ISSN 0278-6419, Moscow University Computational Mathematics and Cybernetics, 2019, Vol. 43, No. 1, pp. 1–11. © Allerton Press, Inc., 2019.
Russian Text © S.I. Veselov, D.V. Gribanov, D.S. Malyshev, 2019, published in Vestnik Moskovskogo Universiteta, Seriya 15: Vychislitel'naya Matematika i Kibernetika, 2019, No. 1, pp. 8–24.

FPT-Algorithm for Computing the Width of a Simplex Given by a Convex Hull

S. I. Veselov[1*], D. V. Gribanov[1], and D. S. Malyshev[2***]**

[1] *Institute of Information Technologies, Mathematics, and Mechanics, State University of Nizhny Novgorod, Nizhny Novgorod, 603950 Russia*

[2] *Faculty of Informatics, Mathematics and Computer Science, National Research University Higher School of Economics, Nizhny Novgorod, 603155 Russia*

Received July 24, 2018

Abstract—The problem of computing the width of simplices generated by the convex hull of their integer vertices is considered. An FPT algorithm, in which the parameter is the maximum absolute value of the rank minors of the matrix consisting from the simplex vertices, is presented.

Keywords: convex body width, integer programming, FPT algorithm, bounded minors.

DOI: 10.3103/S0278641919010084

1. INTRODUCTION

1.1. Definitions and Notation

An algorithm parameterized by a parameter k is referred to as an *FPT algorithm* (a fixed parameter tractable algorithm) if its complexity can be estimated by the function $f(k)n^{O(1)}$, where n is the length of the input, and $f(k)$ is some computable function depending only on the parameter k. A computational problem parameterized by a parameter k is referred to as an *FPT problem* (a fixed parameter tractable problem) if it can be solved using an FPT algorithm. The theory of parameterized complexity was described in detail in [1, 2].

Let $A \in \mathbb{Z}^{m\times n}$ and $b \in \mathbb{Z}^m$, $P(A,b) = \{x \in \mathbb{R}^n : Ax \leqslant b\}$ be a polyhedron defined by the system of inequalities $Ax \leqslant b$, $\Delta_k(A)$ is the maximum absolute value of the $k \times k$ minors of A, $\Delta(A) = \Delta_{\mathrm{rank}(A)}(A)$, $A_{i\,j}$ is an element in the i-row and the jth column of A ($A_{i\,*}$ and $A_{j\,*}$ are the ith row and jth column of A, respectively, $A^* = \det(A)A^{-1}$ is a matrix adjoint to A, and $||A||_{\max}$ is the element of the matrix A that is maximal in absolute value. It is obvious that $||A||_{\max} = \Delta_1(A)$. The set of integers starting with i and ending with j is denoted by $i : j = \{i, i+1, \ldots, j\}$.

We define the length of the bit record of an integer x, a rational irreducible fraction $r = \frac{p}{q}$, a rational vector $v \in \mathbb{Q}^n$, and a rational matrix $A \in \mathbb{Q}^{d\times n}$, according to [3, 4]:

$$\mathrm{size}(x) = 1 + \lceil \log_2(|x| + 1)\rceil,$$

$$\mathrm{size}(r) = 1 + \mathrm{size}(p) + \mathrm{size}(q),$$

$$\mathrm{size}(v) = n + \sum_{i=1}^{n} \mathrm{size}(v_i),$$

$$\mathrm{size}(A) = mn + \sum_{i=1}^{d}\sum_{j=1}^{n} \mathrm{size}(A_{i\,j}).$$

*E-mail: sergey.veselov@itmm.unn.ru.
**E-mail: dimitry.gribanov@gmail.com.
***E-mail: dsmalyshev@rambler.ru.

For a matrix $A \in \mathbb{R}^{m\times n}$, let cone.hull$(A) = \{At : t \in \mathbb{R}^n_+\}$ be the cone generated by columns of A, conv.hull$(A) = \{At : t \in \mathbb{R}^n_+, \sum_{i=1}^{n} t_i = 1\}$ be the convex hull generated by columns of A.

A matrix $A \in \mathbb{Z}^{n\times n}$ is referred to as *unimodular* if $|\det(A)| = 1$. The theory of unimodular matrices was described in [4, 5].

