Graph Colouring Is Hard on Average for Polynomial Calculus

Jonas Conneryd

Lund University and University of Copenhagen

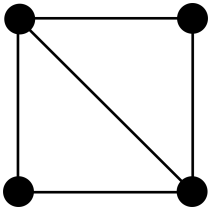
Proof Complexity and Beyond
Oberwolfach, Germany
March 27, 2025

Joint work with Susanna F. de Rezende, Jakob Nordström, Shuo Pang, and Kilian Risse

Graph Colouring

Can vertices of graph G be coloured with k colours so that no edge is monochromatic?

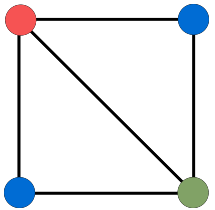
One of Karp's original 21 NP-complete problems [Kar72]



Graph Colouring

Can vertices of graph G be coloured with k colours so that no edge is monochromatic?

One of Karp's original 21 NP-complete problems [Kar72]



✓: $k = 3$

✗: $k = 2$

Is Colouring Hard?

On one hand, colouring is hard—even to approximate:

- if G is k-colourable, best efficient algorithm uses $kn/\mathrm{polylog}(n)$ colours [Hal93]
- if G promised 3-colourable, best efficient algorithm uses $n^{0.199..}$ colours [KT17]
- NP-hard to approximate within $n^{1-\varepsilon}$ factor [FK98; Zuc07]

Is Colouring Hard?

On one hand, colouring is hard—even to approximate:

- if G is k-colourable, best efficient algorithm uses $kn/\mathrm{polylog}(n)$ colours [Hal93]
- if G promised 3-colourable, best efficient algorithm uses $n^{0.199..}$ colours [KT17]
- NP-hard to approximate within $n^{1-\varepsilon}$ factor [FK98; Zuc07]

...but practical algorithms often perform surprisingly well, e.g.

- backtracking search [Kor75; Lew21]
- integer programming [MT96; GM12]
- algebraic algorithms [DLMM08; DLMO09; DLMM11; DMP+15]

Is Colouring Hard?

On one hand, colouring is hard—even to approximate:

- if G is k-colourable, best efficient algorithm uses $kn/\mathrm{polylog}(n)$ colours [Hal93]
- if G promised 3-colourable, best efficient algorithm uses $n^{0.199..}$ colours [KT17]
- NP-hard to approximate within $n^{1-\varepsilon}$ factor [FK98; Zuc07]

...but practical algorithms often perform surprisingly well, e.g.

- backtracking search [Kor75; Lew21]
- integer programming [MT96; GM12]
- algebraic algorithms [DLMM08; DLMO09; DLMM11; DMP+15]

Algebraic algorithms captured by algebraic proof systems

Proof complexity lower bounds $\implies$ unconditional hardness for these algorithms

Our Results

For algebraic proof systems, *worst-case* exponential lower bounds known for colouring [LN17; AO19]

Colouring *easy* except in few artificial cases?

Our Results

For algebraic proof systems, *worst-case* exponential lower bounds known for colouring [LN17; AO19]

Colouring *easy* except in few artificial cases?

To refute this, want average-case hardness, just as for resolution [BCMM05]

Main Result

With probability $1 - o(1)$, *polynomial calculus* requires exponential size for refuting 3-colouring on random graphs

Polynomial Calculus [CEI96]

To prove set of polynomials $\mathcal{P} = \{p_1, \ldots, p_m\}$ has no common root, derive new polynomials in ideal $\langle \mathcal{P} \rangle$ through

$$\text{Linear combination: } \frac{p \quad q}{\alpha p + \beta q} \qquad \alpha, \beta \in \mathbb{F}$$

$$\text{Multiplication: } \frac{p}{x \cdot p} \qquad x \text{ any variable}$$

Refutation of $\mathcal{P}$ is derivation of 1—sound and complete for Boolean $\mathcal{P}$

