Séminaire Lotharingien de Combinatoire **82B** (2019)
Article #52, 12 pp.

Proceedings of the 31st Conference on Formal Power Series and Algebraic Combinatorics (Ljubljana)

Computational complexity, Newton polytopes, and Schubert polynomials

Anshul Adve*[1], Colleen Robichaux†[2], and Alexander Yong‡[2]

[1]*UCLA, Los Angeles, CA 90095*
[2]*Dept. of Mathematics, U. Illinois at Urbana-Champaign, Urbana, IL 61801, USA*

Abstract. The nonvanishing problem asks if a coefficient of a polynomial is nonzero. Many families of polynomials in algebraic combinatorics admit combinatorial counting rules and simultaneously enjoy having *saturated Newton polytopes* (SNP). Thereby, in amenable cases, nonvanishing is in the complexity class NP ∩ coNP of problems with "good characterizations". This suggests a new algebraic combinatorics viewpoint on complexity theory.

This paper focuses on the case of *Schubert polynomials*. These form a basis of all polynomials and appear in the study of cohomology rings of flag manifolds. We give a tableau criterion for nonvanishing, from which we deduce the first polynomial time algorithm. These results are obtained from new characterizations of the *Schubitope*, a generalization of the permutahedron defined for any subset of the $n \times n$ grid, together with a theorem of A. Fink, K. Mészáros, and A. St. Dizier (2018), which proved a conjecture of C. Monical, N. Tokcan, and the third author (2017).

Keywords: Schubert polynomials, Newton polytopes, computational complexity

1 Introduction

The main results of this extended abstract of [1] concern Schubert polynomials; these are found in Section 2. Those results illustrate a general algebraic combinatorics paradigm for computational complexity theory that we wish to put forward in this introduction.

*aadve@g.ucla.edu.

†cer2@illinois.edu. Colleen Robichaux was supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE 1746047.

‡ayong@illinois.edu. Alexander Yong was supported by an NSF grant, a Simons Collaboration grant and a UIUC Campus Research Board grant.

1.1 Nonvanishing decision problems and SNP

Algebraic combinatorics studies families of polynomials parameterized by combinatorial objects $\diamondsuit$

$$F_{\diamondsuit} = \sum_{\alpha \in \mathbb{Z}^n_{\geq 0}} c_{\alpha,\diamondsuit} x^{\alpha} = \sum_{s \in \mathcal{S}} \mathsf{wt}(s) \in \mathbb{Z}[x_1, x_2, \ldots, x_n],$$

each viewed as the multivariate weight generating series for some combinatorially defined set $\mathcal{S}$.

Example 1.1 (Schur polynomials)**.** *$F_{\diamondsuit} = s_{\lambda}$ is a* Schur polynomial, *where $\diamondsuit = \lambda$ is an integer partition. Here, $\mathcal{S}$ is the set of* semistandard Young tableaux *of shape λ with entries in $[n]$, and $\mathsf{wt}(s) = \prod_i x_i^{\# i \in s}$. Schur polynomials are an important basis of the vector space of all symmetric polynomials.*

Example 1.2 (Stanley's chromatic symmetric polynomial)**.** *Another symmetric polynomial is Stanley's* chromatic polynomial *$F_{\diamondsuit} = \chi_G$ [19]. This time $\diamondsuit = G = (V, E)$ is a simple graph, $\mathcal{S}$ is the set of proper n-colorings of G, i.e., functions $s : V \to \{1, 2, \ldots, n\}$ such that $s(i) \neq s(j)$ if $\{i, j\} \in E$, and $\mathsf{wt}(s) = \prod_i x_i^{\# s^{-1}(i)}$.*

Example 1.3 (Schubert polynomials)**.** *The central example of this paper is non-symmetric. It is the family of* Schubert polynomials *$F_{\diamondsuit} = \mathfrak{S}_w$, a basis of all polynomials. Now, $\diamondsuit = w$ is a permutation. There are many choices for $\mathcal{S}$, such as the* reduced compatible sequences *of [4]. Definitions are given in Section 2.*

Problem 1.4 (nonvanishing)**.** *What is the complexity of deciding $c_{\alpha,\diamondsuit} \neq 0$, as measured in the input size of α and $\diamondsuit$ (under the assumption that arithmetic operations take constant time)?*

In this paper, we give a polynomial time algorithm to determine $c_{\alpha,w} \neq 0$ for the Schubert polynomial. In general, nonvanishing may be undecidable: fix $S \subseteq \mathbb{N}$ that is not recursively enumerable, and let $F_m = \sum_{i=1}^{m} c_{i,m} x^m$ with $c_{i,m} = 1$ if $i \in S$ and 0 otherwise. Such sets S exist because there are uncountably many subsets of $\mathbb{N}$, but only countably many algorithms. One can explicitly take S to be the set of halting Turing machines under some numerical encoding [20], or the set of Gödel encodings [10] of statements about $(\mathbb{N}, +, \times)$ provable in first-order Peano arithmetic. All this said, in our cases of interest, $c_{\alpha,\diamondsuit} \in \mathbb{Z}_{\geq 0}$ has *combinatorial positivity*: it is given by a counting rule that implies nonvanishing is in the class NP of problems with a polynomial time checkable certificate of a YES decision.

