 0$, if a zero exists. We refer to a redundant polynomial equation appended to a system of polynomial equations, with the goal of reducing the degree of a Nullstellensatz certificate, as a *degree-cutter*.

For example, in the case of graph 3-colorability, we consider a triangle described by the vertices $\{x, y, z\}$. As we described in Lemma 2.2.8, we capture the additional requirement that each of these vertices must have a different color with the equation

$$x^2 + y^2 + z^2 = 0 \ , \tag{5.1}$$

which is satisfied if and only if $x \neq y \neq z \neq x$. We note that the equation $x + y + z = 0$ also implies $x \neq y \neq z \neq x$, but the homogenous of degree two version of this "triangle" equation closely mirrors the edge polynomials $x_i^2 + x_i x_j + x_j^2 = 0$ from the original system, which are likewise homogeneous of degree two. In order to reduce the minimum degree of a Nullstellensatz certificate, our experimental investigations have indicated that we should try to keep any additional degree-cutter equations added to the system as close as possible in degree and structure to the original polynomials of the system.

Consider the Koester graph [29] from Figure 5.1, a graph with 40 vertices and 80 edges. This graph has chromatic number four, and a corresponding minimum-degree

non-3-colorability certificate of degree four over $\mathbb{F}_2$. The size of the associated linear system required by **N**ul**LA** to produce this certificate was $8,724,468 \times 10,995,831$ and required 5 hours and 17 minutes of computation time.

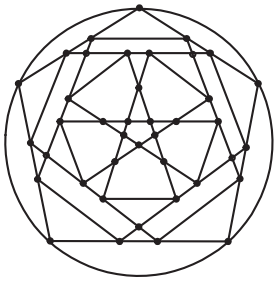

Figure 5.1: Koester graph

When we inspect the Koester graph in Figure 5.1, we can see that this graph contains 25 triangles. When we append these additional 25 triangle equations to the system of polynomial equations describing the graph, the degree of the Nullstellensatz certificate drops from four to one. Therefore, despite appending an additional 25 equations to the system, **N**ul**LA** only needs to solve a $4,626 \times 4,346$ linear system to produce a degree one certificate, which takes 0.2 seconds of computation time. Note that even though we have *appended* equations to the system of polynomial equations, because the degree of the overall certificate is *reduced*, the size of the resulting linear system is still much, much smaller.

As we noted in Lemma 2.2.8, these subgraph equations can be extended to k-colorability, and any degree d can be used as long as $d \not| k$. The minimum-degree non-5-colorability Nullstellensatz certificate of K_6 over $\mathbb{F}_2$ has degree six. However, in K_6, every subset of five vertices is isomorphic to K_5; thus, every subset of five vertices must have each vertex colored differently. We capture this additional requirement by adding

five "K_5-equations" of degree four. For example, given the complete subgraph formed by $\{x_0, x_1, x_2, x_3, x_4\}$, we add the following equation:

$$x_0^4 + x_1^4 + x_2^4 + x_3^4 + x_4^4 = 0 \ .$$

These "K_5-equations" are homogenous of degree four, and they closely match the degree and structure of the edge polynomials $x_i^4 + x_i^3 x_j + x_i^2 x_j^2 + x_i x_j^3 + x_j^4 = 0$. When these five equations are appended to the original system, the degree of the Nullstellensatz certificate drops from six to one.

However, as we will see in Subsection 5.1.3, the triangle degree-cutter equations for 3-colorability (5.1) are not always sufficient to reduce the degree of the Nullstellensatz. The difficulty with the generalized degree-cutter approach is in finding candidate degree-cutter equations, and in determining how many of the candidate degree-cutters to append to the system. There is an obvious trade-off between the time spent finding degree-cutter equations (and the increased size of the linear systems involved due to the addition of the degree-cutter equations), as compared to the benefit of reducing the degree of the Nullstellensatz certificate.

5.1.3 Alternative Nullstellensätze

A second approach to reducing the minimum degree of a Nullstellensatz certificate is to find an *alternative* Nullstellensatz certificate.

Corollary 5.1.3 (Alternative Nullstellensatz) *Let $\mathbb{K}$ be an algebraically closed field. A system of polynomial equations $f_1 = \cdots = f_s = 0$ where $f_i \in \mathbb{K}[x_1, \ldots, x_n]$ has no solution*

in $\mathbb{K}^n$ *if and only if there exist polynomials* $\beta_1, \ldots, \beta_s \in \mathbb{K}[x_1, \ldots, x_n]$ *and* $g \in \mathbb{K}[x_1, ..., x_n]$ *such that* $g = \sum \beta_i f_i$ *and the system* $f_1 = \cdots = f_s = g = 0$ *has no solution.*

Hilbert's Nullstellensatz is a special case of this alternative Nullstellensatz when $g(x) = 1$. The importance of these alternative Nullstellensatz in terms of practical computation is that the minimum degree of an alternative Nullstellensatz certificate may be lower than the minimum degree of an ordinary Nullstellensatz certificate for a given system of polynomial equations.

Example 5.1.4 When testing for non-3-colorability over $\mathbb{F}_2$, the graph in Figure 5.2 has a minimum degree Nullstellensatz certificate of degree four. This graph also has three triangles: $\{x_1, x_2, x_6\}, \{x_2, x_5, x_6\}$ and $\{x_2, x_6, x_7\}$. However, in this case, these three triangle equations were not sufficient to reduce the degree: we appended these equations to the system of polynomial equations, and the minimum degree Nullstellensatz certificate remained four. However, when we searched for a degree one *alternative* Nullstellensatz certificate, we were able to find a certificate with $g(x) = x_1 x_8 x_9$:

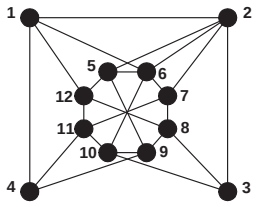

Figure 5.2: A graph with a degree four non-3-colorability certificate over $\mathbb{F}_2$.

$$
\begin{aligned}
x_1x_8x_9 = {} & (x_1+x_2)(x_1^2+x_1x_2+x_2^2)+(x_4+x_9+x_{12})(x_1^2+x_1x_4+x_4^2) \\
& +(x_1+x_4+x_8)(x_1^2+x_1x_{12}+x_{12}^2)+(x_2+x_7+x_8)(x_2^2+x_2x_3+x_3^2) \\
& +(x_3+x_8)(x_2^2+x_2x_7+x_7^2)+(x_{10}+x_{12})(x_4^2+x_4x_{11}+x_{11}^2) \\
& +(x_1+x_4+x_{10})(x_4^2+x_4x_9+x_9^2)+(x_2+x_7+x_8)(x_3^2+x_3x_8+x_8^2) \\
& +(x_2+x_{10})(x_5^2+x_5x_6+x_6^2)+(x_5+x_{10})(x_5^2+x_5x_9+x_9^2) \\
& +(x_2+x_3+x_{12})(x_7^2+x_7x_8+x_8^2)+(x_1+x_7+x_8)(x_8^2+x_8x_{12}+x_{12}^2) \\
& +(x_2+x_{10})(x_6^2+x_6x_7+x_7^2)+(x_{10}+x_{12})(x_7^2+x_7x_{11}+x_{11}^2) \\
& +(x_5)(x_2^2+x_2x_5+x_5^2)+(x_5+x_7)(x_6^2+x_6x_{10}+x_{10}^2) \\
& +(x_4+x_7)(x_{10}^2+x_{10}x_{11}+x_{11}^2)+(x_4+x_5)(x_9^2+x_9x_{10}+x_{10}^2) \\
& +(x_1)(x_8^2+x_8x_9+x_9^2)+(x_4+x_7)(x_{11}^2+x_{11}x_{12}+x_{12}^2)+(x_5+x_7)(x_2^2+x_2x_6+x_6^2) \\
& +(x_8+x_9)\underbrace{(x_1^2+x_2^2+x_6^2)}_{\text{degree-cutter}}+(x_9)\underbrace{(x_2^2+x_5^2+x_6^2)}_{\text{degree-cutter}}+(x_8)\underbrace{(x_2^2+x_6^2+x_7^2)}_{\text{degree-cutter}}.
\end{aligned}
$$

We note $g(x) = x_1x_8x_9$ was not the only alternative Nullstellensatz certificate that we were able to find: $g(x) = x_7x_4x_9$ also produced a certificate. □

Nul$\mathbf{g}$**LA** can easily be adapted to construct alternative Nullstellensatz certificates if the polynomial g is specified as part of the input. In the case of graph k-colorability, any non-trivial monomial is a possible g, because the equations $x_i^3 - 1 = 0$ force every x_i to assume the value of a root of unity, and $g(x) = 0$ implies that $x_i = 0$ for some variable x_i. Thus, in the case of graph k-colorability, **Nul**$\mathbf{g}$**LA** can easily be adapted to search for alternative Nullstellensatz certificates of a given degree, along with ordinary Hilbert's Nullstellensatz certificates. For example, for the graph in Figure 5.2, we searched for alternative Nullstellensatz certificates of degree one by enumerating the set of all possible

monomials of degree three. Since choosing different $g(x)$ only means changing the constant terms of the **N**ul**LA** linear system (the other coefficients remain the same), enumerating a set of possible $g(x)$ can be accomplished efficiently.

