

ON SEMIRING COMPLEXITY OF SCHUR POLYNOMIALS

SERGEY FOMIN, DIMA GRIGORIEV,
DORIAN NOGNENG, AND ÉRIC SCHOST

May 17, 2018

Abstract. Semiring complexity is the version of arithmetic circuit complexity that allows only two operations: addition and multiplication. We show that semiring complexity of a Schur polynomial $s_\lambda(x_1, \dots, x_k)$ labeled by a partition $\lambda = (\lambda_1 \geq \lambda_2 \geq \dots)$ is bounded by $O(\log(\lambda_1))$ provided the number of variables k is fixed.

Keywords. Semiring complexity, Schur function, Young tableaux.

Subject classification. *2010 Mathematics Subject Classification* Primary 68Q25, Secondary 05E05.

1. Introduction and main results

Let $f(x_1, \dots, x_k)$ be a polynomial with nonnegative integer coefficients. As such, f can be computed using addition and multiplication only—without subtraction or division. To be more precise, one can build an *arithmetic circuit* wherein

- each gate performs an operation of addition or multiplication;
- the inputs are x_1, \dots, x_k , possibly along with some positive integer scalars;
- the sole output is $f(x_1, \dots, x_k)$.

The *semiring complexity* (or $\{+, \times\}$ -*complexity*) of f is the smallest size of (i.e., the smallest number of gates in) such an arithmetic circuit. This notion is illustrated in [Figure 1.1](#). For additional details, see [Fomin *et al.* \(2016, Section 2\)](#) and references therein.

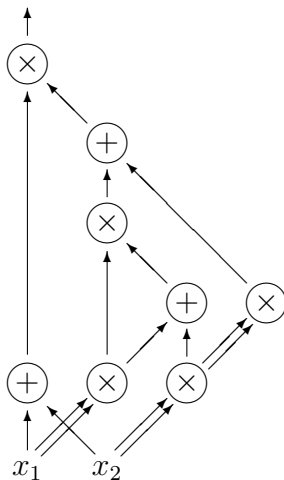


Figure 1.1: The smallest $\{+, \times\}$ -circuit computing the polynomial $f(x_1, x_2) = h_5(x_1, x_2) = x_1^5 + x_1^4x_2 + x_1^3x_2^2 + x_1^2x_2^3 + x_1x_2^4 + x_2^5$. This circuit utilizes the formula $h_5(x_1, x_2) = (x_1 + x_2)(x_1^2(x_1^2 + x_2^2) + x_2^4)$.

This paper is devoted to the problem of determining semiring complexity of symmetric polynomials. More specifically, we focus our attention on Schur functions, an important class of symmetric polynomials which play prominent roles in several branches of mathematics; see, e.g., [Macdonald \(2015, Chapter I\)](#) and [Stanley \(1999, Chapter 7\)](#).

Let $\lambda = (\lambda_1 \geq \lambda_2 \geq \dots \geq 0)$ be an integer partition. The *Schur function* (or *Schur polynomial*) $s_\lambda(x_1, \dots, x_k)$ is a symmetric polynomial of degree $|\lambda| = \sum_i \lambda_i$ in the variables x_1, \dots, x_k which can be defined in many different ways. One remarkable feature of Schur polynomials that makes them an exciting object of study in algebraic complexity theory is that the classical formulas defining them fall into two categories. On the one hand, there are determinantal expressions (e.g., the Jacobi-Trudi formula or the bialternant formula) which provide efficient ways to compute Schur functions in an unrestricted setting, i.e., when all arithmetic operations are allowed. On the other hand, Schur functions are generating functions for semistandard Young tableaux. This description represents them as polynomials with manifestly positive coefficients; so they can be computed using addition and multi-

plication only. We note however that the naïve approach based on these monomial expansions yields algorithms whose (semiring) complexity is very high—and indeed very far from the optimum.

Our main result is the following. (We use the notation $\lambda' = (\lambda'_1 \geq \lambda'_2 \geq \dots)$ for the partition conjugate to λ .)

THEOREM 1.1. *The semiring complexity of a Schur polynomial $s_\lambda(x_1, \dots, x_k)$ labeled by partition $\lambda = (\lambda_1 \geq \dots \geq \lambda_\ell)$ is at most $O(\log(\lambda_1)k^5 2^{k\ell} \ell^d)$ where $d = \max_j \lambda'_j(k - \lambda'_j)$.*

Since $\ell \leq k$ (or else $s_\lambda(x_1, \dots, x_k) = 0$) and $d \leq \frac{k^2}{4}$, we obtain:

COROLLARY 1.2. *The semiring complexity of a Schur polynomial $s_\lambda(x_1, \dots, x_k)$ is bounded from above by $k^{k^2(\frac{1}{4} + o(1))} O(\log(\lambda_1))$. If the number of variables k is fixed, this complexity is $O(\log(\lambda_1))$.*

REMARK 1.3. *The problem of designing efficient algorithms employing addition and multiplication arises naturally in the context of numerical computation, as these algorithms have valuable stability properties. Motivated by such considerations, Demmel & Koev (2006) developed $\{+, \times\}$ -algorithms for computing Schur polynomials using a dynamic programming approach. In the notation of Theorem 1.1, Proposition 5.3 in loc. cit. asserts that the semiring complexity of $s_\lambda(x_1, \dots, x_k)$ is bounded from above by $O(e^{5.2\sqrt{|\lambda|}} \ell k)$. When k is fixed, and the shape λ grows, this bound is much larger than the one in Corollary 1.2. On the other hand, in the regime where λ is fixed and the number of variables k grows, the complexity of the Demmel-Koev algorithm is linear in k whereas the bound in Theorem 1.1 is exponential in k . It would be interesting to find a common generalization of these results.*

We prove Theorem 1.1 in two stages. At the first stage (see Section 3), we treat a special case where partition λ has only one (nonzero) part. More explicitly, we obtain the following result.

Recall that the complete homogeneous symmetric polynomial

$$h_n(x_1, \dots, x_k) = \sum_{1 \leq i_1 \leq \dots \leq i_n \leq k} x_{i_1} \cdots x_{i_n}$$

is the sum of all monomials of degree n in the variables x_1, \dots, x_k . See an example in [Figure 1.1](#).

THEOREM 1.4. *The semiring complexity of a complete homogeneous symmetric polynomial $h_n(x_1, \dots, x_k)$ is $O(k^2 \log(n))$.*

Our proof of [Theorem 1.1](#), presented in [Section 6](#), relies on three main ingredients:

- [Theorem 1.4](#);
- a formula expressing a multichain-generating function of a *shellable poset* in terms of complete homogeneous polynomials, see [Section 4](#); and
- a representation of a Schur polynomial as a multichain generating function, or more precisely an iterated sum thereof, see [Section 5](#).