Polynomial Calculus [CEI96]

To prove set of polynomials $\mathcal{P} = \{p_1, \ldots, p_m\}$ has no common root, derive new polynomials in ideal $\langle \mathcal{P} \rangle$ through

$$\text{Linear combination: } \frac{p \quad q}{\alpha p + \beta q} \qquad \alpha, \beta \in \mathbb{F}$$

$$\text{Multiplication: } \frac{p}{x \cdot p} \qquad x \text{ any variable}$$

Refutation of $\mathcal{P}$ is derivation of 1—sound and complete for Boolean $\mathcal{P}$

Complexity measures:

- **Size:** Total # of monomials in proof lines (with multiplicities)
- **Degree:** Largest degree among monomials in proof lines

Graph Colouring and Polynomials

Encode k-colouring as polynomials over field $\mathbb{F}$

$x_{v,i} = 1 \iff$ "vertex v gets colour i"

$\sum_{i=1}^{k} x_{v,i} - 1,$	$\forall v$	"every vertex gets a colour"
$x_{v,i} \cdot x_{v,i'},$	$\forall v,\ i \neq i'$	"no vertex gets > 1 colour"
$x_{u,i} \cdot x_{v,i},$	$\forall (u,v) \in E(G)$	"no monochromatic edges"
$x_{v,i}^2 - x_{v,i},$	$\forall v, i$	Boolean axioms

Graph Colouring and Polynomials

Encode k-colouring as polynomials over field $\mathbb{F}$

$$x_{v,i} = 1 \iff \text{“vertex } v \text{ gets colour } i\text{”}$$

$\sum_{i=1}^{k} x_{v,i} - 1,$	$\forall v$	“every vertex gets a colour”
$x_{v,i} \cdot x_{v,i'},$	$\forall v,\ i \neq i'$	“no vertex gets > 1 colour”
$x_{u,i} \cdot x_{v,i},$	$\forall (u,v) \in E(G)$	“no monochromatic edges”
$x_{v,i}^2 - x_{v,i},$	$\forall v, i$	Boolean axioms

Can also deal with other encoding [Bay82] more common in math:

- Add kth root of unity ξ to $\mathbb{F}$
- $x_v = \xi^i \iff$ “vertex v gets colour i”

Formal Statement of Main Result

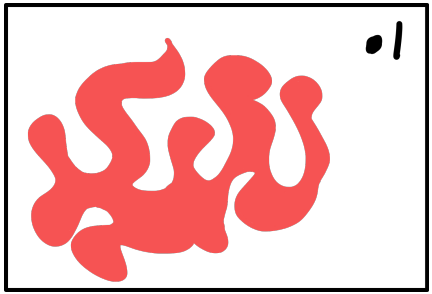
Theorem

If G is sparse random graph on n vertices, then with probability $1 - o(1)$ polynomial calculus requires size $\exp(\Omega(n))$ to refute G is 3-colourable.

- Holds over any field
- Holds for both random regular graphs and Erdős–Rényi random graphs

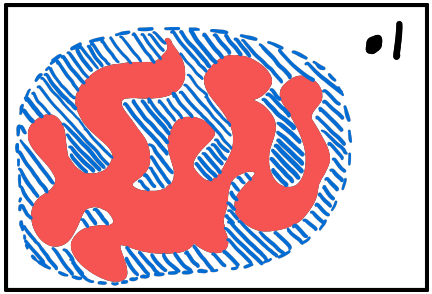
Prove $\Omega(n)$ degree lower bound; implies $\exp(\Omega(n))$ size lower bound [IPS99]