5.1.4 Probabilistic Nullstellensätze

The systems of linear equations produced by **N**ul**LA** are quite large in practice, even for degrees as low as four. Subsections 5.1.3 and 5.1.2 describe ideas for reducing the *degree* of Nullstellensatz certificates, and thus reducing the size of the **N**ul**LA** linear systems, and Section 4.3 describes many lemmas that allow us to eliminate monomials from non-k-colorability certificates, which reduces the number of unknowns in the associated linear systems in the case of graph k-colorability. In this section, we describe a probabilistic approach which applies to arbitrary encodings and further reduces the number of unknowns: instead of allowing *all* monomials of degree d to appear in the Nullstellensatz certificate, we can randomly set coefficients of some monomials to zero— e.g., independently set variables to zero with probability p. Consider the following example:

Example 5.1.5 As in Example 3.2.1, consider the input system of polynomial equations $x_1^2 - 1 = 0, x_1 + x_2 = 0, x_1 + x_3 = 0, x_2 + x_3 = 0$. This system has *no* solution, and in Example 3.2.1, we discovered that it has a Nullstellensatz certificate of degree one. As before, we begin by constructing a certificate of degree one, with unknowns for coefficients,

but this time, we arbitrary set unknowns to zero.

$$1 = \underbrace{(c_0x_1 + c_1x_2 + 0 \cdot x_3 + c_2)}_{\beta_1}\underbrace{(x_1^2 - 1)}_{f_1} + \underbrace{(c_3x_1 + 0 \cdot x_2 + c_4x_3 + 0)}_{\beta_2}\underbrace{(x_1 + x_2)}_{f_2}$$

$$+ \underbrace{(c_5x_1 + 0 \cdot x_2 + 0 \cdot x_3 + c_6)}_{\beta_3}\underbrace{(x_1 + x_3)}_{f_3} + \underbrace{(c_7x_1 + 0 \cdot x_2 + c_8x_3 + 0)}_{\beta_4}\underbrace{(x_2 + x_3)}_{f_4}.$$

Thus, we have reduced the number of unknowns from 16 (in Example 3.2.1) to 9, and the resulting Nullstellensatz certificate remains unchanged.

$$1 = \frac{1}{2}x_1(x_1 + x_2) - \frac{1}{2}x_1(x_2 + x_3) + \frac{1}{2}x_1(x_1 + x_3) - (x_1^2 - 1)\ .$$

□

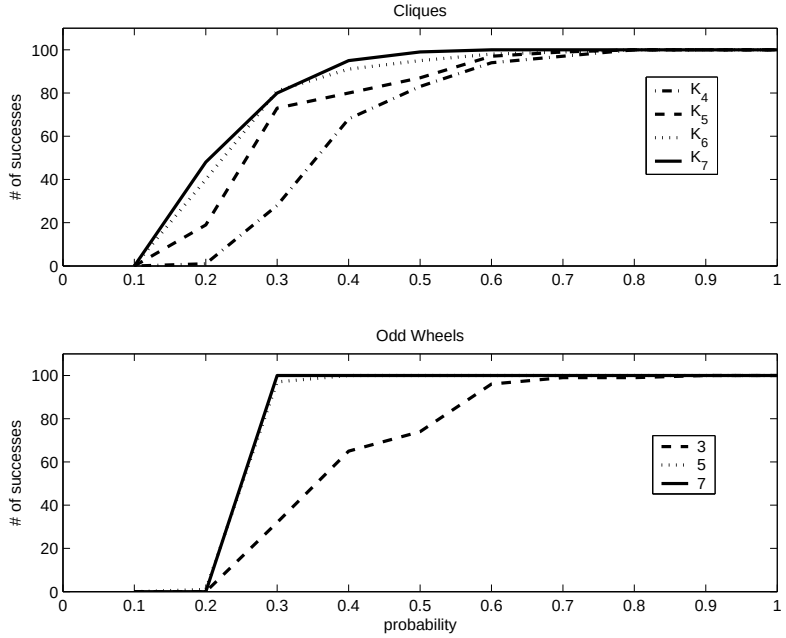

Figure 5.3: Probability tests on cliques and odd-wheels over $\mathbb{Q}$.

This heuristic works quite well in the case of graph 3-colorability over $\mathbb{Q}$, where any minimum degree Nullstellensatz certificate has degree at least four. In Figure 5.3, we

see the results of a probabilistic search for non-3-colorability Nullstellensatz certificates for cliques and odd-wheels. The probability p of keeping an unknown in the linear system appears on the x-axis. Thus, if $p = 0.1$, 90% of the time we set the unknown to 0, and 10% of the time we keep it in the system. For cliques and odd-wheels, we know that there is always a certificate of degree four. For every probability $0.1, 0.2, \ldots, 1$ we performed 100 searches for a degree four certificate. For the cliques and odd-wheels at $p = 0.1$ and $p = 0.2$, we almost never found certificates. But for $p = 0.4$, we found certificates 95% of the time. In practice, we can reduce the number of variables in the linear system by 60%, and still find a Nullstellensatz certificate 90% of the time.

However, the results of graph 3-colorability over $\mathbb{F}_2$ are somewhat less favorable. Figures 5.4 and 5.5 both illustrate the results of probabilistic searches for non-3-colorability Nullstellensatz certificates over $\mathbb{F}_2$ for odd-wheels, Kneser, flowers and cat-ear graphs (all graphs are described in Subsection 5.2.2). The odd-wheel and Kneser graphs tested both have certificates of degree one. However, the flower and cat-ear graphs tested have no degree one certificates unless degree-cutter triangle equations are applied; thus, the certificates tested for these graphs are degree four *without* degree-cutter equations.

There is an interesting discrepancy between the probabilistic results of the degree four and degree one certificates. For the odd-wheels and their degree one certificates, the linear systems are quite dense. When $p = .9$ (90% of the unknowns are present), the probability of success is only about 50%. By contrast, for the cat-ears graphs and their degree four certificates, when $p = .5$, the probability of success is over 80%. The Kneser graphs (also with degree one certificates) have an 80% success rate when $p = .7$,

and the flowers (degree four certificates) have an 80% success rate when $p = .4$. The probability results for the degree four certificates are consistent with the results reported over $\mathbb{Q}$. However, the $\mathbb{F}_2$ degree one certificates seem to reliably require somewhat more dense linear systems.

5.2 Graph 3-colorability Experimental Results

In this section, we present our experimental results for graph 3-colorability. We describe our software and testing platform (Subsection 5.2.1), our test cases (Subsection 5.2.2), and then we present our results over both $\mathbb{Q}$ and $\mathbb{F}_2$ (Subsections 5.2.3 and 5.2.4, respectively). We also compare **N**ul**LA** to other algebraic methods (Subsection 5.2.5), and explore supposed "hard instances" of 3-colorability (Subsection 5.2.6).

Our experiments over $\mathbb{F}_2$ were surprisingly successful. We were able to compute the non-3-colorability of graphs with almost 2000 vertices and tens of thousands of edges.

5.2.1 Methods

The computationally-intensive aspect of **N**ul**LA** is constructing and solving the linear systems associated with a given degree. Towards that end, we implemented an exact-arithmetic linear system solver in C++ that works over $\mathbb{Q}$ and $\mathbb{F}_p$. Early on, we observed that the systems of linear equations were numerically unstable in floating point arithmetic, and we were thus forced to write our own back-end solver. Our computations were performed on machines with dual Opteron nodes, 2 GHz clock speed, and 12 GB of RAM. No degree-cutter equations or alternative Nullstellensatz certificates were used, and we preprocessed

the linear systems according to the lemmas presented in Section 4.3. However, when testing over $\mathbb{Q}$, we extensively used probabilistic Nullstellensätze; thus, the value p appearing in the tables is the probability (most often .4, as described in Subsection 5.1.4).

5.2.2 Test Cases

We tested the following graphs:

1. **DIMACS:** The graphs from the DIMACS Computational Challenge (1993, 2002) are described in detail at `http://mat.gsia.cmu.edu/COLORING02/`. This set of graphs is the standard benchmark for graph coloring algorithms. We tested every DIMACS graph whose associated **N**ul**LA** matrix could be instantiated within 12 GB of RAM. For example, we did *not* test `C4000.5.clq`, which has 4,000 vertices and 4,000,268 edges, yielding a degree one **N**ul**LA** matrix of 758 million non-zero entries and 1 trillion columns.

2. **Mycielski:** The Mycielski graphs are known for the gap between their clique and chromatic number. The Mycielski graph of order k is a triangle-free graph with chromatic number k. The first few instances and the algorithm for their construction can be seen at `http://mathworld.wolfram.com/MycielskiGraph.html`.

3. **Kneser:** The nodes of the Kneser-(t, r) graph are represented by the $\binom{t}{r}$ r-subsets of $\{1, \ldots, t\}$. Two nodes are adjacent if and only if their subsets are disjoint.

4. **Flowers:** These graphs are pictured in Figure 5.6. Note that the 3-flower is 3-colorable, whereas the 4 and 5 flowers are non-3-colorable. It is easy to see that

(0 mod 3)-flowers are 3-colorable, whereas the (1 mod 3) or (2 mod 3)-flowers are non-3-colorable.

5. **Cat-ears:** These graphs are pictured in Figure 5.7. They are an infinite family of connected near-4-cliques (K_4 with one edge removed), and each graph is non-3-colorable.

6. **Random:** We tested random graphs in 16 nodes with an edge probability of .27. This probability was experimentally selected based on the boundary between 3-colorable and non-3-colorable graphs and is explained in detail in Section 5.2.4.

7. **Other graphs:** We also tested many, many other graphs, simply because they were non-3-colorable. For example, a *uniquely 3-colorable* graph is a graph that can be colored with three colors in only one way, up to permutation of the color labels. Figure 5.8 displays a uniquely 3-colorable, triangle-free graph [9]. Because the graph is uniquely 3-colorable, the addition of a single edge between two similarly-colored vertices will result in a new non-3-colorable graph. Also pictured in Figure 5.8 are the Grötzch graph and the Jin graph [28], both of which have chromatic number four.

5.2.3 Experimental Results over $\mathbb{Q}$

Our graph 3-colorability experiments over $\mathbb{Q}$ were motivated by trying to find the infinite family of non-3-colorable graphs with growth in the degree of their associated Nullstellensatz certificates: in particular, the infinite family of graphs proposed in Corollary 4.1.2. At this stage in our project, our goal was not an efficient algorithm for practical computation; rather, it was an attempt to explore the degree and structure of Nullstellen-

satz certificates. We tested hundreds of non-3-colorable graphs, hoping to find an explicit example with growth in the certificate degree. However, every graph that we tested had a Nullstellensatz certificate of degree four. In Table 5.1, we present a sampling of the many graphs we tried during this stage of our computational experiments. Note that the graph UNQ-3-CLR is the uniquely 3-colorable graph from Figure 5.8, with various edges added. We also tested every non-3-colorable graph with six vertices or less: every graph had a Nullstellensatz certificate of degree four.

Our experimental investigations over $\mathbb{Q}$ led us to the following conclusion: not only was growth in the degree of non-3-colorability Nullstellensatz certificates rare for small graphs, but low-degree certificates were common. This caused us to reconsider the possibility of using **N**ul**LA** for practical computation.