2. Related problems

The general problem of determining the semiring complexity of a Schur polynomial is open. In particular, the following tantalizing problem remains out of reach.

PROBLEM 2.1 ([Fomin et al. 2016](#), Problem 3.2). *Is the semiring complexity of $s_\lambda(x_1, \dots, x_k)$ bounded by a polynomial in k and $|\lambda|$?*

REMARK 2.2. *A general method for obtaining lower bounds on semiring complexity was suggested by [Schnorr \(1976\)](#). Schnorr's bound only depends on the support of a polynomial, i.e., on the set of monomials that contribute with a positive coefficient. Schnorr's argument was further refined by [Shamir & Snir \(1977\)](#); powerful applications were given by [Jerrum & Snir \(1982\)](#). As mentioned in [Fomin et al. \(2016, Remark 3.3\)](#), Schnorr-type lower bounds are useless in the case of Schur functions since computing a Schur function is difficult not because of its support but because of the complexity of its coefficients (the Kostka numbers). The problem of computing an individual Kostka number is known to be $\#\mathbf{P}$ -complete ([Narayanan 2006](#)) whereas the support of a Schur function is very easy to determine.*

REMARK 2.3. [Fomin et al. \(2016\)](#) investigated the notion of semiring complexity alongside other similar computational models involving restricted sets of arithmetic operations. In brief, the results obtained in *loc. cit.*, together with [Jerrum & Snir \(1982\)](#) and [Valiant \(1980\)](#), demonstrate that adjoining subtraction and/or division to the two-element set $\{+, \times\}$ of allowed arithmetic operations can, in some cases, dramatically decrease computational complexity. (By contrast, removing division from $\{+, -, \times, \div\}$ comes at merely polynomial cost, as shown by [Strassen \(1973\)](#).) We refer the reader to [Fomin et al. \(2016\)](#) for the discussion of these issues.

REMARK 2.4. In the unrestricted model, one can compute a Schur polynomial $s_\lambda(x_1, \dots, x_k)$ in time polynomial in k and $\log(\lambda_1)$, via the bialternant formula ([Stanley 1999](#), Section 7.15), and using repeated squaring to compute the powers of variables appearing in the relevant determinants.

One important complexity model studied in [Fomin et al. \(2016\)](#) is *subtraction-free complexity*, which allows the operations of addition, multiplication, and division. It turns out that subtraction-free complexity of a Schur function is indeed polynomial:

THEOREM 2.5 ([Koev 2007](#), Section 6, [Chan et al. 2008](#), Section 4, [Fomin et al. 2016](#), Theorem 3.1). *Subtraction-free complexity of a Schur polynomial $s_\lambda(x_1, \dots, x_k)$ is at most $O(n^3)$ where $n = k + \lambda_1$.*

The algorithms presented in *loc. cit.* utilize division in essential ways, so they do not bring us any closer to the resolution of [Problem 2.1](#).

Since subtraction-free complexity is bounded from above by semiring complexity, [Theorem 1.1](#) implies that the subtraction-free complexity of a particular Schur polynomial $s_\lambda(x_1, \dots, x_k)$ can be much smaller (for small k) than the upper bound of [Theorem 2.5](#).

PROBLEM 2.6. *Find a natural upper bound on subtraction-free complexity of a Schur polynomial that simultaneously strengthens [Theorem 1.1](#) and [Theorem 2.5](#).*

REMARK 2.7. *Grigoriev & Koshevoy (2016)* gave an exponential lower bound on the $\{+, \times\}$ -complexity of a monomial symmetric function.

3. Semiring complexity of complete homogeneous polynomials

In this section, we prove [Theorem 1.4](#). We fix k , and use the notation

$$\begin{aligned} h_m &= h_m(x_1, \dots, x_k) = \sum_{1 \leq i_1 < \dots < i_m \leq k} x_{i_1} \cdots x_{i_m}, \\ \tilde{h}_m &= h_m(x_1^2, \dots, x_k^2), \\ e_m &= e_m(x_1, \dots, x_k) = \sum_{1 \leq i_1 < \dots < i_m \leq k} x_{i_1} \cdots x_{i_m}. \end{aligned}$$

LEMMA 3.1. *One can compute the polynomials h_{n-k+1}, \dots, h_n from $\tilde{h}_{\lfloor \frac{n}{2} \rfloor - k + 1}, \dots, \tilde{h}_{\lfloor \frac{n}{2} \rfloor}$ and e_1, \dots, e_k , using $O(k^2)$ additions and multiplications.*

PROOF. The key algebraic observation is that

$$\sum_{m \geq 0} h_m t^m = \prod_{i=1}^k \frac{1}{1-x_i t} = \prod_{i=1}^k (1+x_i t) \prod_{i=1}^k \frac{1}{1-x_i^2 t^2} = \sum_{a=0}^k e_a t^a \sum_{b \geq 0} \tilde{h}_b t^{2b}$$

and consequently

$$(3.2) \quad h_m = \sum_{m-k \leq 2b \leq m} e_{m-2b} \tilde{h}_b.$$

For $n-k+1 \leq m \leq n$, the indices b appearing on the right-hand side of (3.2) satisfy $b \leq \lfloor \frac{m}{2} \rfloor \leq \lfloor \frac{n}{2} \rfloor$ and $b \geq \lceil \frac{m-k}{2} \rceil \geq \lceil \frac{n-2k+1}{2} \rceil = \lfloor \frac{n}{2} \rfloor - k + 1$. Thus we can use (3.2) to compute these h_m ; this takes $O(k)$ operations for each of the k values of m . \square

LEMMA 3.3. *One can compute e_1, \dots, e_k using $O(k^2)$ additions and multiplications.*

PROOF. The requisite algorithm is obtained by iterating the Pascal-type recurrence

$$e_m(x_1, \dots, x_j) = x_j e_{m-1}(x_1, \dots, x_{j-1}) + e_m(x_1, \dots, x_{j-1}). \quad \square$$

We note that in the unrestricted model, the complexity of computing e_1, \dots, e_k is of the order $k \log(k)$, see [Strassen \(1972/73\)](#).