Degree Lower Bounds and R-operators



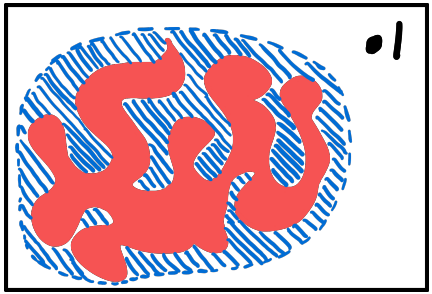
Derivable in degree $\leq D$

Degree Lower Bounds and R-operators



- Derivable in degree $\leq D$
- Overapproximation

Degree Lower Bounds and R-operators



Derivable in degree $\leq D$

Overapproximation

Define so-called R**-operator** [Raz98] on polynomials such that

- $R(p) = 0$, for each input polynomial p
- $R(p) + R(q) = R(p + q)$
- If $R(p) = 0$ then $R(x \cdot p) = 0$, for all p of degree $\leq D - 1$
- $R(1) = 1$

Overapproximation is kernel of R

R as in Reduction

Put total order $\prec$ on monomials in $\mathbb{F}[\mathbf{x}]$, where 1 smallest

R as in Reduction

Put total order $\prec$ on monomials in $\mathbb{F}[\mathbf{x}]$, where 1 smallest

Ideal $\langle \mathcal{P} \rangle$ of $\mathcal{P} = \{p_1, \ldots, p_m\}$ is set of polynomials $q = \sum_i q_i p_i$

For ideal $\langle \mathcal{P} \rangle$, define **reduction operator** $R_{\langle \mathcal{P} \rangle} : p \mapsto r$

- r is polynomial with smallest terms such that $r = p - q$, where $q \in \langle \mathcal{P} \rangle$
- analogous to remainder term after division

R as in Reduction

Intuition: if set of input polynomials satisfiable, 1 not derivable

R as in Reduction

Intuition: if set of input polynomials satisfiable, 1 not derivable

Then, R-operator can be reduction modulo input polynomials:

- $R(p) = 0$ for each input polynomial p
- $R(p) + R(q) = R(p + q)$
- If $R(p) = 0$, then $R(x \cdot p) = 0$ for *all* p
- $R(1) = 1$ by above

} by definition

R as in Reduction

Intuition: if set of input polynomials satisfiable, 1 not derivable

Then, R-operator can be reduction modulo input polynomials:

- $R(p) = 0$ for each input polynomial p
- $R(p) + R(q) = R(p + q)$
- If $R(p) = 0$, then $R(x \cdot p) = 0$ for *all* p
- $R(1) = 1$ by above

For unsatisfiable input, *pseudo-reduction* operator R pretends to be above reduction operator. Low-degree computations cannot tell the difference.

Proof Ideas

If set $\mathcal{P}$ of input polynomials satisfiable, get *perfect* R-operator from reduction modulo $\langle \mathcal{P} \rangle$

...but $\mathcal{P}$ unsatisfiable, so $1 \in \langle \mathcal{P} \rangle$

“Almost” works. Still leverage reduction somehow?

Proof Ideas

If set $\mathcal{P}$ of input polynomials satisfiable, get *perfect* R-operator from reduction modulo $\langle \mathcal{P} \rangle$

...but $\mathcal{P}$ unsatisfiable, so $1 \in \langle \mathcal{P} \rangle$

"Almost" works. Still leverage reduction somehow?

Alekhnovich–Razborov [AR03]

- define R using real reduction
- reduce different monomials modulo different satisfiable subsets of $\mathcal{P}$
- carefully choose subsets so inconsistencies invisible in low degree

Local” ReductionLocal” ReductionLocal” ReductionLocal” ReductionLocal” ReductionLocal” Reduction“Local” Reduction

In more detail, idea is:

1. Associate $m \sim S(m) \subseteq V$ and ideal $\langle S(m) \rangle$ generated by k-colouring polynomials on $G[S(m)]$
2. Define R “locally” on each monomial:

$$R(p) = R\Big(\sum_i a_i m_i\Big) \coloneqq \sum_i a_i R_{\langle S(m_i) \rangle}(m_i)$$