5.2.4 Experimental Results over $\mathbb{F}_2$

In this subsection, we describe our experimental investigations of graph 3-colorability over $\mathbb{F}_2$. To summarize, almost all of the graphs tested by **N**ul**LA** over $\mathbb{F}_2$ had degree *one* certificates. This algebraic property, coupled with our ability to compute over $\mathbb{F}_2$, allowed us to prove the non-3-colorability of graphs with almost two thousand nodes.

Although testing for graph 3-colorability is well-known to be NP-complete, there exist many efficient (and even trivial), polynomial-time algorithms for finding 4-cliques in a graph. Because we are now interested in practical computation, we break our computational investigations into two tables: Table 5.2 contains graphs *without* 4-cliques, and Table 5.4 contains graphs *with* 4-cliques (considered "easy" instances of 3-colorability). For space considerations, we only display representative results for graphs of various sizes for

Graph	*vertices*	*edges*	*row*	*col*	p	deg
flower 8	16	32	51,819	49,516	.4	4
flower 10	20	40	178,571	362,705	1	4
flower 11	22	44	278,737	278,844	.5	4
flower 13	26	52	629,666	495,051	.4	4
flower 14	28	56	923,580	705,536	.4	4
flower 16	32	64	1,979,584	1,674,379	.4	4
flower 17	34	68	2,719,979	2,246,535	.4	4
flower 19	38	76	4,862,753	3,850,300	.5	4
kneser-(6,2)	15	45	39,059	68,811	.5	4
kneser-(7,2)	21	105	230,861	558,484	.5	4
kneser-(8,2)	28	210	1,107,881	3,307,971	.5	4
kneser-(9,2)	36	378	1,107,955	3,304,966	.5	4
kneser-(10,2)	45	630	15,567,791	36,785,283	.5	4
jin graph	12	24	12,168	13,150	.4	4
Grötzsch	11	20	7,903	8,109	.4	4
UNQ-3-CLR + $\{(3,4)\}$	12	24	12,257	13,091	.4	4
UNQ-3-CLR + $\{(7,12)\}$	12	24	12,201	13,085	.4	4
UNQ-3-CLR + $\{(1,8)\}$	12	24	12,180	13,124	.4	4
UNQ-3-CLR + $\{(3,4),(12,7)\}$	12	25	12,286	13,804	.4	4

Table 5.1: Experimental investigations of graph 3-colorability over $\mathbb{Q}$.

each family. We also point out certain properties of **N**ul**LA**-constructed certificates, and conclude with tests on random graphs. Surprisingly, all but four of the DIMACS, Mycielski and Kneser graphs tested with **N**ul**LA** have degree one certificates.

Not all of the DIMACS challenge graphs had degree one certificates. We were not able to produce certificates for `mug88_1`, `mug88_25`, `mug100_1` or `mug100_25`, even when using degree-cutters and searching for alternative Nullstellensatz certificates. When testing for a degree four certificate, the smallest of these graphs (`mug88_1` with 88 vertices and 146 edges) yielded a linear system with 1,170,902,966 non-zero entries and 390,340,149 columns. A matrix of this size is not computationally tractable at this time because it cannot be instantiated within available memory.

Graph	*vertices*	*edges*	*rows*	*cols*	*deg*	*sec*
m7 (Mycielski 7)	95	755	64,281	71,726	1	.46
m9 (Mycielski 9)	383	7,271	2,477,931	2,784,794	1	268.78
m10 (Mycielski 10)	767	22,196	15,270,943	17,024,333	1	14835
$(8,3)$-Kneser	56	280	15,737	15,681	1	.07
$(10,4)$-Kneser	210	1,575	349,651	330,751	1	3.92
$(12,5)$-Kneser	792	8,316	7,030,585	6,586,273	1	466.47
$(13,5)$-Kneser	1,287	36,036	45,980,650	46,378,333	1	216105
ash331GPIA.col	662	4,185	3,147,007	2,770,471	1	13.71
ash608GPIA.col	1,216	7,844	10,904,642	9,538,305	1	34.65
ash958GPIA.col	1,916	12,506	27,450,965	23,961,497	1	90.41
1-Insertions_5.col	202	1,227	268,049	247,855	1	1.69
2-Insertions_5.col	597	3,936	2,628,805	2,349,793	1	18.23
3-Insertions_5.col	1,406	9,695	15,392,209	13,631,171	1	83.45

Table 5.2: Graphs without 4-cliques over $\mathbb{F}_2$.

Recall that the Nullstellensatz certificates returned by **N**ul**LA** consist of a single vertex polynomial (via preprocessing), and edge polynomials describing either the original graph in its entirety, or a non-3-colorable subgraph of the original graph. For example, if the graph contains a 4-clique as a subgraph, often the Nullstellensatz certificate will only display the edges contained in the 4-clique. In this case, we say that **N**ul**LA** *isolates* a non-3-colorable subgraph from the original graph. The size difference between these subgraphs and the input graphs is often dramatic, as shown in Table 5.3.

An overall analysis of these computational experiments shows that **N**ul**LA** performs best on sparse graphs. For example, the `3-Insertions_5.col` graph (with 1,406 nodes and 9,695 edges) runs in 83 seconds, while the `3-FullIns_5.col` graph (with 2,030 nodes and 33,751 edges) runs in 15027 seconds. Another example is `p_hat700-2.clq` (with 700 nodes and 121,728 edges) and `will199GPIA.col` (with 701 nodes and 7,065 edges). **N**ul**LA** proved the non-3-colorability of `will199GPIA.col` in 35 seconds, while

Graph	*vertices*	*edges*	*subgraph vertices*	*subgraph edges*
miles1500.col	128	10,396	6	10
hamming8-4.clq	256	20,864	19	33
m10 (Mycielski 10)	767	22,196	11	20
$(12, 5)$-Kneser	792	8,316	53	102
dsjc1000.1.col	1,000	49,629	15	24
ash608GPIA.col	1,216	7,844	23	44
3-Insertions_5.col	1,406	9,695	56	110
ash958GPIA.col	1,916	12,506	24	45

Table 5.3: Original graph vs. non-3-colorable subgraph.

`p_hat700-2.clq` took 30115 seconds.

Finally, as an informal measure of the distribution of degree one certificates, we generated random graphs of 16 nodes with edge probability .27. We selected this probability because it lies on the boundary between feasible and infeasible instances. In other words, graphs with edge probability less than .27 were almost always 3-colorable, and graphs with edge probability greater than .27 were almost always non-3-colorable. However, we experimentally found that an edge probability of .27 created a distribution that was almost exactly half and half. Of 100 trials, 48 were infeasible. Of those 48 graphs, 40 had degree one certificates and 8 had degree four certificates. Of these remaining 8 instances, we were able to find degree one certificates for all 8 by appending degree-cutters or by finding alternative Nullstellensatz certificates. This tentative measure indicates that non-3-colorability certificates of degrees greater than one may be rare.

Graph	*vertices*	*edges*	*rows*	*cols*	*deg*	*sec*
miles500.col	128	2,340	143,640	299,521	1	1.35
miles1000.col	128	6,432	284,042	823,297	1	7.52
miles1500.col	128	10,396	349,806	1,330,689	1	24.23
mulsol.i.5.col	197	3,925	606,959	773,226	1	6
zeroin.i.1.col	211	4,100	643,114	865,101	1	6
queen16_16.col	256	12,640	1,397,473	3,235,841	1	106
hamming8-4.clq	256	20,864	2,657,025	5,341,185	1	621.1
school1_nsh.col	352	14,612	4,051,202	5,143,425	1	210.74
MANN_a27.clq	378	70,551	9,073,144	26,668,279	1	9809.22
brock400_4.clq	400	59,765	10,579,085	23,906,001	1	4548.59
gen400_p0.9_65.clq	400	71,820	10,735,248	28,728,001	1	9608.85
le450_5d.col	450	9,757	4,168,276	4,390,651	1	304.84
fpsol2.i.1.col	496	11,654	4,640,279	57,803,85	1	93.8
C500.9.clq	500	112,332	20,938,304	56,166,001	1	72752
homer.col	561	3,258	1,189,065	1,827,739	1	8
p_hat700-2.clq	700	121,728	48,301,632	85,209,601	1	30115
will199GPIA.col	701	7,065	5,093,201	4,952,566	1	35
inithx.i.1.col	864	18,707	13,834,511	16,162,849	1	1021.76
qg.order30.col	900	26,100	23,003,701	23,490,001	1	13043
wap06a.col	947	43,571	37,703,503	41,261,738	1	1428
dsjc1000.1.col	1,000	49,629	45,771,027	49,629,001	1	2981.91
5-FullIns_4.col	1,085	11,395	13,149,910	12,363,576	1	200.09
3-FullIns_5.col	2,030	33,751	70,680,086	68,514,531	1	15027.9

Table 5.4: Graphs with 4-cliques over $\mathbb{F}_2$.

5.2.5 NulLA over $\mathbb{F}_2$ vs. other graph coloring algorithms

In this subsection, we compare **N**ul**LA** over $\mathbb{F}_2$ to two other methods for detecting 3-colorability; the Alon-Tarsi (AT) method, and the Gröbner basis (GB) method. We also briefly compare **N**ul**LA** to the two well-known graph coloring heuristics DSATUR and Branch-and-Cut [46]. We implemented the Alon-Tarsi method in C++, and used CoCoA Lib [11] to test the Gröbner basis method. For brevity, we do not record any "internal data" about the various algorithmic runs, such as the size of the underlying linear systems solved by **N**ul**LA** or the maximum number of monomials in the normal forms pro-

duced by the Alon-Tarsi method. In the tables below, all certificates have degree one and a "–" signifies that the method was terminated after four hours of computation.

The Gröbner basis method simply refers to taking the Gröbner basis of the ideal defined in Lemma 2.2.6. By Hilbert's Nullstellensatz, the Gröbner basis is a constant if and only if the graph is non-3-colorable.