PROOF ([Theorem 1.4](#)). Let $T(n)$ denote the semiring complexity of computing h_{n-k+1}, \dots, h_n . [Lemma 3.1](#) and [Lemma 3.3](#) imply that $T(n) \leq T(\lfloor \frac{n}{2} \rfloor) + O(k^2)$. (Squaring the variables x_1, \dots, x_k , which is needed to compute the \tilde{h}_b 's, takes linear time.) We conclude that $T(n) = O(k^2 \log(n))$, as desired. \square

4. Linear orderings of maximal chains in partially ordered sets

DEFINITION 4.1 (*Poset, chain, proper ordering*). Let \mathbf{P} be a finite graded partially ordered set (poset) with a unique minimal element $\hat{0}$ and a unique maximal element $\hat{1}$. A linearly ordered subset of \mathbf{P} is called a chain. We denote by $\text{MaxChains}(\mathbf{P})$ the set of all maximal (by inclusion) chains in \mathbf{P} . Under the above assumptions, all chains in $\text{MaxChains}(\mathbf{P})$ have the same cardinality m .

Let us fix a linear ordering on $\text{MaxChains}(\mathbf{P})$, and write $Q' \prec Q$ to denote that Q' (strictly) precedes Q in this order. For $Q \in \text{MaxChains}(\mathbf{P})$, we denote

$$(4.2) \quad Q^* \stackrel{\text{def}}{=} \{c \in Q \mid Q - \{c\} \subset Q' \text{ for some } Q' \prec Q\}.$$

Thus Q^* consists of those elements of a maximal chain Q which can be replaced by another element so that the resulting maximal chain precedes Q . We call a linear ordering of $\text{MaxChains}(\mathbf{P})$ proper if for any $Q \in \text{MaxChains}(\mathbf{P})$, none of the chains preceding Q contains Q^* :

$$(4.3) \quad Q' \prec Q \implies Q' \not\supset Q^*.$$

REMARK 4.4. *In algebraic/geometric combinatorics, the notions introduced in Definition 4.1 are traditionally described in the language of simplicial complexes and their shellings; see, e.g., Wachs (2007) for an introduction to this subject. In this paper, we try to avoid this terminology in order to keep the exposition self-contained. The brief comments below are intended for the readers interested in the broader combinatorial context, and will not be relied upon in the sequel.*

The order complex of \mathbf{P} is the simplicial complex on the ground set \mathbf{P} whose simplices are the chains in \mathbf{P} . The maximal simplices of the order complex are the maximal chains. A linear ordering of $\text{MaxChains}(\mathbf{P})$ is called a shelling (of the order complex) if for any $Q \in \text{MaxChains}(\mathbf{P})$, the subcomplex of the order complex formed by the simplices Q' with $Q' \prec Q$ (or more precisely the geometric realization of this subcomplex) intersects (the geometric realization of) the maximal simplex Q at a union of codimension 1 faces of Q . It is well known—and not hard to see—that any shelling order is proper, in the sense of Definition 4.1. More concretely, the subchain $Q^* \subset Q$ defined via (4.2) can be seen to coincide with the complement (inside Q) of the intersection of the aforementioned codimension 1 faces. Put differently, Q^* is the unique smallest face of Q not contained in the subcomplex $\bigcup_{Q' \prec Q} Q'$.

Our use of the notion of a proper ordering of maximal chains will rely on the following key lemma.

LEMMA 4.5. *Let \mathbf{P} be a poset with a proper linear ordering on $\text{MaxChains}(\mathbf{P})$, as in Definition 4.1. For a chain C and a maximal chain Q , the following are equivalent:*

- (i) Q is the smallest maximal chain containing C (with respect to the linear ordering on $\text{MaxChains}(\mathbf{P})$);
- (ii) $Q^* \subset C \subset Q$ (recall that Q^* is defined by (4.2)).

PROOF. First assume that (i) holds. Let $\mathbf{c} \in Q^*$. If $\mathbf{c} \notin C$, then C is contained in some maximal chain $Q' \prec Q$ (see (4.2)), contradicting (i).

Going in the opposite direction, assume that $Q^* \subset C \subset Q$. Suppose there exists a maximal chain $Q' \prec Q$ containing C . Then $Q' \supset Q^*$, contradicting (4.3). \square

DEFINITION 4.6 (*Multichain, support*). A “weakly increasing” sequence

$$M = \{p_1 \leq \dots \leq p_m\} \subset \mathbf{P}$$

is called a multichain of size m ; we write $|M| = m$. The elements of \mathbf{P} which appear in M (with nonzero multiplicity) form the support of M , denoted by $\text{supp}(M)$. The support of a multichain is a chain.

Let us associate a formal variable $z_{\mathbf{c}}$ with each element $\mathbf{c} \in \mathbf{P}$. For a multiset M of elements in \mathbf{P} , we denote by \mathbf{z}^M the corresponding monomial: $\mathbf{z}^M = \prod_{\mathbf{c} \in M} z_{\mathbf{c}}$.

LEMMA 4.7. Let \mathbf{P} be a poset endowed with a proper linear ordering of its maximal chains, see Definition 4.1. (Or: assume that a shelling of the order complex of \mathbf{P} is given.) Then the generating function for the multichains of size m in \mathbf{P} is given by

$$(4.8) \quad \sum_{\substack{\text{multichain } M \\ |M|=m}} \mathbf{z}^M = \sum_{Q \in \text{MaxChains}(\mathbf{P})} \mathbf{z}^{Q^*} h_{m-|Q^*|}((z_{\mathbf{c}})_{\mathbf{c} \in Q}),$$

with Q^* defined by (4.2).

PROOF. By Lemma 4.5, the set of chains in \mathbf{P} splits into the disjoint union of (poset-theoretic) intervals of the form $[Q^*, Q]$. Categorizing the multichains M by their support, and applying this observation to $C = \text{supp}(M)$, we obtain the identity

$$\sum_{\substack{\text{multichain } M \\ |M|=m}} \mathbf{z}^M = \sum_{Q \in \text{MaxChains}(\mathbf{P})} \sum_{\substack{Q^* \subset \text{supp}(M) \subset Q \\ |M|=m}} \mathbf{z}^M,$$

which readily implies (4.8). \square

In Section 5, we will relate Schur polynomials to a special case of the above construction involving a class of (shellable) posets $\mathbf{P}_{h,k}$ described in Definition 4.9 below. These posets have been

extensively studied in algebraic combinatorics, due to the role they play in representation theory and the classical Schubert Calculus. In particular, $\mathbf{P}_{h,k}$ describes the attachment of Schubert cells in the Grassmann manifold $\text{Gr}(h, k)$.