Evidently, nonvanishing concerns the *Newton polytope*,

$$\mathsf{Newton}(F_{\diamondsuit}) = \mathrm{conv}\{\alpha : c_{\alpha,\diamondsuit} \neq 0\} \subseteq \mathbb{R}^n.$$

C. Monical, N. Tokcan and the third author [16] showed that for many examples, $F_{\diamondsuit}$ has *saturated Newton polytope* (SNP), i.e., $\gamma \in \mathsf{Newton}(F_{\diamondsuit}) \cap \mathbb{Z}^n \iff c_{\gamma,\diamondsuit} \neq 0$. The relevance of SNP to Problem 1.4 is:

> SNP $\Rightarrow$ $\mathsf{nonvanishing}(F_\diamondsuit)$ is equivalent to checking membership of a lattice point in $\mathsf{Newton}(F_\diamondsuit)$.

Example 1.5 ($\mathsf{nonvanishing}(s_\lambda)$ is in P)**.** *$\mathsf{Newton}(s_\lambda)$ is the λ-*permutahedron *$\mathcal{P}_\lambda$, the convex hull of the S_n-orbit of $\lambda \in \mathbb{R}^n$. By symmetry one may assume α is a partition. Thus $c_{\alpha,\lambda}$ is the* Kostka coefficient*, and $c_{\alpha,\lambda} = 0$ if and only if $\alpha \leq \lambda$ in* dominance order*. So $\mathsf{nonvanishing}(s_\lambda)$ is in the class P of polynomial time problems.*

Does the "niceness" of combinatorial positivity and SNP transfer to complexity?

Question 1.6. *Under what conditions does combinatorial positivity and SNP of $\{F_\diamondsuit\}$ imply $\mathsf{nonvanishing}(F_\diamondsuit) \in \mathsf{P}$, or at least that $\mathsf{nonvanishing}(F_\diamondsuit) \notin \mathsf{NP}$-complete?*

On the other hand, χ_G is not generally SNP [16] and $\mathsf{nonvanishing}(\chi_G)$ is hard:

Example 1.7 (χ_G-nonvanishing is NP-complete)**.** *For χ_G, nonvanishing is clearly in NP. In fact, for each fixed $n \geq 3$ it is NP-complete. The n-coloring problem of deciding if a graph has an n-proper coloring is NP-complete for each fixed $n \geq 3$. Given an efficient oracle to solve $\mathsf{nonvanishing}(\chi_G)$, call it on each partition of $|V|$ with n parts to decide if there exists a proper n-coloring. This requires only $O(|V|^n)$ calls, so it is a polynomial reduction of n-coloring to $\mathsf{nonvanishing}(\chi_G)$.*

1.2 Context from computer science; connection to Stanley's Schur positivity conjecture

Examples 1.5 and 1.7 show that $\mathsf{nonvanishing}$ can achieve the extremes of NP. What about the non-extremes?

The class NP-intermediate consists of NP problems that are neither in P nor NP-complete. *Ladner's theorem* states that if $\mathsf{P} \neq \mathsf{NP}$ there exists an (artificial) NP-intermediate problem. Many natural problems from algebra, number theory, game theory and combinatorics are *suspected* to be NP-intermediate. An example is the *Graph Isomorphism problem*.

The class coNP consists of problems whose complements are in NP, i.e., those with a polynomial time checkable certificate of a `NO` decision.

> SNP $\Rightarrow$ given a halfspace description of the Newton polytope, an inequality violation checkable in polynomial time gives a coNP certificate.

The above implication of SNP says that any solution $\{F_\diamondsuit\}$ to the following problem gives $\mathsf{nonvanishing}(F_\diamondsuit) \in \mathsf{NP} \cap \mathsf{coNP}$.

Problem 1.8. *For a combinatorially positive family of SNP polynomials $\{F_\diamondsuit\}$, determine half space descriptions of $\mathsf{Newton}(F_\diamondsuit)$.*

The class $\mathsf{NP} \cap \mathsf{coNP}$ is intriguing. Membership of a problem in $\mathsf{NP} \cap \mathsf{coNP}$ sometimes foreshadows the harder proof that it is in P. For example, consider

$$\mathsf{primes} = \text{“is a positive integer } n \text{ prime?”}$$

Clearly, $\mathsf{primes} \in \mathsf{coNP}$. In 1975, V. Pratt [17] showed $\mathsf{primes} \in \mathsf{NP}$. It was about thirty years before the celebrated discovery of the *AKS primality test* of M. Agrawal, N. Kayal, and N. Saxena [2], establishing $\mathsf{primes} \in \mathsf{P}$.