The Alon-Tarsi method is based on the following (see Section 7 of [1] and references therein):

Theorem 5.2.1 *Given a graph G with n vertices, let $I_G = \langle x_1^3-1, \ldots, x_n^3-1\rangle$. Additionally, let*

$$P_G = \prod_{\{i,j\}\in E(G)} (x_i - x_j)$$

Then $P_G \in I_G$ if and only if G is non-3-colorable

In order to compute with the Alon-Tarsi method, we note that the set $B = \{x_1^3 - 1, \ldots, x_n^3 - 1\}$ is a Gröbner basis for I_G. Thus, we simply take the normal form of P_G with respect to B. If the normal form is zero, $P_G \in I_G$, and the graph is non-3-colorable. The efficiency of the Alon-Tarsi method can be increased by incrementally constructing P_G [24]: we order the edges, and then find the normal form of $(x_{i_1} - x_{j_1})$ with respect to B, and then the normal form of $(x_{i_1} - x_{j_1})(x_{i_2} - x_{j_2})$ with respect to B, etc.

We compared **N**ul**LA** to the Gröbner basis and Alon-Tarsi methods on graphs with and without 4-cliques; results are displayed in Tables 5.7 and 5.8, respectively.

Nul**LA** consistently out-performed the Gröbner basis method. For example, on `zeroin.i.1`, **N**ul**LA** ran in 6 seconds, while CoCoA Lib took almost one hour. These ex-

perimental results indicate that **N**ul**LA** scales more efficiently with respect to input size than the Gröbner basis method.

Nul**LA** also compared extremely favorably with the Alon-Tarsi method, which usually did not terminate within the requisite time bounds. However, in the special case when the first few vertices and edges of the graph happen to describe a non-3-colorable subgraph (such as a 4-clique, or the Grötzch graph), the Alon-Tarsi method ran very quickly, because of the iterative approach incorporated during implementation. Consider the example of the ninth Mycielski graph (383 vertices and 7,271 edges): the Alon-Tarsi method terminated in .24 seconds. However, after we permuted the vertices and edges, the method did not terminate. Indeed, after consuming 9 GB of RAM over 4 hours of computation, the method had only processed 30 out of 7,271 edges. This example shows that the Alon-Tarsi method is extremely sensitive to the vertex and edge ordering. If a similar iterative approach was incorporated either into **N**ul**LA** or the Gröbner basis method, these algorithms would likewise terminate early in this special case.

As another example of the draw-backs of the Alon-Tarsi method, we considered edge-critical graphs, where the entire input must be read. For example, the odd-wheels form a trivial family of edge-critical non-3-colorable graphs. The Alon-Tarsi method was unable to determine the non-3-colorability of the 17-odd-wheel (18 vertices and 34 edges): after two hours of computation, the normal form contained over 19 million monomials, and had consumed over 8 GB of RAM. The experimental results are displayed in Table 5.5.

We conclude with a short comment comparing **N**ul**LA** to DSATUR and Branch-and-Cut [46]. These heuristics return bounds on the chromatic number. In Table 5.6 (data

odd-wheels	*vertices*	*edges*	***N**ul**LA***	*GB*	*AT*
9	10	18	0	0	.05
11	12	22	0	0	.74
13	14	26	0	0	8.47
15	16	30	0	0	369.45
17	18	34	0	0	–
151	152	302	.21	2.21	–
501	502	1,002	15.58	126.83	–
1001	1,002	2,002	622.73	1706.69	–
2001	2,002	4,002	12905.6	–	–

Table 5.5: **N**ul**LA**, GB and AT on odd-wheel graphs.

taken from [46]), we display the bounds returned by Branch-and-Cut (B&C) and DSATUR, respectively. We do not include running times for B&C and DSATUR, because the authors did not report running times in [46]; both algorithms exceeded the two hour computation time limit set by the authors. By contrast, in the case of these graphs, **N**ul**LA** determined non-3-colorability very rapidly (establishing a lower bound of four), while the two heuristics returned lower bounds of three and two, respectively. Thus, **N**ul**LA** returned a tighter lower bound on the chromatic number than B&C or DSATUR.

			B&C		DSATUR		**N**ul**LA**
Graph	*vertices*	*edges*	*lb*	*up*	*lb*	*up*	*sec*
4-Insertions_3	79	156	3	4	2	4	0
3-Insertions_4	281	1,046	3	5	2	5	1
4-Insertions_4	475	1,795	3	5	2	5	3
2-Insertions_5	597	3,936	3	6	2	6	12
3-Insertions_5	1,406	9,695	3	6	2	6	83

Table 5.6: **N**ul**LA** vs. Branch-and-Cut and DSATUR.

Graph	*vertices*	*edges*	***NulLA***	*GB*	*AT*
miles500	128	2,340	1.35	133.91	.07
miles1000	128	6,432	7.52	802.23	0
miles1500	128	10,396	24.23	2598.84	.01
mulsol.i.5	197	3,925	6	18804.5	0
zeroin.i.1	211	4,100	6	2753.37	0
queen16_16	256	12,640	106	59466.9	0
hamming8-4	256	20,864	621.1	–	–
le450_5d	450	9,757	304.84	–	–
homer	561	3,258	8	–	–
dsjc1000.1	1,000	49,629	2981.91	–	–
5-FullIns_4	1,085	11,395	200.09	–	557.12
3-FullIns_5	2,030	33,751	15027.9	–	3.97

Table 5.7: **N**ul**LA**, GB, AT on graphs with 4-cliques.

Graph	*vertices*	*edges*	***NulLA***	*GB*	*AT*
Mycielski 4	11	20	0	.01	.22
Mycielski 5	23	71	0	.08	.23
Mycielski 6	47	236	.04	3.99	.22
Mycielski 7	95	755	.46	179.94	.23
Mycielski 8	191	2,360	7.72	9015.06	.23
Mycielski 9	383	7,271	268.78	–	.22
Mycielski 9 permuted	383	7,271	497.47	–	–
$(6, 2)$-Kneser	15	45	0	.03	1.87
$(8, 3)$-Kneser	56	280	.07	18.39	–
$(10, 4)$-Kneser	210	1,575	3.92	9771.76	–
$(12, 5)$-Kneser	792	8,316	466.47	–	–
ash331GPIA	662	4,185	13.71	–	–
1-Insertions_4	67	232	.04	3.71	–
2-Insertions_4	149	541	.26	32.42	–
1-Insertions_5	202	1,227	1.69	940.7	–
3-Insertions_4	281	1,046	.97	237.69	–
4-Insertions_4	475	1,795	3.02	1596.35	–
2-Insertions_5	597	3,936	18.23	–	–

Table 5.8: **N**ul**LA**, GB, AT on graphs without 4-cliques.

5.2.6 Hard Instances of 3-colorability

The question of whether "hard" instances of graph 3-colorability have specific, identifiable, and systematically reproducible properties is an area of active research. Examples of graph-theoretic properties proposed as *order parameters* separating "easy" instances from "hard" include 3-paths [60], minimal unsolvable subproblems [42] and frozen developments [15]. Some of these proposed order parameters have resulted in algorithms [60] [48] [37] for generating infinite families of non-3-colorable graphs conjectured (and often computationally verified) to be "hard". In this section, we investigate a link between Nullstellensatz certificate degree and "hard" non-3-colorable graphs.

We begin by describing the algorithms generating the "hard" instances that we tested, which were the minimum unsolvable graphs (MUGs) from [48], and the 4-critical graph units (4-CGUs) from [37]. We then display our experimental results, comparing **NulLA** with the Gröbner basis method, and conclude with comments about the relationship between Nullstellensatz certificate degree and "hard" instances of 3-colorability.

Minimal Unsolvable (non-3-colorable) Subgraphs (MUGs)

In [48], a randomized algorithm for generating infinitely large instances of quasi-regular, 4-critical graphs is described. These quasi-regular, 4-critical graphs are referred to by the authors as *minimal unsolvable subgraphs*, where the term "unsolvable" refers to the non-3-colorability of the graph. In this case, *quasi-regular* refers to graphs containing only vertices of degree three or four, and *4-critical* refers to graphs with chromatic number four such that the removal of any edge decreases the chromatic number to three. The MUG

generation algorithm relies on five core 4-critical, quasi-regular minimal unsolvable graphs, which are randomly chosen and then iteratively constructed using the Hajós calculus, creating larger and larger 4-critical graphs. The five core 4-critical, quasi-regular MUGs used in the algorithm are displayed in Figure 5.9.

The algorithm for generating a sequence of Mizuno-Nishihara MUGs is as follows:

****
ALGORITHM: MUG Hard Instance Generation Algorithm
INPUT: An integer k
OUTPUT: A sequence $G_0, G_1, \ldots, G_k$ of near-4-clique-free, 4-critical graphs
1 $G_0 \leftarrow$ a random MUG
2 **for** $i = 0$ to k **do**
3 $G_{\text{join}} \leftarrow$ a random MUG
4 Select an edge $\{u, v\}$ at random from G_i, and an edge $\{x, y\}$ at random from G_{join}, such that $\deg(u)$ and $\deg(x)$ are ≤ 3.
5 Remove the edges $\{u, v\}$ and $\{x, y\}$ from G_i, G_{join}, respectively.
6 $G_{i+1} \leftarrow G_i \cup G_{\text{join}}$, with an additional edge $\{v, y\}$, and where x and u are merged.
7 **end for**
8 **return** $G_0, G_1, \ldots, G_k$
****

The Hajós calculus is the technique used in merging G_i and G_{join}. This construction can be used to generated the entire class of non-3-colorable graphs (see [27] and references therein).

Proposition 5.2.2 *Every graph in the sequence $G_0, G_1, \ldots, G_k$ produced by the MUG instance generation algorithm is 4-critical.*

Proof: Since G_i and G_{join} are 4-critical graphs, when the edges $\{u, v\}$ and $\{x, y\}$ are removed from G_i and G_{join}, respectively, the resulting graphs are 3-colorable. This implies that there exists a proper 3-coloring of each graph where the colors assigned to vertices u and j, and to vertices x and y are the same. Without loss of generality, let the colors

assigned to u, v, x and y be the same. Thus, when the edge $\{v, y\}$ is added, and the vertices x and u are merged, the resulting graph is 4-critical. □

4-critical graph units (4-CGUs)

In [37], a randomized algorithm for generating infinitely large instances of triangle-free, 4-critical graphs is described. The 4-CGU algorithm constructs a particular 4-critical core, which is than joined to the previous graph in the sequence using the Hajós construction. An example of a 4-CGU is displayed in Figure 5.10, and the algorithm for generating a sequence of 4-CGUs follows below.