DEFINITION 4.9 (*Posets $\mathbf{P}_{h,k}$*). Let h and k be positive integers, with $h \leq k$. We denote by $\mathbf{P}_{h,k}$ the poset whose elements are column vectors (or simply columns) \mathbf{c} of height h whose entries lie in the set $\{1, \dots, k\}$ and strictly increase downwards:

$$(4.10) \quad \mathbf{c} = \begin{bmatrix} c_1 \\ \vdots \\ c_h \end{bmatrix} \in \mathbb{Z}^h, \quad 1 \leq c_1 < \dots < c_h \leq k;$$

by definition, $\begin{bmatrix} c_1 \\ \vdots \\ c_h \end{bmatrix} \leq \begin{bmatrix} c'_1 \\ \vdots \\ c'_h \end{bmatrix}$ if and only if $\begin{cases} c_1 \leq c'_1 \\ \vdots \\ c_h \leq c'_h \end{cases}$.

Let us make a few simple but useful observations.

LEMMA 4.11.

- (i) The cardinality of $\mathbf{P}_{h,k}$ is $\binom{k}{h}$.
- (ii) The columns $\hat{0} = \begin{bmatrix} 1 \\ \vdots \\ h \end{bmatrix}$ and $\hat{1} = \begin{bmatrix} k-h+1 \\ \vdots \\ k \end{bmatrix}$ are the unique minimal and maximal elements of $\mathbf{P}_{h,k}$, respectively.
- (iii) The poset $\mathbf{P}_{h,k}$ is graded, with the rank function given by

$$\text{rk}(\mathbf{c}) = c_1 + \dots + c_h - \frac{h(h+1)}{2}.$$

- (iv) Each maximal chain in $\mathbf{P}_{h,k}$ has cardinality $h(k-h) + 1$.

REMARK 4.12. The poset $\mathbf{P}_{h,k}$ is canonically isomorphic to the poset of integer partitions (partially ordered component-wise) having at most h parts all of which are $\leq k-h$. The isomorphism is given by

$$\mathbf{c} \mapsto (c_h - h, \dots, c_1 - 1).$$

Put another way, $\mathbf{P}_{h,k}$ is canonically isomorphic to the poset of Young diagrams fitting inside the $h \times (k - h)$ rectangle, ordered by inclusion. Such a Young diagram $\lambda = (\lambda_1 \geq \dots \geq \lambda_h)$ corresponds to the column \mathbf{c} as in (4.10) described pictorially as follows. We assume the “English” convention for drawing Young diagrams, with the longest row at the top. Starting at the lower-left corner of the $h \times (k - h)$ box, trace the lower-right boundary of λ , making the total of k unit steps. Among them, there are exactly h vertical steps. The location of the i th vertical step, counting from the bottom, among the k unit steps, is given by the i th entry $c_i = \lambda_{h-i+1} + i$.

Under this isomorphism, the maximal chains

$$Q = \{\mathbf{c}_0 \leq \mathbf{c}_1 \leq \dots \leq \mathbf{c}_{h(k-h)}\} \in \text{MaxChains}(\mathbf{P}_{h,k})$$

are interpreted as the standard Young tableaux of rectangular shape $h \times (k - h)$. (The reader unfamiliar with the tableau terminology is referred to Definition 5.1.) In concrete terms, the column \mathbf{c}_j describes (the lower-right boundary of) the diagram formed by the entries $1, \dots, j$ of the standard tableau Q .

EXAMPLE 4.13. Let $h = 2$ and $k = 5$. The poset $\mathbf{P}_{2,5}$ consists of $\binom{5}{2} = 10$ elements of the form $\begin{bmatrix} a \\ b \end{bmatrix}$, with $1 \leq a < b \leq 5$. These are in bijection with partitions $\mu = (\mu_1, \mu_2) = (b - 2, a - 1)$ satisfying $3 \geq \mu_1 \geq \mu_2 \geq 0$ (equivalently, Young diagrams fitting inside the 2×3 rectangle). There are 5 maximal chains in $\mathbf{P}_{2,5}$, corresponding to the 5 standard Young tableaux of this rectangular shape. See Figure 4.1. \diamond

We will later need the following crude estimate.

LEMMA 4.14. *The number of maximal chains in $\mathbf{P}_{h,k}$ does not exceed $h^{h(k-h)}$.*

PROOF. At each of the $h \times (k - h)$ steps in a maximal chain, we add 1 to one of the h components of a column. \square

$Q \in \text{MaxChains}(\mathbf{P}_{2,5})$	standard tableau	Q^*	descents
$[\frac{1}{2}] < [\frac{1}{3}] < [\frac{2}{3}] < [\frac{2}{4}] < [\frac{3}{4}] < [\frac{3}{5}] < [\frac{4}{5}]$	$\begin{bmatrix} 135 \\ 246 \end{bmatrix}$	\emptyset	
$[\frac{1}{2}] < [\frac{1}{3}] < [\frac{2}{3}] < [\frac{2}{4}] < [\frac{2}{5}] < [\frac{3}{5}] < [\frac{4}{5}]$	$\begin{bmatrix} 134 \\ 256 \end{bmatrix}$	$[\frac{2}{5}]$	4
$[\frac{1}{2}] < [\frac{1}{3}] < [\frac{1}{4}] < [\frac{2}{4}] < [\frac{3}{4}] < [\frac{3}{5}] < [\frac{4}{5}]$	$\begin{bmatrix} 125 \\ 346 \end{bmatrix}$	$[\frac{1}{4}]$	2
$[\frac{1}{2}] < [\frac{1}{3}] < [\frac{1}{4}] < [\frac{2}{4}] < [\frac{2}{5}] < [\frac{3}{5}] < [\frac{4}{5}]$	$\begin{bmatrix} 124 \\ 356 \end{bmatrix}$	$[\frac{1}{4}] < [\frac{2}{5}]$	2, 4
$[\frac{1}{2}] < [\frac{1}{3}] < [\frac{1}{4}] < [\frac{1}{5}] < [\frac{2}{5}] < [\frac{3}{5}] < [\frac{4}{5}]$	$\begin{bmatrix} 123 \\ 456 \end{bmatrix}$	$[\frac{1}{5}]$	3

Figure 4.1: Maximal chains in the poset $\mathbf{P}_{2,5}$.