In retrospect, another example is the *linear programming* problem

$$\mathsf{LPfeasibility} = \text{“is } A\mathbf{x} = b, \mathbf{x} \geq 0 \text{ feasible?”}$$

Clearly $\mathsf{LPfeasibility} \in \mathsf{NP}$. Membership in coNP is a consequence of *Farkas' Lemma* (1902), which is a foundation for LP duality, conjectured by J. von Neumann and proved by G. Dantzig in 1948 (cf. [6]). Yet, it was not until 1979, with L. Khachiyan's work on the *ellipsoid method* that $\mathsf{LPfeasibility} \in \mathsf{P}$ was proved; see, e.g., the textbook [18].

These examples suggest $\mathsf{P} = \mathsf{NP} \cap \mathsf{coNP}$. However, one has *integer factorization*

$$\mathsf{factorization} = \text{“given } 1 < a < b \text{ does there exist a divisor } d \text{ of } b \text{ where } 1 \leq d \leq a\text{?”}$$

An NP certificate is a divisor. A coNP certificate is a prime factorization (verified using the AKS test). Most public key cryptography (such as RSA) relies on $\mathsf{P} \neq \mathsf{NP} \cap \mathsf{coNP}$.

The debate $\mathsf{P} \stackrel{?}{=} \mathsf{NP} \cap \mathsf{coNP}$ may be rephrased as “are problems with good characterizations in P?”. One wants new examples of members of $\mathsf{NP} \cap \mathsf{coNP}$ that are not known to be in P. If such examples are proved to be in P, this adds evidence for “$=$”. Yet, relatively few examples are known. In addition to integer factorization, one has (decision) *Discrete Log*, *Stochastic Games* [5], *Parity Games* [13] and *Lattice Problems* [3]. (It is open whether *Graph Isomorphism* is in coNP.) We now connect this discussion with Example 1.7.

Problem 1.9. *Does restricting to a subclass of graphs G where χ_G is SNP (or Schur positive) change the complexity of n-coloring?*

Conjecture 1.10 (R. P. Stanley [19])**.** *If G is claw-free (i.e., it contains no induced $K_{1,3}$ subgraph), then χ_G is Schur positive.*

Conjecture 1.11 (C. Monical [15])**.** *If χ_G is Schur positive, then it is SNP.*

Combining these two conjectures gives

Conjecture 1.12. *If G is claw-free then χ_G is SNP.*

If coNP contains an NP-complete problem then $\mathsf{NP} = \mathsf{coNP}$ [11], solving an open problem with “$=$”.[1] Now, by [12], n-coloring claw-free graphs is NP-complete. Therefore:

[1] In this circumstance, the (complexity) polynomial hierarchy unexpectedly collapses to the first level.

If Conjecture 1.12 holds, Problem 1.9 and Question 1.6 are answered negatively. Moreover, a solution to Problem 1.8 proves $\mathsf{nonvanishing}(\chi_{\text{claw-free } G})$ is coNP, and hence NP = coNP.

This suggests a new complexity-theoretic rationale for the study of χ_G.

1.3 An algebraic combinatorics paradigm for complexity

Summarizing, we are motivated by complexity to study nonvanishing in algebraic combinatorics. Many polynomial families $\{F_\diamondsuit\}$ have combinatorial positivity and (conjecturally) SNP [16]. Together, with a solution to Problem 1.8, $\mathsf{nonvanishing} \in \mathsf{NP} \cap \mathsf{coNP}$.

For each family $\{F_\diamondsuit\}$, one arrives at one of four logical outcomes, depending on the complexity of $\mathsf{nonvanishing}(F_\diamondsuit)$ within $\mathsf{NP} \cap \mathsf{coNP}$:

(I) Unknown: it is a problem, in and of itself, to find additional problems that are in $\mathsf{NP} \cap \mathsf{coNP}$ that are not *known* to be in P.

(II) P: Give an algorithm. It will likely illuminate some special structure, of independent combinatorial interest.

(III) NP-complete: proof solves $\mathsf{NP} \stackrel{?}{=} \mathsf{coNP}$ with (a surprising) "=".

(IV) NP-intermediate: proof solves NP-intermediate$\stackrel{?}{=} \emptyset$ with "$\neq$" (hence $\mathsf{P} \neq \mathsf{NP}$).

Our main results in Section 2 illustrate (II) for Schubert polynomials.