****
ALGORITHM: Liu-Zhang CGU Hard Instance Generation Algorithm
INPUT: An integer n
OUTPUT: A 4-critical graph in $\lfloor \frac{n}{3} \rfloor$ vertices

Step 1: Let $n = 3m + r$ with both m and r non-negative integers, and $r < 3$.

Step 2: Construct a triangle ΔABC and a circle with $3(m-1)$ vertices denoted as $a_1, b_1, c_1, a_2, b_2, c_2, \ldots, a_{m-1}, b_{m-1}, c_{m-1}$ successively.

Step 3: Connect A with all a_i $(i = 1, \ldots, m-1)$;
Connect B with all b_i $(i = 1, \ldots, m-1)$;
Connect C with all c_i $(i = 1, \ldots, m-1)$.

Step 4: (a) **if** $r = 0$ **then** choose two vertices a_k, a_l from a_i ($i = 1, \ldots, m-1$), connect a_k and a_l;
(b) **if** $r = 1$ **then** choose a vertex a_k from a_i ($i = 1, \ldots, m-1$), a vertex b_l from b_i $(i = 1, \ldots, m-1)$ and a vertex c_m from $(i = 1, \ldots, m-1)$, introduce a new vertex O, connect O with a_k, O with b_l, O with c_m;
(c) **if** $r = 2$ **then** choose two vertices a_{k_1}, a_{k_2} from a_i ($i = 1, \ldots, m-1$), choose two vertices b_{l_1}, b_{l_2} from b_i from b_i $(i = 1, \ldots, m-1)$, introduce two vertices O_1, O_2, connect O_1 with a_{k_1}, O_1 with b_{l_1}, O_2 with a_{k_2}, O_2 with b_{l_2}, O_1 with O_2;

Step 4: Stop.
****

An infinite family of 4-critical, triangle-free graphs is generated by (1) choosing the Grötzch graph (4-critical and triangle free) as the initial graph, (2) choosing a random

number ≥ 9 with $r > 0$, and generating a 4-CGU of that size, (3) merging the two graphs by choosing the edge $\{i, j\}$ randomly from the Grötzch graph, and the edge $\{x, y\}$ as one of the edges in the triangle ΔABC. If non-adjacent vertices are chosen during the 4-CGU construction step, the resulting graph is triangle-free and 4-critical.

Experimental Results on Hard Instances of 3-colorability

We implemented both the MUG hard instance generation algorithm, and the 4-CGU hard instance generation algorithm. We tested both families with **N**ul**LA** over $\mathbb{F}_2$, and also with the Gröbner basis method using CoCoA Lib . In [48], the MUG instances were tested with the Smallk [14] and Brélaz heuristics [6], as well as with six major constraint satisfaction problem (CSP) solvers. In each case, exponential growth in the runtimes were reported by the authors. When we tested the MUG random instances using **N**ul**LA**, we immediately saw corresponding growth in the degree of the Nullstellensatz. We report on these results in Table 5.9.

Graph	*vertices*	*edges*	*rows*	*cols*	*deg*	*terms*	*sec*	*GB sec*
G_0	10	18	198	181	1	3	0	0
G_1	20	37	178,012	329,916	4	563	6.33	.05
G_2	30	55	1,571,328	2,257,211	4	1,961	52.83	.46
G_3	39	72	6,481,224	8,072,429	4	2,272	201.96	5.5
G_4	49	90	22,054,196	24,390,486	≥ 7	–	773.16	150.47
G_5	60	110	–	–	–	–	–	1718.62
G_6	69	127	–	–	–	–	–	3806.17
G_7	78	144	–	–	–	–	–	19837.4

Table 5.9: Hard instances of graph 3-colorability: MUGs.

In Table 5.9, we record both the minimum-degree of the Nullstellensatz certificates, and also the maximum number of monomials in any coefficient (as a measure of density).

For G_4, we record the degree as ≥ 7, since we are certain the degree is not equal to four; thus, by Lemma 4.3.3, the degree must be seven or larger. We were only able to compute the degrees of the first few certificates in the sequence; thus, it is impossible to infer a precise rate of growth for the MUG family. Furthermore, the use of triangle equations as degree-cutters did not reduce the degree, and we were also unable to find alternative Nullstellensatz certificates of lower degree for these graphs. Thus, these graphs appear to be "hard" for **N**ul**LA**, although in order to prove a result about the growth in these certificates, a far more thorough understanding of the certificate structure is needed. We also note that the Gröbner basis method outperformed **N**ul**LA** on these instances. Since the complexity of finding a Nullstellensatz certificate and the complexity of finding a Gröbner basis are closely related, this suggests that there may be further simplifications to **N**ul**LA** that we have not yet discovered.

Graph	*vertices*	*edges*	*rows*	*cols*	*deg*	*terms*	*sec*	*GB sec*
G_0	11	20	247	221	1	3	0	0
G_1	20	37	177,760	329,916	4	655	7.35	.1
G_2	29	54	1,306,695	1,947,902	4	1,636	82.77	.75
G_3	38	71	5,621,140	7,202,749	4	3,204	364.23	1.65
G_4	47	88	17,629,974	20,288,961	≥ 7	–	688.35	10.46
G_5	56	105	–	–	≥ 7	–	–	13.41
G_6	65	122	–	–	≥ 7	–	–	20.82
G_7	74	139	–	–	≥ 7	–	–	75.02
G_8	83	156	–	–	≥ 7	–	–	570.96

Table 5.10: Hard instances of graph 3-colorability: 4-CGUs.

In Table 5.10, we report the results of the **N**ul**LA** experiments on the 4-CGU hard instances of graph 3-colorability. The 4-CGU instance generation algorithm has not been tested as thoroughly with multiple graph coloring algorithms as compared to the MUGs

in [48]. However, the 4-CGUs were tested with Smallk, and exponential running times were reported in [37]. When we tested the 4-CGU algorithm with **N**ul**LA** over $\mathbb{F}_2$, we immediately found corresponding growth in the degree of the Nullstellensatz certificates, at a rate of growth very similar to the rate of growth in the MUG family. For example, G_0 in both families has degree one, G_1, G_2 and G_3 in both families all have degree four, and G_4 is both families has degree ≥ 7. We also note that the 4-CGUs are triangle-free. Thus, no reductions in degree via degree-cutter equations are possible. Furthermore, as in the case of the MUGs, we could not find alternative Nullstellensatz certificates for the 4-CGUs. However, the running times returned by CoCoA Lib in the Gröbner basis experiments were very different between the two families: for example, CoCoA Lib found a Gröbner basis for the 4-CGU G_7 in 75.02 seconds, as compared with 19837.4 seconds for the MUG G_7. This dramatic difference between the Gröbner basis runtimes suggests that MUGS are somehow "algebraically" harder than the 4-CGU's, although a rigorous characterization of "algebraic hardness" has yet to be developed.

The underlying cause in the degree growth of graph 3-colorability certificates remains an open question. It is possible that a thorough understanding of the non-3-colorability certificates will illuminate properties in the underlying graphs that force growth in the certificate degree; and perhaps, those same properties will cause exponential growth in runtimes with respect to other heuristics and solvers. It is interesting to note that of the hundreds of graphs present in the DIMACS computational challenge, the only graphs with degrees greater than one were the MUG graphs, specifically proposed as "hard" instances of graph 3-colorability. The advantage of **N**ul**LA** is that the difference between "hard" and

"easy" instances is stark and clear; graphs with degrees one, four and seven are "easy", "hard" and "harder". Our goal to use **N**ul**LA** not only as a tool for practical computation, but also as a means of exposing structural properties in the underlying graphs that lead to differences in complexity.

5.3 Beyond 3-colorability

In this section, we briefly explore minimum-degree non-k-colorability certificates for $k > 3$. The complete graphs K_n are non-$(n-1)$-colorable. It can be shown that when the n degree-cutter equations capturing the $(n-1)$-cliques in K_n are applied, the degree of the non-$(n-1)$-colorability certificate drops to one. However, in Table 5.11, we explore the minimum-degree non-$(n-1)$-colorability certificates *without* degree-cutter equations, or alternative Nullstellensatz certificates, to gain an intuitive sense about the minimum-degree of "hard" k-colorability instances. Since K_n is the most trivial non-$(n-1)$-colorable graph, its minimum-degree is an indicator of the complexity of the $(n-1)$-colorability problem with respect to **N**ul**LA**.

Graph	k	*vertices*	*edges*	*rows*	*cols*	*deg*	*sec*
K_4	3	4	6	39	35	1 ($\mathbb{F}_2$)	0
K_5	4	5	10	1,413	2,772	5 ($\mathbb{F}_3$)	.02
K_6	5	6	15	8,464	14,784	6 ($\mathbb{F}_2$)	.17
K_7	6	7	21	507,831	1,705,440	13 ($\mathbb{F}_5$)	16385.8
K_8	7	8	28	310,367	373,230	8 ($\mathbb{F}_2$)	304.39
K_9	8	9	36	1,835,286	1,798,940	≥ 17 ($\mathbb{F}_3$)	1410.74
K_10	9	10	45	10,699,214	8,498,776	≥ 19 ($\mathbb{F}_2$)	192358

Table 5.11: K_n: minimum-degrees for non-$(n-1)$-colorability certificates.

In Table 5.11, we display the minimum-degrees of non-$(n-1)$-colorability cer-

tificates of K_n. Note that certificate degrees are with respect to different finite fields $\mathbb{F}_p$, where p is relatively prime to $k = n - 1$. Only 4-colorability over $\mathbb{F}_3$ has a computationally-tractable certificate degree of one. The other K_n graphs have minimum-degrees such as $5, 6$ and even 13. We were not able to find a certificate for K_9, a graph with 9 vertices and 36 edges; we were only able to determine that the certificate has minimum-degree ≥ 17. This suggests that **N**ul**LA** may not be effective for chromatic numbers higher than three, unless degree-cutter equations or alternative Nullstellensatz certificates can be found.