DEFINITION 4.15 (*Intervals $\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}]$, and lexicographic ordering of maximal chains*). For $\mathbf{a}, \mathbf{b} \in \mathbf{P}_{h,k}$ satisfying $\mathbf{a} \leq \mathbf{b}$, we denote by $[\mathbf{a}, \mathbf{b}] = \mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}]$ the corresponding (order-theoretic) interval:

$$\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}] = \{\mathbf{c} \in \mathbf{P}_{h,k} \mid \mathbf{a} \leq \mathbf{c} \leq \mathbf{b}\}.$$

In the special case $\mathbf{P}_{h,k}[\hat{0}, \hat{1}] = \mathbf{P}_{h,k}$, we recover the entire poset $\mathbf{P}_{h,k}$.

The lexicographic ordering on $\text{MaxChains}(\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}])$ (denoted by the symbol \prec) is the linear order defined as follows. Let

$$Q = \{\mathbf{a} = \begin{bmatrix} a_{11} \\ \vdots \\ a_{h1} \end{bmatrix} < \cdots < \begin{bmatrix} a_{1N} \\ \vdots \\ a_{hN} \end{bmatrix} = \mathbf{b}\},$$

$$Q' = \{\mathbf{a} = \begin{bmatrix} a'_{11} \\ \vdots \\ a'_{h1} \end{bmatrix} < \cdots < \begin{bmatrix} a'_{1N} \\ \vdots \\ a'_{hN} \end{bmatrix} = \mathbf{b}\}$$

be two maximal chains in $\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}]$. Let j indicate the leftmost position where these two chains differ, i.e., the smallest index for which there exists i with $a_{ij} \neq a'_{ij}$. Furthermore, let i be the largest index (i.e., the lowermost location) for which this inequality occurs (for the minimal choice of j). Then $Q' \prec Q$ if and only if $a'_{ij} < a_{ij}$.

EXAMPLE 4.16. In Example 4.13, the maximal chains in $\mathbf{P}_{2,5} = \mathbf{P}_{2,5}[\hat{0}, \hat{1}]$ are listed in the lexicographic order, top down. \diamond

REMARK 4.17. Under the canonical isomorphism described in Remark 4.12, the maximal chains in $\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}]$ correspond to the standard Young tableaux of a fixed skew shape λ/μ , with λ and μ corresponding to \mathbf{b} and \mathbf{a} , respectively. The lexicographic ordering on $\text{MaxChains}(\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}])$ translates into the linear order on the standard tableaux of shape λ/μ defined as follows. Let Q and Q' be two such tableaux, and let i be the smallest entry whose locations in Q and Q' differ from each other. Specifically, let b and b' be the boxes containing i in Q and Q' , respectively. Note that b and b' are located in different rows and different columns. Then

$$(4.18) \quad Q' \prec Q \stackrel{\text{def}}{\iff} b' \text{ is located to the left of } b.$$

In the case $\mathbf{P} = \mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}]$ under our consideration, the definition (4.2) of the chain Q^* translates into the language of tableaux as follows: the elements of Q^* are in bijection with the descents of Q , i.e., those entries j for which $j+1$ appears in Q strictly to the left of j —so that switching j and $j+1$ yields a lexicographically smaller tableau. More precisely, each descent j contributes a column $\mathbf{c} \in Q^*$ corresponding to the Young diagram formed by the entries $1, \dots, j$ of Q . See Figure 4.1.

The following statement is (a reformulation of) a special case of the well known result on shellability of intervals in Bruhat order quotients for finite Coxeter groups, see Björner & Wachs (1982, 1988); Proctor (1982). We provide a proof in order to keep the exposition self-contained.

LEMMA 4.19. The lexicographic ordering on $\text{MaxChains}(\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}])$ is proper.

PROOF. In the language of Young diagrams fitting inside the $h \times (k - h)$ rectangle (cf. Remark 4.17), the claim (4.3) translates into the following statement: if Q and Q' are standard tableaux of skew shape λ/μ such that $Q' \prec Q$, then there is a descent j in Q such that the entries $1, \dots, j$ in Q form a shape different from the one formed by those entries in Q' . To prove this, consider the smallest entry i whose locations in Q and Q' differ from each other. Let b and b' denote the corresponding boxes, as in (4.18).

By construction, the tableau Q must have a descent $j \geq i$; let us consider the smallest such descent. None of the entries $1, \dots, j$ in Q is located in box b' . On the other hand, in the tableau Q' , the box b' contains $i \leq j$, and the claim follows. \square

Combining [Lemma 4.7](#) and [Lemma 4.19](#) enables us to express a generating function for multichains in $\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}]$ in terms of complete homogeneous symmetric functions. These expressions, reformulated in terms of semistandard tableaux, will be used in [Section 5](#) to obtain efficient $\{+, \times\}$ -algorithms for computing Schur functions.

5. Schur polynomials as multichain generating functions

Let us recall the combinatorial definition of a Schur polynomial $s_\lambda(x_1, \dots, x_k)$ labeled by an integer partition $\lambda = (\lambda_1 \geq \dots \geq \lambda_\ell \geq 0)$. Note that we allow trailing zeroes at the end of λ .

We assume that $\ell \leq k$. This condition does not restrict the generality, since $\lambda_\ell > 0$ and $\ell > k$ imply $s_\lambda(x_1, \dots, x_k) = 0$.

We use the notation $n = |\lambda| = \lambda_1 + \dots + \lambda_\ell$ for the size of the partition λ .

DEFINITION 5.1 (*Tableaux, Schur functions*). A semistandard Young tableau T of shape $\lambda = |\lambda|$ is an array of integers

$$T = (t_{i,j} \mid 1 \leq i \leq \ell, 1 \leq j \leq \lambda_i)$$

satisfying $t_{i,j} < t_{i+1,j}$ and $t_{i,j} \leq t_{i,j+1}$ whenever these inequalities make sense. A tableau T is called *standard* if each of the numbers $1, \dots, n$ appears exactly once among the n tableau entries $t_{i,j}$. We denote by \mathbf{x}^T the monomial associated with T :

$$\mathbf{x}^T = \prod_{i,j} x_{t_{i,j}}.$$

The Schur function (or Schur polynomial) $s_\lambda(x_1, \dots, x_k)$ is the generating function for semistandard tableaux of shape λ and entries in $\{1, \dots, k\}$:

$$s_\lambda(x_1, \dots, x_k) = \sum_{|T|=\lambda} \mathbf{x}^T.$$

By construction, $s_\lambda(x_1, \dots, x_k)$ is a homogeneous polynomial of degree n in the variables x_1, \dots, x_k , with positive integer coefficients. It is well known (Stanley 1999, Chapter 7) that $s_\lambda(x_1, \dots, x_k)$ is symmetric with respect to permutations of the variables.