2 Main results: Schubert polynomials

Schubert polynomials form a linear basis of all polynomials $\mathbb{Z}[x_1, x_2, x_3, \ldots]$. They were introduced by A. Lascoux–M.-P. Schützenberger [14] to study the cohomology ring of the flag manifold. These polynomials represent the Schubert classes under the Borel isomorphism. A reference is the textbook [9].

The Schubert polynomial $\mathfrak{S}_w(x_1, \ldots, x_n)$ is defined recursively for any permutation $w \in S_n$ as follows. If $w_0 = n\ n-1\ \cdots 2\ 1$ is the longest length permutation in S_n, then

$$\mathfrak{S}_{w_0}(x_1, \ldots, x_n) := x_1^{n-1} x_2^{n-2} \cdots x_{n-1}.$$

Otherwise, $w \neq w_0$ and there exists i such that $w(i) < w(i+1)$. Then one sets

$$\mathfrak{S}_w(x_1, \ldots, x_n) = \partial_i \mathfrak{S}_{ws_i}(x_1, \ldots, x_n),$$

where $\partial_i f := \frac{f - s_i f}{x_i - x_{i+1}}$, and s_i is the transposition swapping i and $i+1$. Since ∂_i satisfies

$$\partial_i \partial_j = \partial_j \partial_i \text{ for } |i-j| > 1, \text{ and } \partial_i \partial_{i+1} \partial_i = \partial_{i+1} \partial_i \partial_{i+1},$$

the above description of $\mathfrak{S}_w$ is well-defined. In addition, under the inclusion $\iota : S_n \hookrightarrow S_{n+1}$ defined by $w(1)\cdots w(n) \mapsto w(1)\ \cdots w(n)\ n+1$, $\mathfrak{S}_w = \mathfrak{S}_{\iota(w)}$. Thus one unambiguously refers to $\mathfrak{S}_w$ for each $w \in S_\infty = \bigcup_{n\geq 1} S_n$.

To each $w \in S_\infty$ there is a unique *code*, $\mathsf{code}(w) = (c_1, c_2, \ldots, c_L) \in \mathbb{Z}^L_{\geq 0}$, where c_i counts the number of boxes in the i-th row of the Rothe diagram $D(w)$ of w. If w is the identity then $\mathsf{code}(w) = \emptyset$; otherwise, $c_L > 0$ (i.e., we truncate any trailing zeroes).

Now, $c_{\alpha,w} = 0$ unless $\alpha_i = 0$ for $i > L$, and moreover, $c_{\alpha,w} \in \mathbb{Z}_{\geq 0}$. Let Schubert be the nonvanishing problem for Schubert polynomials. The `INPUT` is $\mathsf{code} = (c_1, \ldots c_L) \in \mathbb{Z}^L_{\geq 0}$ with $c_L > 0$ and $\alpha \in \mathbb{Z}^L_{\geq 0}$. Schubert returns `YES` if $c_{\alpha,w} > 0$ and `NO` otherwise.

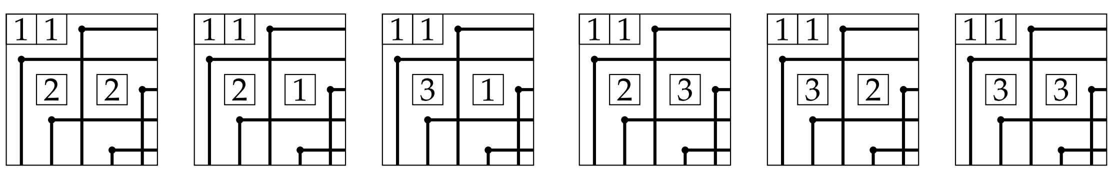
Theorem 2.1. Schubert $\in$ P.

We prove Theorem 2.1 using another result. For $w \in S_n$, let $\mathsf{PerfectTab}(D(w), \alpha)$ be the fillings of $D(w)$ with α_k many k's, where entries in each column are distinct, and any entry in row i is $\leq i$. Let $\mathsf{PerfectTab}_<(D(w), \alpha) \subseteq \mathsf{PerfectTab}(D(w), \alpha)$ be fillings where entries in each column increase from top to bottom.

Theorem 2.2. $c_{\alpha,w} > 0 \iff \mathsf{PerfectTab}(D(w), \alpha) \neq \emptyset \iff \mathsf{PerfectTab}_<(D(w), \alpha) \neq \emptyset$

In general $\#\mathsf{PerfectTab}(D(w), \alpha) \neq c_{\alpha,w}$ but rather $\#\mathsf{PerfectTab}(D(w), \alpha) \geq c_{\alpha,w}$ (cf. [8]).

Example 2.3. *Here are the tableaux in* $\bigcup_\alpha \mathsf{PerfectTab}_<(D(31524), \alpha)$*:*



Hence, for instance, $c_{(2,1,1),31524} > 0$ *but* $c_{(4),31524} = 0$.