Our Goal

Want R to look like reduction modulo ideal for low-degree p

Maybe, for well-chosen S and, say, $p = m_1 + m_2$, could get

$$\begin{aligned} R(m_1 + m_2) &= R_{\langle S(m_1)\rangle}(m_1) + R_{\langle S(m_2)\rangle}(m_2) \\ &\stackrel{!}{=} R_{\langle S(m_1)\cup S(m_2)\rangle}(m_1 + m_2) \end{aligned}$$

That is, want $R_{\langle S(m)\rangle}(m) = R_{\langle U\rangle}(m)$ for all $U \supseteq S(m)$ not too large

Our Goal

What does it mean that $R_{\langle S(m)\rangle}(m) = R_{\langle U\rangle}(m)$?

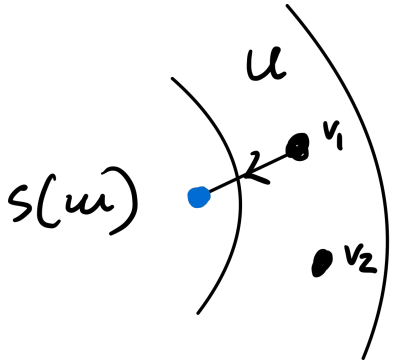
Syntactically: best reduction of m by $\langle U\rangle$ could be done already in $\langle S(m)\rangle$

Our Goal

Semantically: put order on V, can extend every colouring of $S(m)$ to one for U in order-preserving way

$\Longrightarrow$ "U says no more than $S(m)$" about colourings of m

Order-preserving: colours in $U \setminus S(m)$ either fixed or depend only on single, smaller vertex in $S(m)$

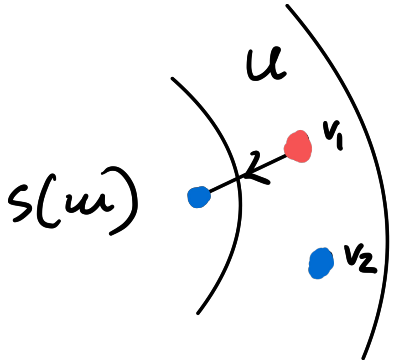


Our Goal

Semantically: put order on V, can extend every colouring of $S(m)$ to one for U in order-preserving way

$\Longrightarrow$ "U says no more than $S(m)$" about colourings of m

Order-preserving: colours in $U \setminus S(m)$ either fixed or depend only on single, smaller vertex in $S(m)$



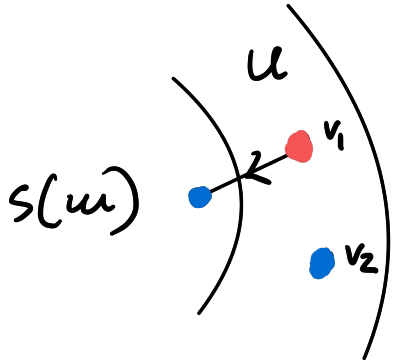
Our Goal

Semantically: put order on V, can extend every colouring of $S(m)$ to one for U in order-preserving way

$\Longrightarrow$ "U says no more than $S(m)$" about colourings of m

Order-preserving: colours in $U \setminus S(m)$ either fixed or depend only on single, smaller vertex in $S(m)$

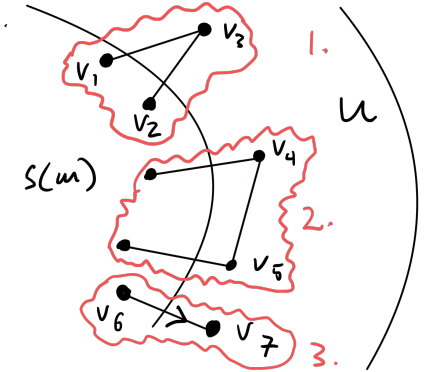
Obstructions?