EXAMPLE 5.2. Let $\ell = 2$ and $\lambda = (r, r)$. A semistandard tableau of shape λ is a $2 \times r$ matrix $T = (t_{i,j})$ with positive integer entries which weakly increase left-to-right in each row, and strictly increase top-down in each column. The corresponding Schur polynomial is given by $s_{(r,r)}(x_1, \dots, x_k) = \sum_T \prod_i \prod_j x_{t_{i,j}}$ where the sum is over all such tableaux with entries $\leq k$. For example, if $r = 2$ and $k = 3$, then we get 6 different tableaux, and the answer is $s_{(2,2)}(x_1, x_2, x_3) = x_1^2 x_2^2 + x_1^2 x_3^2 + x_2^2 x_3^2 + x_1^2 x_2 x_3 + x_1 x_2^2 x_3 + x_1 x_2 x_3^2$. \diamond

Our next goal is to restate Definition 5.1 using the language of multichain generating functions introduced in Section 4.

The connection between Schur functions and the posets $\mathbf{P}_{h,k}$ comes from the straightforward observation that the multichains of size m in $\mathbf{P}_{h,k}$ are in a canonical bijection with the semistandard tableaux of rectangular shape $h \times m$ and entries $\leq k$. (This bijection should not be confused with the construction described in Remark 4.12 above, which is of a rather different nature.) We next extend this correspondence to arbitrary shapes. This will require some preparation.

DEFINITION 5.3 (*Dissecting Young diagrams into rectangular shapes*). Let $\lambda = (\lambda_1 \geq \dots \geq \lambda_\ell)$ be an integer partition. As usual, we denote by λ' the conjugate partition, i.e., the partition whose parts are the column lengths of (the shape of) λ . We then denote by $\tilde{\lambda}'_1 > \dots > \tilde{\lambda}'_s$ the integers, listed in the decreasing order, which appear as parts of λ' . In other words, $\tilde{\lambda}'_1, \dots, \tilde{\lambda}'_s$ are all the different heights of columns in the Young diagram of λ . We denote by $\tilde{\lambda} = (\tilde{\lambda}_1 \geq \dots \geq \tilde{\lambda}_\ell)$ the partition conjugate to $\tilde{\lambda}' = (\tilde{\lambda}'_1, \dots, \tilde{\lambda}'_s)$. To rephrase, the shape $\tilde{\lambda}$ is obtained from λ by keeping one column of each height, and striking out the rest.

We can now dissect the Young diagram λ by vertical cuts into s rectangular shapes of sizes $h \times (\lambda_h - \lambda_{h+1})$ where h runs over the

set of parts of $\tilde{\lambda}'$ (equivalently, the distinct column lengths of λ). To simplify notation for the sake of future arguments, we denote $h_j = \tilde{\lambda}'_j$ and $m_j = \lambda_{h_j} - \lambda_{h_{j+1}} - 1$, so that λ gets dissected into rectangles of sizes $h_j \times (m_j + 1)$, for $j = 1, \dots, s$.

EXAMPLE 5.4. Let $\lambda = (6, 6, 4, 1, 1)$, $\ell = 5$. Then

$$\lambda' = (5, 3, 3, 3, 2, 2), \tilde{\lambda}' = (5, 3, 2), \tilde{\lambda} = (3, 3, 2, 1, 1), s = 3.$$

The shape λ can be dissected by vertical cuts into three rectangles of sizes 5×1 , 3×3 , and 2×2 , respectively. In this example, we have $h_1 = 5, h_2 = 3, h_3 = 2, m_1 = 0, m_2 = 2, m_3 = 1$. \diamond

DEFINITION 5.5 (*Pruning of tableaux*). Let T be a semistandard tableau of shape λ . The pruning of T is the semistandard tableau \tilde{T} of shape $\tilde{\lambda}$ obtained from T by selecting the rightmost column of each height (and removing all columns of that height located to the left of it). We denote by $\mathbf{a}_1, \dots, \mathbf{a}_s$ the columns of \tilde{T} , listed left to right. (These columns have heights h_1, \dots, h_s , respectively.) We denote by $\bar{\mathbf{a}}_j$ the column of height h_{j+1} obtained from \mathbf{a}_j by removing the $h_j - h_{j+1}$ bottom entries.

We furthermore denote by T_1, \dots, T_s the semistandard tableaux of rectangular shapes $h_1 \times m_1, \dots, h_s \times m_s$ obtained by dissecting T by the vertical cuts described in Definition 5.3, and then removing the rightmost column from each of the resulting tableaux. (If $m_j = 0$, then T_j is empty.) Thus T is obtained by interlacing the rectangular tableaux T_j with the columns of the pruning: $T = [T_1 | \mathbf{a}_1 | T_2 | \mathbf{a}_2 | \dots | T_s | \mathbf{a}_s]$.

EXAMPLE 5.6. Continuing with Example 5.4, let $T = \begin{bmatrix} 1 & 1 & 2 & 2 & 2 & 4 \\ 2 & 2 & 3 & 3 & 3 & 5 \\ 4 & 5 & 6 & 6 & & \\ 5 & & & & & \\ 6 & & & & & \end{bmatrix}$.

Then $\tilde{T} = \begin{bmatrix} 1 & 2 & 4 \\ 2 & 3 & 5 \\ 4 & 6 & \\ 5 & & \\ 6 & & \end{bmatrix}$, $T_1 = \emptyset$, $\mathbf{a}_1 = \begin{bmatrix} 1 \\ 2 \\ 4 \\ 5 \\ 6 \end{bmatrix}$, $T_2 = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 5 & 6 \end{bmatrix}$, $\mathbf{a}_2 = \begin{bmatrix} 2 \\ 3 \\ 6 \end{bmatrix}$, $T_3 = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$, $\mathbf{a}_3 = \begin{bmatrix} 4 \\ 5 \end{bmatrix}$. \diamond

Consider the set of semistandard tableaux T of a given shape λ , with entries $\leq k$, and with a given pruning $\tilde{T} = [\mathbf{a}_1 | \cdots | \mathbf{a}_s]$. Note that once \tilde{T} and λ have been fixed, each tableau T_j , for $1 \leq j \leq s$, can be chosen independently of the others, as long as it satisfies the following restrictions:

- T_j is a semistandard tableau of rectangular shape $h_j \times m_j$, with entries $\leq k$; as such, it can be viewed as a multichain of size m_j in the poset $\mathbf{P}_{h_j, k}$;
- every column \mathbf{a} in T_j (i.e., every element of this multichain) satisfies the inequalities $\bar{\mathbf{a}}_{j-1} \leq \mathbf{a} \leq \mathbf{a}_j$, with respect to the partial order in $\mathbf{P}_{h_j, k}$.