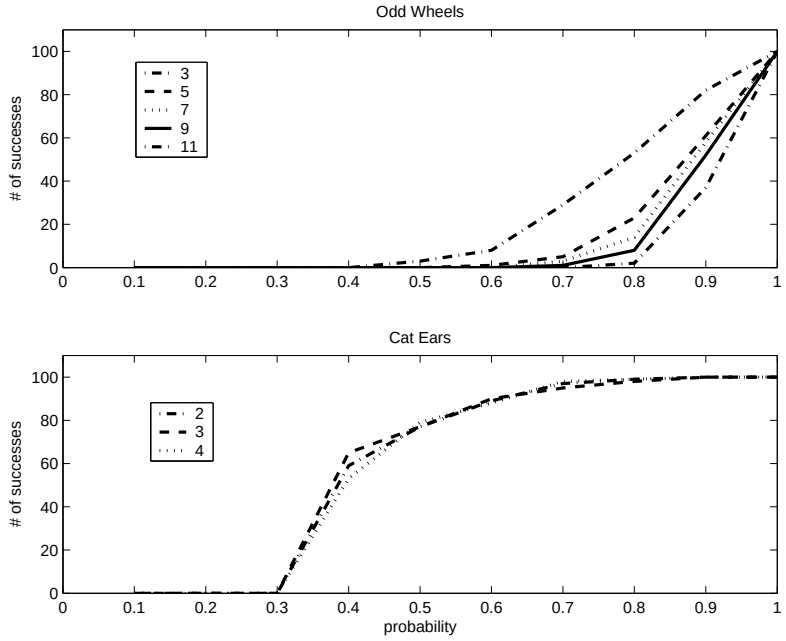

Figure 5.4: Probability tests on odd-wheels and cat-ear graphs over $\mathbb{F}_2$.

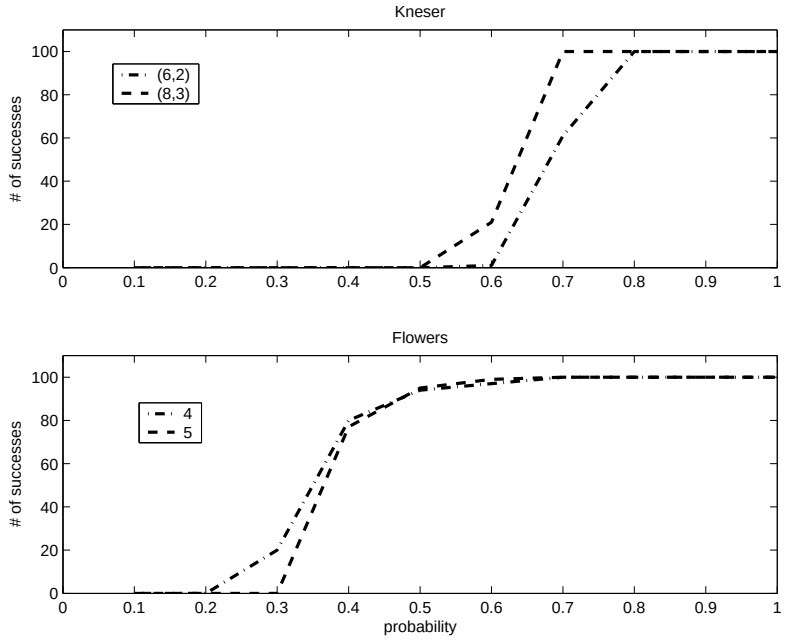

Figure 5.5: Probability tests on Kneser and flower graphs over $\mathbb{F}_2$.

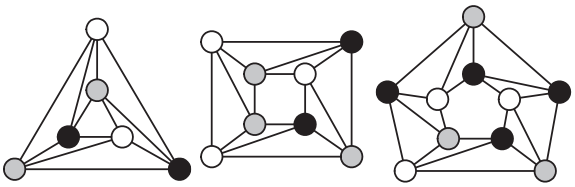

Figure 5.6: $3, 4$ and 5 flowers.

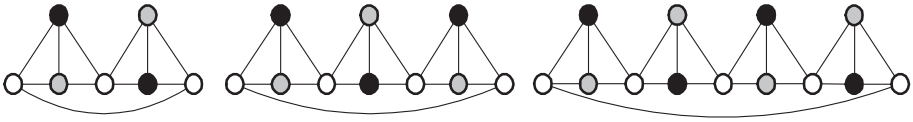

Figure 5.7: $2, 3$ and 4 cat-ears.

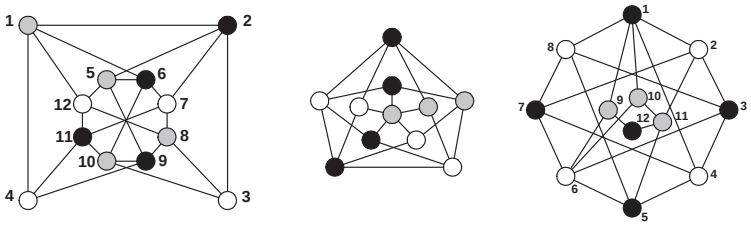

Figure 5.8: A uniquely 3-colorable graph, the Grötzsch graph, and the Jin graph.

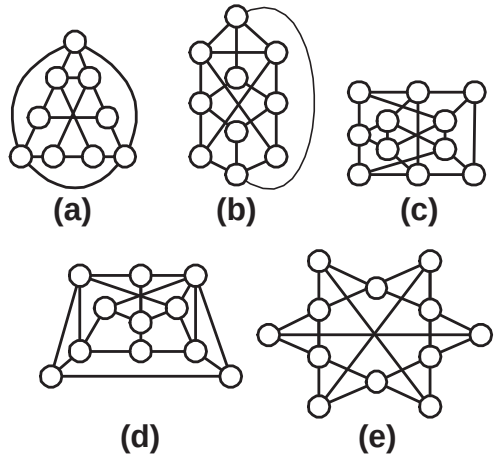

Figure 5.9: 4-critical, near-4-clique-free minimum unsolvable graphs (MUGs).

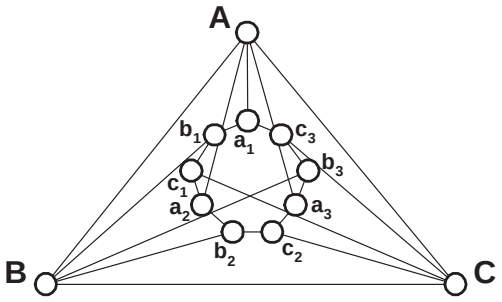

Figure 5.10: An example of a Liu-Zhang 4-CGU.

Chapter 6

Summary and Future Work

"This is not the end. It is not even the beginning of the end.
But it is, perhaps, the end of the beginning."
–Sir Winston Churchill,
1874-1965 .

6.1 Summary

In this dissertation, we described the Nullstellensatz Linear Algebra algorithm (**N**ul**LA**), explored various links between **N**ul**LA** and complexity theory, investigated theoretical lower bounds on the degrees of Nullstellensatz certificates associated with particular encodings, and reported on experimental results. In terms of complexity theory, we demonstrated links between Nullstellensatz certificate bit-size complexity, and the questions of P vs. NP, NP vs. coNP, and NP as a proper or improper set of EXPTIME. Additionally, we proved that the minimum-degree of a Nullstellensatz certificate associated with the Lovász encoding of the independent set problem from Lemma 2.1.1 is $\alpha(G)$, or the size of the largest independent set in the graph, and moreover, we proved that these certificates contain one

monomial for every independent set in the graph. This answered an open question posed by Lovász in [41]. As a by product of this result, we demonstrated that the Lazard bound on projective Nullstellensatz certificates described in Lemma 3.2.3 is tight.

In terms of non-3-colorability, we proved that the minimum-degree of a non-3-colorability Nullstellensatz certificate associated with the encodings of Lemma 2.2.2 or 2.2.6 follows the sequence $1, 4, 7, \ldots$, etc.. We also proved that the minimum-degree of a non-3-colorability Nullstellensatz certificate associated with Lemma 2.2.2 (the encoding over $\mathbb{C}$) is at least four, and the minimum-degree of a non-3-colorability Nullstellensatz certificate associated with Lemma 2.2.6 (the encoding over $\mathbb{F}_2$) is at least one. Additionally, we demonstrated that any graph containing K_4 has a Nullstellensatz certificate of degree four or one (for the encodings of Lemmas 2.2.2 and 2.2.6, $\mathbb{C}$ and $\mathbb{F}_2$, respectively). Furthermore, there are infinite families of graphs that do *not* exhibit any growth at all: the n-th odd-wheel has a minimum-degree non-3-colorability Nullstellensatz certificate of degree four or one (for the encodings of Lemmas 2.2.2 and 2.2.6, $\mathbb{C}$ and $\mathbb{F}_2$, respectively).

Using the encoding of Lemma 2.2.6 and our own C++ implementation of **N**ul**LA**, we computationally proved the non-3-colorability of graphs with almost 2,000 vertices and over 10,000 edges. We also described two infinite families of graphs that exhibit growth in their minimum-degree non-3-colorability certificates using the encoding of Lemma 2.2.6; their first few terms followed the sequence $1, 4, 7, \ldots$, etc..

6.2 Future Work

The Nullstellensatz Linear Algebra (**N**ul**LA**) algorithm requires efficient large-scale linear algebra computations, intersects with aspects of complexity and graph theory, and relies on the development of a library of decision problem encodings that are particularly well-suited to computations with Hilbert's Nullstellensatz. It is exciting that at the close of this dissertation, there are still many open questions and directions for future research still remaining. In this section, we present some of these open questions with respect to computation, complexity theory and encodings.

- **primal/dual**:

  Nul**LA** is an algorithm currently used for detecting combinatorial *infeasibility*; R. Weismantel has suggested developing a "primal/dual" approach, where **N**ul**LA** is combined with another algorithmic approach used for detecting combinatorial *feasibility*.

  For example, the central idea behind degree-cutter equations is that, by *appending* equations to a system of polynomial equations, we can *reduce* the minimum-degree of the associated Nullstellensatz certificates. It is natural to ask if it is possible to develop a systematic methodology for generating degree-cutter equations. Such a method proposed by P. Malkin for generating degree-cutter equations is to simultaneously solve a low-degree **N**ul**LA** linear system, and compute the Gröbner basis for the ideal. If the low-degree **N**ul**LA** linear system has no solution, than we can append one of the intermediate polynomials calculated during the parallel Gröbner basis computation to the system of polynomial equations. Then, we can construct the **N**ul**LA** linear

system associated with the new system of equations (the original system with the new intermediate polynomial added). Essentially, we can use these intermediate Gröbner basis polynomials as a possible degree-cutter equations. Such a "primal/dual" method has yet to be investigated or implemented.