To prove Theorems 2.1 and 2.2 we establish more general results about the *Schubitope* introduced in [16]. This polytope $\mathcal{S}_D$ generalizes the λ-permutahedron of Example 1.5. It is defined with a halfspace description for any diagram of boxes $D \subseteq [n]^2$.

In the case of Rothe diagrams $D := D(w)$, it was conjectured in [16] that $\mathcal{S}_{D(w)}$ is the Newton polytope of $\mathfrak{S}_w$ and moreover that $\mathfrak{S}_w$ has the SNP property. These conjectures were proved by A. Fink-K. Mészáros-A. St. Dizier [7]. This, combined with Theorem 3.5 and properties of perfect tableaux, proves Theorem 2.2.

Key polynomials κ_β are a specialization of the non-symmetric Macdonald polynomials. Similarly to the above case, for skyline diagrams $D := D_\beta$, [16] conjectured that $\mathcal{S}_{D_\beta}$ is the Newton polytope of κ_β and moreover that κ_β are SNP; this is proved in [7]. Nonvanishing is also in P, provable using results of Section 4 in a manner analogous to that used for the Schubert polynomials.

3 The Schubitope

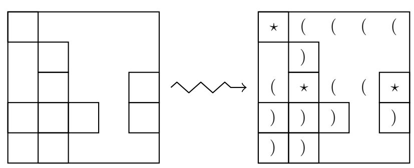
Fix $n \in \mathbb{Z}_{>0}$ and let $D \subseteq [n]^2$. We call D a *diagram* and visualize D as a subset of an $n \times n$ grid of boxes, oriented so that $(r, c) \in [n]^2$ represents the box in the rth row from the top and the cth column from the left. Given $S \subseteq [n]$ and a column $c \in [n]$, construct a string denoted $\mathsf{word}_{c,S}(D)$ by reading column c from top to bottom and recording

- (if $(r, c) \notin D$ and $r \in S$,
-) if $(r, c) \in D$ and $r \notin S$, and
- $\star$ if $(r, c) \in D$ and $r \in S$.

Let $\theta^c_D(S) = \#\{\star\text{'s in } \mathsf{word}_{c,S}(D)\} + \#\{\text{paired ()'s in } \mathsf{word}_{c,S}(D)\}$ and

$$\theta_D(S) = \sum_{c=1}^{n} \theta^c_D(S).$$

Example 3.1. *In the diagram D below, we labelled the corresponding strings for* $\mathsf{word}_{c,S}(D)$ *for* $S = \{1, 3\}$. *For instance, we see* $\mathsf{word}_{5,\{1,3\}}(D) = (\star)$.



The *Schubitope* $\mathcal{S}_D$, as defined in [16], is the polytope

$$\left\{ (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n_{\geq 0} : \alpha_1 + \cdots + \alpha_n = \#D \text{ and } \sum_{i \in S} \alpha_i \leq \theta_D(S) \text{ for all } S \subseteq [n] \right\}. \tag{3.1}$$

3.1 Characterizations via tableaux

A *tableau* of *shape* D is a map

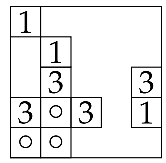
$$\tau : D \to [n] \cup \{\circ\},$$

where $\tau(r, c) = \circ$ indicates that the box (r, c) is unlabelled. Let $\mathsf{Tab}(D)$ denote the set of such tableaux. One of the ideas in our proofs is to reformulate the original definition of $\theta_D(S)$ into the language of tableaux. Given $S \subseteq [n]$, define $\pi_{D,S} \in \mathsf{Tab}(D)$ by

$$\pi_{D,S}(r, c) := \begin{cases} r & \text{if } (r, c) \text{ contributes a “}\star\text{” to } \mathsf{word}_{c,S}(D), \\ s & \text{if } (r, c) \text{ contributes a “)” to } \mathsf{word}_{c,S}(D) \text{ which is} \\ & \text{paired with an “(” from } (s, c), \\ \circ & \text{otherwise.} \end{cases} \tag{3.2}$$

In (3.2) we pair by the standard “inside-out” convention.

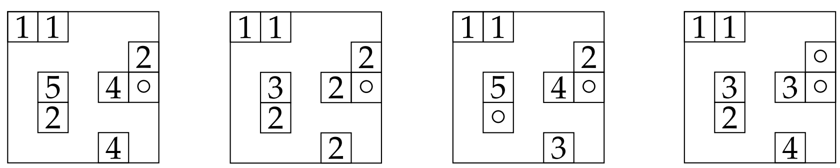
Example 3.2. *Continuing Example 3.1, below is* $\pi_{D,\{1,3\}}(D)$.



Theorem 3.3. *Let* $D \subseteq [n]^2$ *and* $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$ *with* $\alpha_1 + \cdots + \alpha_n = \#D$. *Then* $\alpha \in \mathcal{S}_D$ *if and only if for each* $S \subseteq [n]$, $\sum_{i \in S} \alpha_i \leq \#\pi^{-1}_{D,S}(S)$.