(We set $\bar{\mathbf{a}}_0 = \hat{0} = \begin{bmatrix} 1 \\ \vdots \\ \ell \end{bmatrix}$ by convention, so that the lower bound is redundant for $j = 1$.) This gives a bijection between the set of tableaux under consideration and the Cartesian product of sets of multichains in the posets $\mathbf{P}_{h_j, k}$:

$$\left\{ \begin{array}{l} \text{semistandard tableaux } T \\ \text{of shape } \lambda, \text{ with entries } \leq k, \\ \text{with pruning } \tilde{T} = [\mathbf{a}_1 | \cdots | \mathbf{a}_s] \end{array} \right\} \longleftrightarrow \prod_{j=1}^s \left\{ \begin{array}{l} \text{multichains} \\ \text{of size } m_j \\ \text{in } \mathbf{P}_{h_j, k}[\bar{\mathbf{a}}_{j-1}, \mathbf{a}_j] \end{array} \right\}$$

Identifying multichains in $\mathbf{P}_{h_j, k}[\bar{\mathbf{a}}_{j-1}, \mathbf{a}_j]$ with semistandard tableaux of rectangular shape, and passing to generating functions, we obtain the following result.

LEMMA 5.7. *With the notation as above, we have*

$$(5.8) \quad s_\lambda(x_1, \dots, x_k) = \sum_{\tilde{T}} \mathbf{x}^{\tilde{T}} \prod_{j=1}^s \sum_{T_j} \mathbf{x}^{T_j},$$

where

- $\tilde{T} = [\mathbf{a}_1 | \cdots | \mathbf{a}_s]$ runs over semistandard tableaux of shape $\tilde{\lambda}$, with entries $\leq k$;
- each T_j runs over semistandard tableaux of rectangular shape $h_j \times m_j$ whose columns form a multichain in $\mathbf{P}_{h_j, k}[\bar{\mathbf{a}}_{j-1}, \mathbf{a}_j]$.

In view of Lemma 4.7 and Lemma 4.19, the sums $\sum_{T_j} \mathbf{x}^{T_j}$ appearing in (5.8) can be computed using the formula (4.8):

LEMMA 5.9. Let $\mathbf{a}, \mathbf{b} \in \mathbf{P}_{h,k}$ be two columns such that $\mathbf{a} \leq \mathbf{b}$. Then

$$(5.10) \quad \sum_T \mathbf{x}^T = \sum_Q \mathbf{x}^{Q^*} h_{m-|Q^*|}(\mathbf{x}^{\mathbf{c}_1}, \dots, \mathbf{x}^{\mathbf{c}_N}),$$

where

- T runs over semistandard tableaux of rectangular shape $h \times m$ whose columns form a multichain in $\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}]$;
- $Q = [\mathbf{c}_1 | \dots | \mathbf{c}_N]$ runs over the maximal chains in $\mathbf{P}_{h,k}[\mathbf{a}, \mathbf{b}]$;
- Q^* is given by (4.2).

For the reader's convenience, we restate the definition of Q^* in concrete terms; cf. also Remark 4.17. For each pair of consecutive columns \mathbf{c}_j and \mathbf{c}_{j+1} , we have $\mathbf{c}_{j+1} = \mathbf{c}_j + \mathbf{e}_{i_j}$ for some $i_j \in \{1, \dots, h\}$, where \mathbf{e}_i denotes the column whose i th component is equal to 1, and all others are equal to 0. The chain/tableau Q^* is formed by the subset of columns \mathbf{c}_j for which $i_{j-1} > i_j$ and moreover $\mathbf{c}_{j-1} + \mathbf{e}_{i_j} \in \mathbf{P}_{h,k}$ (so that replacing \mathbf{c}_j by $\mathbf{c}_{j-1} + \mathbf{e}_{i_j}$ transforms Q into a lexicographically smaller maximal chain).

EXAMPLE 5.11. Let $h = 2$, $k = 5$, $\mathbf{a} = \hat{0}$, $\mathbf{b} = \hat{1}$, cf. Example 4.13. Then (5.10) becomes

$$\begin{aligned} s_{(m,m)}(x_1, \dots, x_5) &= h_m(x_1x_2, x_1x_3, x_2x_3, x_2x_4, x_3x_4, x_3x_5, x_4x_5) \\ &\quad + x_2x_5 h_{m-1}(x_1x_2, x_1x_3, x_2x_3, x_2x_4, x_2x_5, x_3x_5, x_4x_5) \\ &\quad + x_1x_4 h_{m-1}(x_1x_2, x_1x_3, x_1x_4, x_2x_4, x_3x_4, x_3x_5, x_4x_5) \\ &\quad + x_1x_4 \cdot x_2x_5 h_{m-2}(x_1x_2, x_1x_3, x_1x_4, x_2x_4, x_2x_5, x_3x_5, x_4x_5) \\ &\quad + x_1x_5 h_{m-1}(x_1x_2, x_1x_3, x_1x_4, x_1x_5, x_2x_5, x_3x_5, x_4x_5). \quad \diamond \end{aligned}$$

6. Proof of the main theorem

Combining (5.8) and (5.10), we obtain:

COROLLARY 6.1. *The Schur polynomial $s_\lambda(x_1, \dots, x_k)$ is given by*

$$(6.2) \quad s_\lambda(x_1, \dots, x_k) = \sum_{|\tilde{T}|=\tilde{\lambda}} \mathbf{x}^{\tilde{T}} \prod_{j=1}^s \sum_Q \mathbf{x}^{Q^*} h_{m_j-|Q^*|}(\mathbf{x}^{\mathbf{c}^1}, \dots, \mathbf{x}^{\mathbf{c}^N}),$$

where

- $\tilde{\lambda}$, s , h_1, \dots, h_s , and m_1, \dots, m_s are described in [Definition 5.3](#);
- $\tilde{T} = [\mathbf{a}_1 | \dots | \mathbf{a}_s]$ runs over semistandard tableaux of shape $\tilde{\lambda}$, with entries $\leq k$;
- $Q = [\mathbf{c}_1 | \dots | \mathbf{c}_N]$ runs over the maximal chains in $\mathbf{P}_{h_j, k}[\bar{\mathbf{a}}_{j-1}, \mathbf{a}_j]$.