- **Farkas' lemma and solution reconstructions:**

  Via Farkas' lemma, if a linear system $Ax = b$ has *no* solution, then there exists a witness vector u such that $uA = 0$ and $ub \neq 0$. Thus, the existence of such a vector u is a certificate of *infeasibility* of the linear system. In terms of **N**ul**LA**, if the linear system has *no* solution, then there may or may not exist a Nullstellensatz certificate; we must increment the degree and try again. However, it is worth investigating if the Farkas' certificate of infeasibility yields information about an actual solution to the system of polynomial equations. For example, in terms of graph 3-colorability, if the linear system has no solution, could the Farkas' witness certificate somehow contain information about an actual 3-coloring of the graph? B. Sturmfels has observed that ideas of Laurent, Lasserre and Rostalski [32] can be modified to yield explicit solutions.

- **Improving the NulLA linear system solver:**

  The current version of **N**ul**LA** constructs the linear system associated with a given degree d. If the degree d linear system has no solution, then **N**ul**LA** constructs the linear system associated with degree $d+1$. In the current implementation, the linear system associated with degree d and the linear system associated with degree $d+1$ are treated as two completely independent objects. But it seems reasonable that a careful study of the d and $d+1$ linear systems might yield a method for reusing the

degree d linear algebra computations in the solution for the degree $d+1$ linear system.

Another example of a strategy for reusing work comes from the Alon-Tarsi method discussed in Section 5.2.5. The Alon-Tarsi method uses an edge-by-edge incremental approach that sometimes allows the method to quickly isolate a non-3-colorable subgraph. An incremental, edge-by-edge approach might also be very useful with **N**ul**LA**. For example, we could start by constructing the linear system associated with the first ten edges. If the linear system is infeasible, another ten edges could be added, and a method could be devised to reuse the work from the first incremental solution.

Finally, the algorithm used to solve the back-end **N**ul**LA** linear systems is simply an efficient Gaussian Elimination optimized for extremely sparse matrices. However, iterative algorithms, such as the block Lanczos algorithm proposed in [50], are known to be much faster than Gaussian Elimination. Therefore, a future release of **N**ul**LA** could include an implementation of a more sophisticated linear system solver algorithm.

- **Gröbner bases and Faugère's F5 algorithm:**

  The complexity of computing Hilbert's Nullstellensatz provides a lower bound on the complexity of computing a Gröbner basis, since a Gröbner basis for an infeasible polynomial system is eventually only a constant (such as one). Although it is known that computing a Gröbner basis is EXPSPACE-complete (see [22] and references therein), it is reasonable to expect that our combinatorial ideals, such as the graph coloring ideals described by Lemmas 2.2.2 and 2.2.6, might have significantly less extreme upper bounds. It is an open question to determine the upper bounds on these comparatively simple NP-complete combinatorial ideals.

Additionally, we have only compared **N**ul**LA** to the Gröbner basis algorithm implemented in CoCoA Lib . Faugère's F5 algorithm for computing a Gröbner basis is widely considered to be more efficient than Buchberger's algorithm for computing a Gröbner basis. Since there are no efficient implementations of Faugère's algorithm, we should eventually write our own implementation to compare with **N**ul**LA**.

- **Developing a library of encodings:**

  The success of **N**ul**LA** as an algorithm for practical computation depends not only on the efficiency of the actual implementation, but also on the "algebraic compatibility" of the input encoding with Hilbert's Nullstellensatz. In Section 4.2, we demonstrated that the binary encoding of the independent set problem yielded certificates where at least one coefficient is basically an enumeration of the independent sets in the graph. Furthermore, there are families where the minimum-degree of the associated certificates is $O(n)$: we therefore conclude that the binary encoding of the independent set decision problem poorly captures the combinatorial properties of independent sets with respect to computation with the Nullstellensatz.

  An example of an encoding that seems to be "algebraically compatible" with **N**ul**LA** is graph 3-colorability. However, at the conclusion of this project, we have yet to discover another encoding that is as well-suited for **N**ul**LA**-style computation as this one. In order to promote **N**ul**LA** as an algorithm that is useful for practical computation, we must develop a library of encodings that capture decision problems of interest with encodings that are compatible with Hilbert's Nullstellensatz. Towards this end, we recommend low degree, homogenous polynomials.

- **Hamiltonian cycle:**

  Here, we briefly explore the Hamiltonian cycle encoding of Lemma 2.3.3 and offer suggestions for future research directions. In order to use this encoding for computation over finite fields, we add the cyclotomic polynomials from Eq. 2.5 to force ω to take on the value of a primitive root of unity. Our test cases are the line graphs, pictured in Figure 6.1. The 3-line has three vertices and two edges, and the n-line has n vertices and $n-1$ edges, etc..

Figure 6.1: The line graphs.

Graph	*rows*	*cols*	*deg*	*sec*
3-line	128	120	2 ($\mathbb{F}_2$)	0
4-line	1,140	1,260	4 ($\mathbb{F}_3$)	0
5-line	5,363	5,544	5 ($\mathbb{F}_2$)	.01
6-line	22,874	24,024	6 ($\mathbb{F}_5$)	.09
7-line	332,386	388,960	9 ($\mathbb{F}_2$)	3.22
8-line	392,577	437,580	8 ($\mathbb{F}_3$)	8.47
9-line	5,838,769	7,054,320	11 ($\mathbb{F}_2$)	2469.24

Table 6.1: Line graphs: minimum-degrees for non-hamiltonicity

In Table 6.1, we display the results for testing the line graphs for non-hamiltonicity. Lemma 2.3.3 is a relatively simple encoding with $n+1$ variables and $n+m$ equations. However, the line graphs are a trivial non-Hamiltonian graph example, and they exhibit linear growth in their corresponding Nullstellensatz certificate degrees. Thus, despite its simplicity, this encoding *with the use of the cyclotomic polynomial*

does not appear to be promising for **Nul**$\mathbf{LA}$. However, we believe it is the addition of the cyclotomic polynomial in particular that causes the linear growth in degree. Thus, we are very interested in trying this encoding over $\mathbb{F}_p \cup \omega$, and seeing if the minimum-degrees are reduced. This is an experiment that must be performed before the computational usefulness of this encoding is understood.

- **SAT:**

  Here, we explore minimum-degree Nullstellensatz certificates for the "induction" principle, which is at the heart of research in logic and propositional proof systems [8, 26]. From Section 4.4, recall that the induction principle is the Boolean formula in n variables:

$$\mathrm{IND}_n = x_1 \wedge (\neg x_1 \vee x_2) \wedge (\neg x_2 \vee x_3) \wedge \cdots \wedge (\neg x_{n-1} \vee x_n) \wedge \neg x_n \ ,$$

  and that via Theorem 4.4.1 from [8], and via the encoding over $\mathbb{Q}$ from Lemma 2.7.1, we have the following lower and upper bounds on the associated Nullstellensatz certificate degrees:

$$\lfloor \log_2(n) \rfloor - 1 \leq d \leq \lceil \log_2(n-1) \rceil \ .$$

In Table 6.2, we display our experimental results testing the "induction" principle encoded via Lemma 2.7.1. We note that this table only displays the values of n where the associated Nullstellensatz certificates increased in degree. For example, from Table 6.2, we can infer that IND_3 through IND_5 have degree two, and IND_6 through IND_{13} have degree three, etc.. By inspection of the table, we can see that

IND_n	*rows*	*cols*	*sec*	*deg*	*lb*	*ub*
3	20	28	0	1	0	1
5	55	66	0	1	1	2
6	210	364	0	2	1	3
13	2,170	2,835	.02	2	2	4
14	11,376	19,720	.16	3	2	4
29	225,126	292,640	4.48	3	3	5
30	1,770,692	2,828,936	585.57	4	3	5
32	2,464,671	3,828,825	952.86	4	4	5

Table 6.2: Minimum-degrees for IND_n

the minimum-degree of the Nullstellensatz does indeed always sit between the lower and upper bounds. However, it is not always tight with either bound, and it remains an open question to determine the precise sequence of the degrees, and to determine the combinatorial meaning of the values of n where the degree increments.

- **Minimum 4-critical subgraphs:**

  The Jin graph from Figure 6.2 is the first graph in an infinite family of triangle-free, 4-chromatic graphs proposed in [28]. However, the Jin graph is not 4-critical, which is a reasonable requirement for a "hard" instance of 3-colorability. In Figure

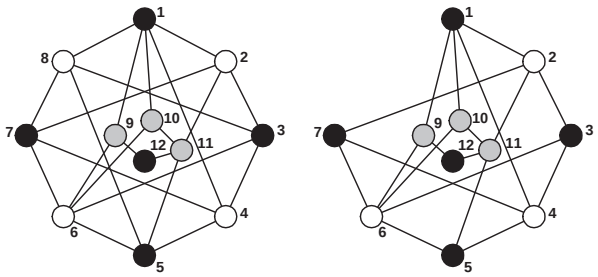

Figure 6.2: The Jin graph and a 4-critical subgraph.

6.2, we see two graphs: the Jin graph, and the subgraph produced by removing the 8-th vertex and all incident edges. This particular subgraph was the subgraph isolated by **N**ul**LA** in the non-3-colorability certificate. We experimentally verified that this subgraph was 4-critical. We also saw similar results in the non-3-colorability certificates for the Kneser graphs. When we inspected the certificates, many vertices and edges were missing.

We *conjecture* that the subgraph isolated by the non-3-colorability Nullstellensatz certificates produced by **N**ul**LA** is always the minimal (with respect to number of edges) 4-critical subgraph in the graph. Thus, we propose to use **N**ul**LA** as a graph-theoretic tool to help isolate 4-critical non-3-colorable subgraphs, and we hope that **N**ul**LA** can facilitate development of infinite families of 4-critical graphs, which are essential to deepening our understanding of "hard" instances of 3-colorability.