Constructing $S(m)$

Three obstructions:

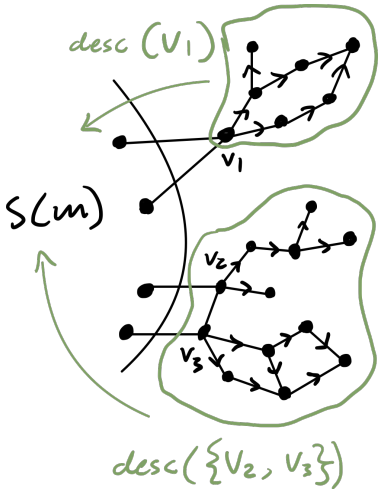
1. dependence on > 1 vertex in $S(m)$
2. dependence between neighbours of $S(m)$
3. small neighbours



Constructing $S(m)$

Construct $S(m)$ iteratively:

1. start with $S(m) = \mathrm{Desc}(V(m))$
2. while bad structure exists, add it and descendants to $S(m)$

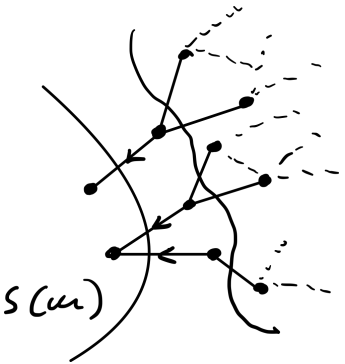


Constructing $S(m)$

Resulting set has no obstructions!

Can extend colouring on $S(m)$ to all of U:

- fix “good” colouring outside neighbourhood
- “patch” it on neighbourhood

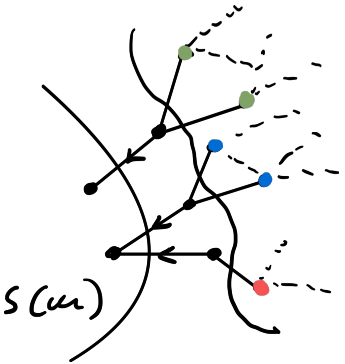


Constructing $S(m)$

Resulting set has no obstructions!

Can extend colouring on $S(m)$ to all of U:

- fix “good” colouring outside neighbourhood
- “patch” it on neighbourhood

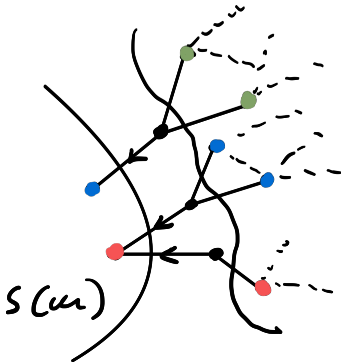


Constructing $S(m)$

Resulting set has no obstructions!

Can extend colouring on $S(m)$ to all of U:

- fix “good” colouring outside neighbourhood
- “patch” it on neighbourhood



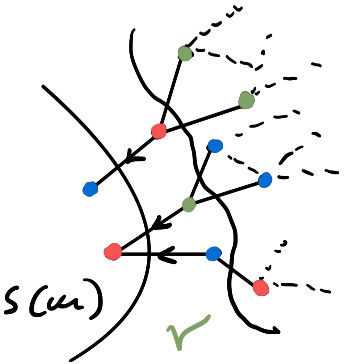
Constructing $S(m)$

Resulting set has no obstructions!

Can extend colouring on $S(m)$ to all of U:

- fix “good” colouring outside neighbourhood
- “patch” it on neighbourhood

But not clear at all size of $S(m)$ does not blow up...