Let P be a convex body. The following quantity is referred to as *the width of P along the direction $c \in \mathbb{Z}^n$*

$$\mathrm{width}_c(P) = \max_{x\in P} c^\top x - \min_{x\in P} c^\top x.$$

The following quantity is referred to as *the width of the convex body P*

$$\mathrm{width}(P) = \min\{\mathrm{width}_c(P) : c \in \mathbb{Z}^n \setminus \{0\}\}.$$

The direction c, on which the minimum width is reached is referred to as the *flat direction of P*.

The algorithms for searching the width and flat direction of a convex body are important components of modern algorithms for integer linear and nonlinear programming, the complexity of which is polynomial for a fixed dimension. The main ideas of such algorithms were described in [6].

1.2. Review of Existing Results and the Aim of this Work

Let $A \in \mathbb{Z}^{m\times n}$, $b \in \mathbb{Z}^m$, and $c \in \mathbb{Z}^n$. Gomory [7–9] (see also [10, 11]) show that if the matrix A is square and nondegenerate, then an FPT algorithm with the complexity $\Delta n \cdot \mathrm{poly}(s)$, where $\Delta = \Delta(A)$ and $s = \mathrm{size}(A) + \mathrm{size}(b) + \mathrm{size}(c)$, can be used to solve the integer linear programming problem (ILPP) $\max\{c^\top x : x \in P(A, b) \cap \mathbb{Z}^n\}$. This result can, e.g., be obtained by a simple algorithm for confirming the presence of an integer point in the simplex specified by the system $Ax \leqslant b$. To do this, it suffices to solve the ILPP with an objective functional that coincides with the normal vector of one facet of the simplex.

The result obtained by Gomory was generalized in [12] for rectangular matrices, with the additional condition that the matrix A must not have degenerate rank order submatrices. The obtained algorithm has the complexity $n^{O(\log^3 \Delta)} \cdot \mathrm{poly}(s)$, which is not the complexity of an FPT-algorithm with respect to the parameter Δ.

The following improvements were made in [13]: The existence of an FPT algorithm was shown, when the matrix A was allowed to have any fixed number of rows. The complexity of this algorithm can be roughly estimated as $n^{2(k+1)}\Delta^{2(k+1)} \cdot \mathrm{poly}(s)$, where $k = m - n$ is the number of additional lines, provided that $\mathrm{rank}(A) = n$. The FPT algorithm for the case, when A is not allowed to have rank order degenerate submatrices was also obtained. Its complexity was no greater than $n^4\Delta^4 \cdot \mathrm{poly}(s)$.

Cases where there are no constraints on the matrix A remain poorly studied. It is well known (see, e.g., [4, 5]) that for $\Delta(A) = 1$ the ILPP $\max\{c^\top x : x \in P(A, b) \cap \mathbb{Z}^n\}$ is equivalent to a linear programming problem, which in turn can be solved in polynomial time [14–17]. The case of $\Delta(A) = 2$ was first studied in [18]. It was shown that the solidity of such a polyhedron implies the existence of an integer point in the polyhedron, meaning there is a polynomial time algorithm for checking the nonemptiness of the intersection of the polyhedron and the integer lattice. It was also shown that in a bimodular case, the vertices of the convex hull of integer points of the original polyhedron lie on its edges. The polynomial solvability of the bimodular ILPP, which almost closes the case of $\Delta(A) = 2$, was recently shown in [19] using these results. It should be noted that the algorithm obtained in [19] was quite complex, so the problem of obtaining a simpler and more efficient algorithm is relevant.

Almost nothing is currently known about the case of $\Delta(A) = 3$. Some results associated with $\{0, 1\}$–*ILPP*, when the variables of the problem and the elements of the matrices and vectors A, b, c consist of zeros and ones, are noteworthy. It was shown in [20] that if $\Delta\binom{c^\top}{A}$ is bounded, and each row of A contains no more than two units, then the ILPP can be solved in polynomial time. The polynomial solvability of $\{0, 1\}$–*ILPP* formulations of the edge and vertex dominating set problems was established in [21–23], provided that $\Delta\binom{c^\top}{A}$ is bounded. We should note recent works on determining the boundary between polynomial solvability and NP completeness for some extremal problems on graphs [24–29].

One possible line for developing this work is to search for a boundary cases when the polynomial solvability (FPT solvability) changes to NP completeness (W[1] completeness).