- **Growth in the minimum-degree of non-3-colorability certificates:**

  In Subsection 5.2.6, we described two families of graphs, proposed in [48] and [37], where the first few graphs in the family exhibited growth in the minimum-degree of their non-3-colorability certificates. However, we were unable to *prove* that the minimum-degree continues to grow, or to explicitly describe the *rate* of growth.

  Nul**LA** may be able to provide a first step in identifying systematically reproducible graph-theoretic properties shared by "hard" instances of graph 3-colorability. If we could understand the combinatorial meaning of the coefficients in the non-3-colorability certificates, we could potentially gain insight into *why* certain graphs are non-3-colorable when a combinatorial explanation is not readily apparent. Further-

more, understanding the combinatorial meaning of the certificate coefficients would allow us to understand *why* certain graphs have higher-degree non-3-colorability certificates than others; or *why*, with respect to **N**ul**LA**, the non-3-colorability of certain 4-critical graphs is harder to determine than others. A central open question remaining in this dissertation is to explicitly describe an infinite family of non-3-colorable graphs where the minimum-degree of the associated Nullstellensatz certificates grows (either linearly or logarithmically) with respect to n. Finding such a family would lead immediately to a second, and perhaps more interesting, question: Would such a family of **N**ul**LA** "hard" instances be a family of "hard" instances for other algorithms, or are the algebraic properties of graphs that are problematic for **N**ul**LA** easily surmountable by other algorithms? Finally, is it possible that **N**ul**LA** could contribute to the development of a rigorous mathematical criteria for identifying universally "hard" instances of graph 3-colorability? In terms of **N**ul**LA**, it is our belief that these goals are inseparable from finding a combinatorial interpretation of the Nullstellensatz certificate coefficients.

Bibliography

[1] N. Alon. Combinatorial Nullstellensatz. *Combinatorics, Probability and Computing*, 8:7–29, 1999.

[2] N. Alon and M. Tarsi. Colorings and orientations of graphs. *Combinatorica*, 12:125–134, 1992.

[3] M. Anjos. Semidefinite optimization approaches for satisfiability and maximum-satisfiability problems. *Journal on Satisfiability, Boolean Modeling and Computation*, 1:1–47, 2005.

[4] E. Arrondo. Another Elementary Proof of the Nullstellensatz. *American Mathematical Monthly*, 113(2):169–170, 2006.

[5] D.A. Bayer. *The Division Algorithm and the Hilbert Scheme*. Ph.D. Thesis, Harvard University, 1982.

[6] D. Brélaz. New methods to color the vertices of a graph. *Communications of the ACM*, 22:251–256, 1979.

[7] W.D. Brownawell. Bounds for the degrees in the Nullstellensatz. *Annals of Mathematics*, 126(3):577–591, 1987.

[8] S. Buss and T. Pitassi. Good degree bounds on Nullstellensatz refutations of the induction principle. In *IEEE Conference on Computational Complexity*, pages 233–242, 1996.

[9] C.-Y. Chao and Z. Chen. On uniquely 3-colorable graphs. *Discrete Mathematics*, 112:374–383, 1993.

[10] P. Cheeseman, B. Kanefsky, and W. Taylor. Where the *Really* hard problems are. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 331–337, 1991.

[11] CoCoATeam. CoCoA: a system for doing Computations in Commutative Algebra. Available at `http://cocoa.dima.unige.it`.

[12] D. Cox, J. Little, and D. O'Shea. *Using algebraic geometry.* Springer, New York, second edition, 1997.

[13] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties and Algorithms.* Springer, New York, 1998.

[14] J. Culberson. Overview of the smallk graph coloring program. `http://www.cs.ualberta.ca/\~joe/Coloring/Colorscr/smallk.html788`.

[15] J. Culberson and I. Gent. Frozen development in graph coloring. *Theoretical Computer Science*, 265:227–265, 2001.

[16] R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer-Verlag, 3rd edition, 2006.

[17] D.S. Dummit and R.M. Foote. *Abstract algebra*. Wiley, 1999.

[18] M. Dyer and C. Greenhill. On Markov chains for independent sets. *J. of Algorithms*, 35:17–49, 2000.

[19] S. Eliahou. An algebraic criterion for a graph to be four-colourable. *Aportaciones Matemáticas*, 6:3–27, 1992.

[20] K.G. Fischer. Symmetric polynomials and Hall's theorem. *Discrete Math*, 69(3):225–234, 1988.

[21] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979.

[22] J. Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, Cambridge, second edition, 1999.

[23] D. Hilbert. Über die vollen Invariantesysteme. *Annals of Mathematics*, 42:313–373, 1893.

[24] C.J. Hillar and T. Windfeldt. An algebraic characterization of uniquely vertex colorable graphs. *Journal of Combinatorial Theory Series B*, 98:400–414, 2008.

[25] I. Holyer. The NP-Completeness of edge coloring. *SIAM Journal of Computing*, 10(4):718–720, 1981.

[26] P. Impagliazzo, P. Pudlák, and J. Sgall. Lower bounds for polynomial calculus and the groebner basis algorithm. *Computational Complexity*, 8:127–144, 1999.

[27] K. Iwama and T. Pitassi. Exponential lower bounds for the tree-like Hajós calculus. *Information Processing Letters*, 54:289–294, 1995.

[28] G. Jin. Triangle-free four-chromatic graphs. *Discrete Mathematics*, 145:151–170, 1995.

[29] E. Koester. On 4-critical planar graphs with high edge density. *Discrete Mathematics*, 98(2):147–151, 1991.

[30] J. Kollár. Sharp effective Nullstellensatz. *Journal of the AMS*, 1(4):963–975, 1988.

[31] M. Kreuzer and L. Robbiano. *Computational Commutative Algebra I.* Springer-Verlag, 2000.

[32] J. Lasserre, M. Laurent, and P. Rostalski. Semidefinite characterization and computation of zero-dimensional real radical ideals. *Foundations of Computational Mathematics*, 2007.

[33] J.B. Lasserre. Polynomials nonnegative on a grid and discrete optimization. *Transactions of the AMS*, 354(2):631–649, 2001.

[34] M. Laurent. Semidefinite representations for finite varieties. *Mathematical Programming*, 109:1–26, 2007.

[35] D. Lazard. Algèbre linéaire sur $\mathbb{K}[x_1, \ldots, x_n]$ et élimination. *Bulletin de las S.M.F*, 105:165–190, 1977.

[36] S.R. Li and W.W. Li. Independence number of graphs and generators of ideals. *Combinatorica*, 1:55–61, 1981.

[37] S. Liu and J. Zhang. Using hajós construction to generate hard graph 3-colorability instances. *Lecture Notes in Computer Science*, 4120:211–225, 2006.

[38] J.A. De Loera. Gröbner bases and graph colorings. *Beitrage zur Algebra und Geometrie*, 36(1):89–96, 1995.

[39] J.A. De Loera, J. Lee, P.N. Malkin, and S. Margulies. Hilbert's Nullstellensatz and an algorithm for proving combinatorial infeasibility. `http://arxiv.org/abs/0801.3788`.

[40] J.A. De Loera, J. Lee, S. Margulies, and S. Onn. Expressing combinatorial optimization problems by systems of polynomial equations and the Nullstellensatz. `http://arxiv.org/abs/0706.0578`.

[41] L. Lovász. Stable sets and polynomials. *Discrete Mathematics*, 124:137–153, 1994.

[42] D. Mammen and T. Hogg. A new look at easy-hard-easy pattern of cominatorial search difficulty. *Journal of Artificial Intelligence Research*, 7:47–66, 1997.

[43] Y. Matiyasevich. A criteria for colorability of vertices stated in terms of edge orientations (in russian). *Discrete Analysis (Novosibirsk)*, 26:65–71, 1974.

[44] Y. Matiyasevich. Some algebraic methods for calculation of the number of colorings of a graph (in russian). *Zapiski Nauchnykh Seminarov POMI*, 293:193–205, 2001.

[45] E.W. Mayr and A.R. Meyer. The complexity of the word problems for commutative semigroups and polynomial ideals. *Advances in Mathematics*, 45:305–329, 1982.

[46] I. Méndez-Díaz and P. Zabala. A branch-and-cut algorithm for graph coloring. *Discrete Applied Mathematics*, 154(5):826–847, 2006.

[47] K. Mizuno and S. Nishihara. Nowhere-zero flow polynomials. *Journal of Combinatorial Theory, Series A*, 108(2):205–215, 2004.

[48] K. Mizuno and S. Nishihara. Constructive generation of very hard 3-colorability instances. *Discrete Applied Mathematics*, 156(2):218–229, 2008.

[49] M. Mnuk. Representing graph properties by polynomial ideals. In *Computer Algebra in Scientific Computing*, pages 431–444. Fourth International Workshop on Computer Algebra in Scientific Computing, 2001.

[50] P. Montgomery. A block Lanczos algorithm for finding dependencies over GF(2). *Lecture Notes in Computer Science*, 921:106–120, 1995.

[51] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[52] C. Papadimitriou. *Computational Complexity*. Addison Wesley Longman, 1994.

[53] P. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming, Series B*, 96(2):293–320, 2003.

[54] C. Picouleau. Complexity of the Hamiltonian cycle in regular graph problem. *Theoretical computer science*, 131(2):463–473, 1994.

[55] W. Schnyder. Planar graphs and poset dimension. *Order*, 5(4):323–343, 1989.

[56] A. Schrijver. *Theory of Linear and Integer Programming.* Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons, Chichester, West Sussex, England, 1986.

[57] S.E. Shauger. Results on the Erdös-Gyárfás conjecture in $k_{1,m}$-free graphs. In *Congressus Numerantium*, pages 61–65. Twenty-ninth Southeastern International Conference on Combinatorics Graph Theory and Computing, 1998.

[58] A. Simis, W. Vasconcelos, and R. Villarreal. On the ideal theory of graphs. *Journal of Algebra*, 167(2):389–416, 1994.

[59] M. Sipser. *Introduction to the Theory of Computation.* PWS Publishing Company, 2001.

[60] R. Vlasie. Systematic generation of very hard cases for graph 3-colorability. In *Tools with Artificial Intelligence*, pages 114–119. Seventh International Conference on Artificial Intelligence, 1995.

[61] M. Yannakakis. Expressing combinatorial optimization problems by linear programs. *Journal of Computer and System Sciences*, 43(3):441–466, 1991.