Key Technical Ingredients

1 **Local sparsity:** Vertex-induced subgraph of every subset $U \subseteq V$ of size $\leq \varepsilon n$ has at most $(1+\delta)|U|$ edges

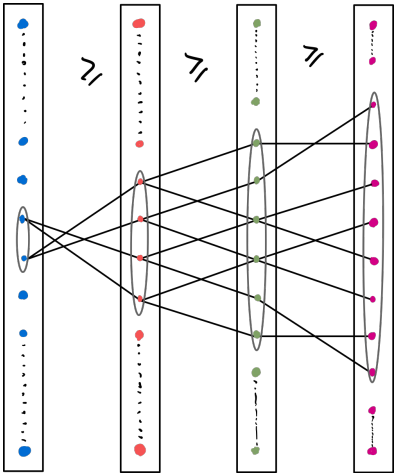
Key Technical Ingredients

1 **Local sparsity:** Vertex-induced subgraph of every subset $U \subseteq V$ of size $\leq \varepsilon n$ has at most $(1+\delta)|U|$ edges

2 **Good vertex order:**

- always add all descendants, so this set must be small for every vertex
- if all ordered paths have length c and max degree is Δ, size is at most Δ^c

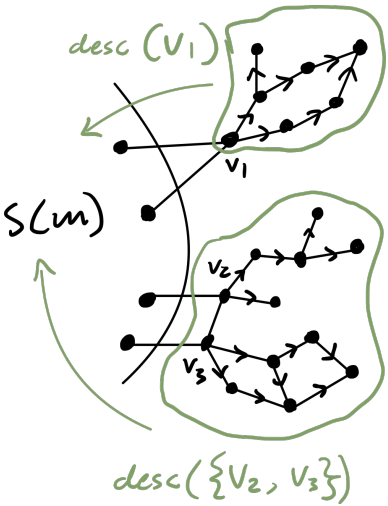
[RT22]: order by proper colouring of graph $\implies$ ordered paths have length $\chi(G) = O(1)$



$S(m)$ Is Small

Proof by picture:

- at each step, add $\geq$ one more edge than vertices
- short ordered paths $\Longrightarrow$ few vertices added per step
- quickly becomes dense—contradicts sparsity



Open Problems

1. Average-case colouring lower bounds for other proof systems?
   - Sherali–Adams
   - sum-of-squares
   - cutting planes

Open Problems

1. Average-case colouring lower bounds for other proof systems?
   - Sherali–Adams
   - sum-of-squares
   - cutting planes

2. Results field-independent; refine to account for characteristic (cf. [AR03])?

Summary

This work:

- Polynomial calculus requires exponential size for colouring on random graphs
- Implies exponential running time for algebraic algorithms successful in practice

Summary

This work:

- Polynomial calculus requires exponential size for colouring on random graphs
- Implies exponential running time for algebraic algorithms successful in practice

Future directions:

- Refine to account for field characteristic?
- Colouring lower bounds for other proof systems?

Summary

This work:

- Polynomial calculus requires exponential size for colouring on random graphs
- Implies exponential running time for algebraic algorithms successful in practice

Future directions:

- Refine to account for field characteristic?
- Colouring lower bounds for other proof systems?

Thank you!

References I

[AO19] A. Atserias and J. Ochremiak, Proof complexity meets algebra, *ACM Transactions on Computational Logic*, vol. 20, 1:1–1:46, Feb. 2019, Preliminary version in *ICALP '17*.

[AR03] M. Alekhnovich and A. A. Razborov, Lower bounds for polynomial calculus: Non-Binomial case, *Proceedings of the Steklov Institute of Mathematics*, vol. 242, pp. 18–35, 2003.

[Bay82] D. A. Bayer, "The division algorithm and the Hilbert scheme," Ph.D. dissertation, Harvard University, Cambridge, Massachusetts, 1982.

[BCMM05] P. Beame, J. C. Culberson, D. G. Mitchell, and C. Moore, The resolution complexity of random graph k-Colorability, *Discrete Applied Mathematics*, vol. 153, no. 1-3, pp. 25–47, Dec. 2005.

References II

[CEI96] M. Clegg, J. Edmonds, and R. Impagliazzo, Using the Groebner basis algorithm to find proofs of unsatisfiability, in *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC '96)*, May 1996, pp. 174–183.

[DLMM08] J. A. De Loera, J. Lee, P. N. Malkin, and S. Margulies, Hilbert's Nullstellensatz and an algorithm for proving combinatorial infeasibility, in *Proceedings of the 21st International Symposium on Symbolic and Algebraic Computation (ISSAC '08)*, Jul. 2008, pp. 197–206.