Define $\tau \in \mathsf{Tab}(D)$ to be *flagged* if $\tau(r,c) \leq r$ whenever $\tau(r,c) \neq \circ$. It is *column-injective* if $\tau(r,c) \neq \tau(r',c)$ whenever $r \neq r'$ and $\tau(r,c) \neq \circ$.

Example 3.4. *Of the tableaux of shape D below, only the second and fourth are flagged, and only the third and fourth are column-injective.*



Further, call a tableau $\tau \in \mathsf{Tab}(D)$ *perfect* if τ is flagged, column-injective, and if no boxes are left unlabelled, i.e., $\tau^{-1}(\{\circ\}) = \emptyset$. Say $\tau \in \mathsf{Tab}(D)$ has *content* α if $\#\tau^{-1}(\{i\}) = \alpha_i$ for each $i \in [n]$. Let $\mathsf{PerfectTab}(D, \alpha)$ denote the set of perfect tableaux of content α.

We use Theorem 3.3 to prove:

Theorem 3.5. *Let* $D \subseteq [n]^2$ *and* $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$. *Then* $\alpha \in \mathcal{S}_D$ *if and only if* $\mathsf{PerfectTab}(D, \alpha) \neq \emptyset$.

4 Polytopal descriptions of perfect tableaux

By Theorem 3.5, to decide $\alpha \in \mathcal{S}_D$, it suffices to determine $\mathsf{PerfectTab}(D, \alpha) \neq \emptyset$. Thus it remains to analyze the complexity of deciding $\mathsf{PerfectTab}(D, \alpha) \neq \emptyset$.

For this, we construct a polytope that characterizes $\mathsf{PerfectTab}(D, \alpha)$. Given $D \subseteq [n]^2$ and $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$, define

$$\mathcal{P}(D, \alpha) \subseteq \mathbb{R}^{n^2}$$

to be the polytope with points of the form $(\alpha_{ij})_{i,j \in [n]} = (\alpha_{11}, \ldots, \alpha_{n1}, \ldots, \alpha_{1n}, \ldots, \alpha_{nn})$ governed by the inequalities (A)-(C) below.

(A) Column-Injectivity Conditions: For all $i, j \in [n]$,

$$0 \leq \alpha_{ij} \leq 1.$$

(B) Content Conditions: For all $i \in [n]$,

$$\sum_{j=1}^{n} \alpha_{ij} = \alpha_i.$$

(C) Flag Conditions: For all $s, j \in [n]$,

$$\sum_{i=1}^{s} \alpha_{ij} \geq \#\{(i,j) \in D : i \leq s\}.$$

Theorem 4.1. *Let $D \subseteq [n]^2$ and $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$. Then $\mathsf{PerfectTab}(D, \alpha) \neq \emptyset$ if and only if $\alpha_1 + \cdots + \alpha_n = \#D$ and $\mathcal{P}(D, \alpha) \cap \mathbb{Z}^{n^2} \neq \emptyset$.*

Theorem 4.1 formulates the problem of determining if $\mathsf{PerfectTab}(D, \alpha) \neq \emptyset$ in terms of feasibility of an integer linear programming problem. In general, integral feasibility is NP-complete. However, $\mathcal{P}(D, \alpha)$ is totally unimodular. Thus feasibility of the problem is equivalent to feasibility of its LP-relaxation:

Theorem 4.2. *Let $D \subseteq [n]^2$ and $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}^n$ with $\alpha_1 + \cdots + \alpha_n = \#D$. Then $\mathcal{P}(D, \alpha) \cap \mathbb{Z}^{n^2} \neq \emptyset$ if and only if $\mathcal{P}(D, \alpha) \neq \emptyset$.*

By combining Theorems 3.5, 4.1 and 4.2 with the fact that $\mathcal{P}(D, \alpha)$ has a polynomial size halfspace description, it follows that $\alpha \in \mathcal{S}_D$ can be decided in $n^{O(1)}$-time. However, this result can be improved. If $D \subseteq [n]^2$ has many identical columns, then many of the flag conditions (C) will look essentially the same. Therefore, our final goal will be to construct a “compressed" version of $\mathcal{P}(D, \alpha)$ that removes some of the repetitive inequalities.

A tuple $\mathcal{C} = (m, \{P_k\}_{k=1}^{\ell})$ is a *compression* of $D \subseteq [n]^2$ if:

- $m \leq n$ is a nonnegative integer such that $(r, p) \notin D$ for $r > m$ and $p \in [n]$, and
- $P = P_1 \dot\cup \cdots \dot\cup P_\ell \subseteq [n]$ such that if $p, p' \in P_k$ then

  $$\{r \in [n] : (r, p) \in D\} = \{r \in [n] : (r, p') \in D\},$$

  and moreover if D is nonempty in column p then $p \in P_k$ for some $k \in [\ell]$.