To prove [Theorem 1.1](#), we analyze the (semiring) complexity of computing a Schur polynomial $s_\lambda(x_1, \dots, x_k)$ using the formula (6.2) together with [Theorem 1.4](#).

We begin by computing the monomials $\mathbf{x}^{\mathbf{c}}$, for all columns \mathbf{c} of height h_j with entries $\leq k$, for each $j \leq s$. This can be done using $\leq \ell \sum_{j \leq s} \binom{k}{h_j}$ multiplications. (Note that $s \leq \ell$.)

Recall that the Young diagram $\tilde{\lambda}$ has s columns, of heights h_1, \dots, h_s . Hence the number of tableaux \tilde{T} appearing in (6.2) is bounded by $\prod_{j \leq s} \binom{k}{h_j}$.

Each monomial $\mathbf{x}^{\tilde{T}}$ can be computed by $s - 1$ multiplications (given all the $\mathbf{x}^{\mathbf{c}_i}$).

The number of maximal chains in $\mathbf{P}_{h_j, k}[\bar{\mathbf{a}}_{j-1}, \mathbf{a}_j]$ is at most $h_j^{h_j(k-h_j)}$, by [Lemma 4.14](#). Each of these chains has length $N \leq h(k-h) + 1$. Since $|Q^*| \leq |Q| = N$, we can compute \mathbf{x}^{Q^*} in time $\leq h(k-h)$. Also, $m_j - |Q^*| \leq \lambda_1$. [Theorem 1.4](#) now implies that we can compute $\mathbf{x}^{Q^*} h_{m_j-|Q^*|}(\mathbf{x}^{\mathbf{c}^1}, \dots, \mathbf{x}^{\mathbf{c}^N})$ in time $O(h^2(k-h)^2 \log(\lambda_1))$. Putting everything together, we obtain the following upper bound on the semiring complexity of $s_\lambda(x_1, \dots, x_k)$:

$$\ell \sum_{j \leq s} \binom{k}{h_j} + \prod_{j \leq s} \binom{k}{h_j} \cdot (2s + \sum_{j \leq s} h_j^{h_j(k-h_j)} (O(h_j^2(k-h_j)^2 \log(\lambda_1)))).$$

This can be replaced by $O(\log(\lambda_1)) s \ell^2 k^2 2^{ks} \ell^d$ where

$$d = \max_j h_j(k-h_j) = \max_j \lambda'_j(k-\lambda'_j),$$

and then by $O(\log(\lambda_1)) k^5 2^{k\ell} \ell^d$. \square

Acknowledgements

Partially supported by the NSF grant DMS-1361789 (S. F.), the RSF grant 16-11-10075 (D. G.), and the NSERC (É. S.). D. G. thanks MCCME Moscow and Max-Planck Institut für Mathematik for their hospitality and inspiring atmosphere.

We thank the referee for a number of suggestions which led to the improvement of the presentation.

References

ANDERS BJÖRNER & MICHELLE WACHS (1982). Bruhat order of Coxeter groups and shellability. *Adv. in Math.* **43**(1), 87–100.

ANDERS BJÖRNER & MICHELLE L. WACHS (1988). Generalized quotients in Coxeter groups. *Trans. Amer. Math. Soc.* **308**(1), 1–37.

CY P. CHAN, VESSELIN DRENSKY, ALAN EDELMAN, RAYMOND KAN & PLAMEN KOEV (2008). On computing Schur functions and series thereof, preprint.

JAMES DEMMEL & PLAMEN KOEV (2006). Accurate and efficient evaluation of Schur and Jack functions. *Math. Comp.* **75**(253), 223–239.

SERGEY FOMIN, DIMA GRIGORIEV & GLEB KOSHEVOY (2016). Subtraction-free complexity, cluster transformations, and spanning trees. *Found. Comput. Math.* **16**(1), 1–31.

DIMA GRIGORIEV & GLEB KOSHEVOY (2016). Complexity of tropical Schur polynomials. *J. Symbolic Comput.* **74**, 46–54.

MARK JERRUM & MARC SNIR (1982). Some exact complexity results for straight-line computations over semirings. *J. Assoc. Comput. Mach.* **29**(3), 874–897.

PLAMEN KOEV (2007). Accurate computations with totally nonnegative matrices. *SIAM J. Matrix Anal. Appl.* **29**(3), 731–751.

IAN G. MACDONALD (2015). *Symmetric functions and Hall polynomials*. Oxford University Press, New York, 2nd edition.

HARIHARAN NARAYANAN (2006). On the complexity of computing Kostka numbers and Littlewood-Richardson coefficients. *J. Algebraic Combin.* **24**(3), 347–354.

ROBERT A. PROCTOR (1982). Classical Bruhat orders and lexicographic shellability. *J. Algebra* **77**(1), 104–126.

CLAUS-PETER SCHNORR (1976). A lower bound on the number of additions in monotone computations. *Theoret. Comput. Sci.* **2**(3), 305–315.

ELI SHAMIR & MARC SNIR (1977). Lower bounds on the number of multiplications and the number of additions in monotone computations. Technical Report RC-6757, IBM.

RICHARD P. STANLEY (1999). *Enumerative combinatorics. Vol. 2.* Cambridge University Press, Cambridge.

VOLKER STRASSEN (1972/73). Die Berechnungskomplexität von elementarsymmetrischen Funktionen und von Interpolationskoeffizienten. *Numer. Math.* **20**, 238–251.

VOLKER STRASSEN (1973). Vermeidung von Divisionen. *J. Reine Angew. Math.* **264**, 184–202.

LESLIE G. VALIANT (1980). Negation can be exponentially powerful. *Theoret. Comput. Sci.* **12**(3), 303–314.

MICHELLE L. WACHS (2007). Poset topology: tools and applications. In *Geometric combinatorics*, volume 13 of *IAS/Park City Math. Ser.*, 497–615. Amer. Math. Soc., Providence, RI.

Manuscript received 20 February 2017

SERGEY FOMIN
 Department of Mathematics
 University of Michigan
 Ann Arbor, MI 48109, USA
 fomin@umich.edu

DIMA GRIGORIEV
 CNRS, Mathématiques
 Université de Lille
 Villeneuve d’Ascq, 59655, France
 Dmitry.Grigoryev@math.
 univ-lille1.fr

DORIAN NOGNENG
LIX, École Polytechnique
91128 Palaiseau Cedex, France
dorian.nogneng@lix.
polytechnique.fr

ÉRIC SCHOST
Cheriton School of Computer Sci-
ence
University of Waterloo
Waterloo, ON, Canada N2L 3G1
eschost@uwaterloo.ca