[DLMM11] J. A. De Loera, J. Lee, P. N. Malkin, and S. Margulies, Computing infeasibility certificates for combinatorial problems through Hilbert's Nullstellensatz, *Journal of Symbolic Computation*, vol. 46, no. 11, pp. 1260–1283, Nov. 2011.

References III

[DLMO09] J. A. De Loera, J. Lee, S. Margulies, and S. Onn, Expressing combinatorial problems by systems of polynomial equations and Hilbert's Nullstellensatz, *Combinatorics, Probability and Computing*, vol. 18, no. 04, pp. 551–582, Jul. 2009.

[DMP+15] J. A. De Loera, S. Margulies, M. Pernpeintner, E. Riedl, D. Rolnick, G. Spencer, D. Stasi, and J. Swenson, Graph-coloring ideals: Nullstellensatz certificates, Gröbner bases for chordal graphs, and hardness of Gröbner bases, in *Proceedings of the 40th International Symposium on Symbolic and Algebraic Computation (ISSAC '15)*, Jul. 2015, pp. 133–140.

[FK98] U. Feige and J. Kilian, Zero knowledge and the chromatic number, *Journal of Computer and System Sciences*, vol. 57, no. 2, pp. 187–199, 1998.

References IV

[GM12] S. Gualandi and F. Malucelli, Exact solution of graph coloring problems via constraint programming and column generation, *INFORMS Journal on Computing*, vol. 24, no. 1, pp. 81–100, 2012.

[Hal93] M. M. Halldórsson, A still better performance guarantee for approximate graph coloring, *Information Processing Letters*, vol. 45, no. 1, pp. 19–23, Jan. 1993.

[IPS99] R. Impagliazzo, P. Pudlák, and J. Sgall, Lower bounds for the polynomial calculus and the Gröbner basis algorithm, *Computational Complexity*, vol. 8, no. 2, pp. 127–144, 1999.

[Kar72] R. M. Karp, Reducibility among combinatorial problems, in *Complexity of Computer Computations*, ser. The IBM Research Symposia Series, Springer, 1972, pp. 85–103.

References V

[Kor75] S. M. Korman, “Graph colouring and related problems in operations research,” Ph.D. dissertation, Imperial College London, 1975.

[KT17] K.-I. Kawarabayashi and M. Thorup, Coloring 3-Colorable graphs with less than $n^{1/5}$ colors, *Journal of the ACM*, vol. 64, no. 1, Mar. 2017.

[Lew21] R. M. R. Lewis, *Guide to graph colouring—algorithms and applications* (Texts in Computer Science), Second. Springer, Cham, 2021, pp. xiv+303.

[LN17] M. Lauria and J. Nordström, Graph colouring is hard for algorithms based on Hilbert’s Nullstellensatz and Gröbner bases, in *Proceedings of the 32nd Annual Computational Complexity Conference (CCC ’17)*, ser. Leibniz International Proceedings in Informatics (LIPIcs), vol. 79, Jul. 2017, 2:1–2:20.

References VI

[MT96] A. Mehrotra and M. A. Trick, A column generation approach for graph coloring, *INFORMS Journal on Computing*, vol. 8, no. 4, pp. 344–354, 1996.

[Raz98] A. A. Razborov, Lower bounds for the polynomial calculus, *Computational Complexity*, vol. 7, no. 4, pp. 291–324, Dec. 1998.

[RT22] J. A. Romero Barbosa and L. Tunçel, Graphs with large girth and chromatic number are hard for Nullstellensatz, arXiv.org, 2212.05365, Dec. 2022.

[Zuc07] D. Zuckerman, Linear degree extractors and the inapproximability of max clique and chromatic number, *Theory of Computing*, vol. 3, no. 6, pp. 103–128, Aug. 2007.