For a compression $\mathcal{C}$ of $D \subseteq [n]^2$ and $\tilde{\alpha} = (\tilde{\alpha}_1, \ldots, \tilde{\alpha}_m) \in \mathbb{Z}^m_{\geq 0}$ define

$$\mathcal{Q}(D, \mathcal{C}, \tilde{\alpha}) \subseteq \mathbb{R}^{m\ell} \tag{4.1}$$

to be the polytope with points of the form $(\tilde{\alpha}_{ik})_{i \in [m], k \in [\ell]}$ satisfying (A′)-(C′) below.

(A′) Column-Injectivity Conditions: For all $i \in [m], k \in [\ell]$,

$$0 \leq \tilde{\alpha}_{ik} \leq 1.$$

(B′) Content Conditions: For all $i \in [m]$,

$$\sum_{k=1}^{\ell} \#P_k \cdot \tilde{\alpha}_{ik} = \alpha_i.$$

(C′) Flag Conditions: For all $s \in [m], k \in [\ell]$,

$$\sum_{i=1}^{s} \tilde{\alpha}_{ik} \geq \#\{(i, p_k) \in D : i \leq s, p_k := \min P_k\}.$$

Theorem 4.3. *Let* $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$ *and* $\tilde{\alpha} = (\tilde{\alpha}_1, \ldots, \tilde{\alpha}_m) := (\alpha_1, \ldots, \alpha_m)$. *Then* $\alpha_1 + \cdots + \alpha_n = \#D$ *and* $\mathcal{P}(D, \alpha) \neq \emptyset$ *if and only if* $\alpha_1 + \cdots + \alpha_m = \#D$, $\alpha_{m+1} = \cdots = \alpha_n = 0$, *and* $\mathcal{Q}(D, \mathcal{C}, \tilde{\alpha}) \neq \emptyset$.

4.1 Deciding membership in the Schubitope

We use the above results to give a polynomial time algorithm to check if a lattice point is in the Schubitope. This more general result gives a polynomial time algorithm for any polynomial family whose Newton polytopes are Schubitopes. Let $D \subseteq [n]^2$, and fix a compression $\mathcal{C} = (m, \{P_k\}_{k=1}^{\ell})$ of D.

Theorem 4.4. *Let* $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$. *Then* $\alpha \in \mathcal{S}_D$ *if and only if* $\alpha_1 + \cdots + \alpha_m = \#D$, $\alpha_{m+1} = \cdots = \alpha_n = 0$, *and* $\mathcal{Q}(D, \mathcal{C}, \tilde{\alpha}) \neq \emptyset$, *where* $\tilde{\alpha} = (\tilde{\alpha}_1, \ldots, \tilde{\alpha}_m) := (\alpha_1, \ldots, \alpha_m)$.

For each $k \in [\ell]$, let $p_k := \min P_k$ and $R_k(\mathcal{C}) := \{r \in [n] : (r, p_k) \in D\} \subseteq [m]$.

Theorem 4.5. *Given as input* $\{R_k(\mathcal{C})\}_{k=1}^{\ell}$, $\{\#P_k\}_{k=1}^{\ell}$, *and* $\tilde{\alpha} = (\tilde{\alpha}_1, \ldots, \tilde{\alpha}_m) \in \mathbb{Z}^m_{\geq 0}$ *satisfying* $\tilde{\alpha}_1 + \cdots + \tilde{\alpha}_m = \#D$, *one can decide if* $\alpha := (\tilde{\alpha}_1, \ldots, \tilde{\alpha}_m, 0, \ldots, 0) \in \mathbb{Z}^n_{\geq 0}$ *lies in* $\mathcal{S}_D$ *in polynomial time in* m *and* ℓ.

5 Application to $D(w)$: proof of Theorems 2.1 and 2.2

For the specialization to Rothe diagrams $D := D(w)$, the results of A. Fink-K. Mészáros-A. St. Dizier [7] imply

$$\alpha \in \mathcal{S}_{D(w)} \iff c_{\alpha,w} > 0.$$

Combining this with Theorem 3.5,

$$c_{\alpha,w} > 0 \iff \mathsf{PerfectTab}(D(w), \alpha) \neq \emptyset.$$

Further, if $\mathsf{PerfectTab}(D(w), \alpha) \neq \emptyset$, we can find $\tau \in \mathsf{PerfectTab}(D(w), \alpha)$ which is also strictly increasing along columns. Thus $\mathsf{PerfectTab}_{>}(D(w), \alpha) \neq \emptyset$, and Theorem 2.2 follows.

To obtain Theorem 2.1 we apply the results of Section 4 to $D(w)$. Suppose $\mathsf{code}(w) = (c_1, \ldots, c_L)$. Let $\sigma \in S_L$ be such that

$$w(\sigma(1)) < w(\sigma(2)) < \ldots < w(\sigma(L)).$$

Set $w(\sigma(0)) := 0$. The key lemma we need is:

Lemma 5.1. *For $1 \leq h \leq L$, and for all*

$$j_1, j_2 \in \{w(\sigma(h-1)) + 1, w(\sigma(h-1)) + 2, \ldots, w(\sigma(h)) - 1\},$$

we have $(i, j_1) \in D(w)$ if and only if $(i, j_2) \in D(w)$.

Using Lemma 5.1, there exists a compression $\mathcal{C} = (L, \{P_k\}_{k=1}^{\ell})$ of $D(w)$ where $\ell \leq 2L$. With the following statement, Theorem 4.5 proves Theorem 2.1.

Proposition 5.2. *There exists an $O(L^2)$-time algorithm to compute $\mathcal{C}$, $\{\#P_k\}_{k=1}^{\ell}$, and $\{R_k\}_{k=1}^{\ell}$ from the input $\mathsf{code}(w) = (c_1, \ldots, c_L)$.*

Acknowledgements

We acknowledge Mathoverflow, S. Kintali's blog "My brain is open," and R. O'Donnell's YouTube videos (from his class at Carnegie Mellon) for background. We thank Philipp Hieronymi, Alexandr Kostochka, Cara Monical, Erik Walsberg, Douglas West, Alexander Woo, and the anonymous referees for their helpful comments.

References

[1] A. Adve, C. Robichaux, and A. Yong. "Complexity, combinatorial positivity, and Newton polytopes". 2018. arXiv:1810.10361.

[2] M. Agrawal, N. Kayal, and N. Saxena. "PRIMES is in P". *Ann. of Math. (2)* **160**.2 (2004), pp. 781–793. Link.

[3] D. Aharonov and O. Regev. "Lattice problems in $\mathsf{NP} \cap \mathsf{coNP}$". *J. ACM* **52**.5 (2005), pp. 749–765. Link.

[4] S. Billey, W. Jockusch, and R. P. Stanley. "Some combinatorial properties of Schubert polynomials". *J. Algebraic Combin.* **2**.4 (1993), pp. 345–374. Link.

[5] A. Condon. "The Complexity of stochastic games". *Inf. Comput.* **96**.2 (1992), pp. 203–224. Link.

[6] G. B. Dantzig. "Reminiscences about the origins of linear programming". *Technical report of the Systems Optimization Laboratory (Stanford U.)* **81**.5 (1981).

[7] A. Fink, K. Mészáros, and A. St. Dizier. "Schubert polynomials as integer point transforms of generalized permutahedra". *Adv. Math.* **332** (2018), pp. 465–475. Link.

[8] S. Fomin, C. Greene, V. Reiner, and M. Shimozono. "Balanced labellings and Schubert polynomials". *European J. Combin.* **18**.4 (1997), pp. 373–389. Link.

[9] W. Fulton. *Young Tableaux. With applications to representation theory and geometry*. London Mathematical Society Student Texts 35. Cambridge University Press, 1997.

[10] K. Gödel. "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I." *Monatshefte für Mathematik und Physik* **38** (1931), pp. 173–198.

[11] O. Goldreich. *P, NP, and NP-completeness: The Basics of Computational Complexity*. Cambridge University Press, 2010.

[12] I. Holyer. "The NP-completeness of some edge-partition problems". *SIAM J. Comput.* **10**.4 (1981), pp. 713–717. Link.

[13] M. Jurdziński. "Deciding the winner in parity games is in UP $\cap$ co-UP". *Inf. Process. Lett.* **68**.3 (1998), pp. 119–124. Link.

[14] A. Lascoux and M. P. Schützenberger. "Polynômes de Schubert". *C. R. Acad. Sci. Paris Sér. I Math.* **294** (1982), pp. 447–450.

[15] C. Monical. "Polynomials in algebraic combinatorics". PhD thesis. University of Illinois at Urbana-Champaign, 2018.

[16] C. Monical, N. Tokcan, and A. Yong. "Newton polytopes in algebraic combinatorics". 2017. arXiv:1703.025843.

[17] V. Pratt. "Every prime has a succinct certificate". *SIAM J. Comput.* **4** (1975), pp. 214–220. Link.

[18] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, 1986.

[19] R. P. Stanley. "A symmetric function generalization of the chromatic polynomial of a graph". *Adv. Math.* **111**.1 (1995), pp. 166–194. Link.

[20] A. Turing. "On Computable Numbers, with an Application to the Entscheidungs-problem. A Correction." *Proc. London Math. Soc.* **43** (1937), pp. 544–546. Link.