The problem of computing the width and flat directions for the polyhedra $P(A, b)$ and convex bodies of more general form is the most important component of integer programming algorithms [30–34] that are polynomial for a fixed dimension. In [35], it was shown that the width computation problem is NP hard even when the system $Ax \leqslant b$ contains only the $n+1$ line and defines a simplex. However, it was shown in [36] that if we bound the value $\Delta = \Delta(A)$ of minors of A and the value $\Delta(A\ b)$ of minors of extended the matrix $(A\ b)$, then we can obtain an algorithm with a polynomial time complexity for the considered problem. The result of [36] was improved in [13], where the algorithm with complexity $O(\log \Delta \cdot n^5 \Delta^3 \cdot \Delta(A\ b) \cdot \mathrm{mult}(n^3 \log \Delta(A\ b) + n^3 \log n))$, where $\mathrm{mult}(t)$ is the bit complexity of multiplying two numbers of the length t. Under the additional condition that the simplex contains no integer points, we can avoid an exponential dependence on $\Delta(A\, b)$ and construct an FPT algorithm with the complexity $O(\log \Delta \cdot n^4 \Delta^4 \cdot \mathrm{mult}(n^3 \log \Delta(A\, b) + n^3 \log n))$.

The aim of this work is to consider a dual case, in which the simplex is given by the convex hull of its vertices $S = \mathrm{conv.hull}(V)$, where V is a matrix composed of the simplex vertices. We show that there is an FPT algorithm parameterized by $\Delta(V)$. We assume that V and the vertices of the simplex are integral. The case of the rational matrix V can be reduced to the integer case by multiplying by the corresponding factors, but $\Delta(V)$ will grow exponentially.

1.3. Structure and Results of This Work

The first part of this work serves as an introduction in which the necessary definitions and notation are given. It presents a brief overview of the results on the solvability of the ILPP and problems close to it, provided that the absolute value of the minors of the matrices included in the problem formulation is bounded. The second part contains intermediate propositions necessary for deriving the main results of this work. The third part presents the main result of this work, i.e, the FPT algorithm for computing the width of a simplex generated by a convex hull of points.

2. AUXILIARY RESULTS

2.1. Hermite and Smith Normal Forms

Proposition 1. *Let $r_1, r_2, \ldots, r_n \in \mathbb{Z}_+$ and r_1 be the number maximum in absolute value among $\{r_i\}$, then*

$$\mathrm{size}(r_1 + r_2 + \cdots + r_n) \leqslant \log_2 n + \mathrm{size}(r_1),$$

$$\mathrm{size}(r_1 r_2 \ldots r_n) \leqslant \mathrm{size}(r_1) + \mathrm{size}(r_2) + \cdots + \mathrm{size}(r_n).$$

Proposition 1 is a direct consequence of the value size definition given in the introduction.

Proposition 2. *Let $x, y \in \mathbb{Q}^n$ and $x^\top y \in \mathbb{Z}$, then*

$$\mathrm{size}(x^\top y) = O(\log n + \log ||x||_\infty + \log ||y||_\infty).$$

Let $A \in \mathbb{Q}^{m\times n}$, $B \in \mathbb{Q}^{n \times k}$, $C \in \mathbb{Z}^{m\times k}$ and $C = AB$. Then

$$\mathrm{size}(||C||_{\max}) = O(\log n + \log ||A||_{\max} + \log ||B||_{\max}).$$

Proof. It is obvious that $|x^\top y| \leqslant n||x||_\infty ||y||_\infty$. We obtain the required equality by finding the logarithm. The second part of the proposition follows directly from the formula $C_{i\,j} = (A_{i\,*})^\top B_{*\,j}$.

Proposition 3. *Let $A \in \mathbb{Q}^{n\times n}$, then*

$$\mathrm{size}(\det A) \leqslant 2\mathrm{size}(A).$$

Proof of Proposition 3 can be found in [3, 4]. The following obvious equality follows from Proposition 3 and the inverse matrix definition

$$\mathrm{size}(||A^{-1}||_{\max}) = O(\mathrm{size}(A)).$$

The following proposition was proved in [13]:

Proposition 4. *Let the matrix* $A \in \mathbb{Z}^{(n+1)\times n}$ *be reduced to the Hermite normal form (HNF). Then*

$$\Delta_{n-1}(A) \leqslant \frac{\Delta^2}{2}(1+\log_2 \Delta),$$

where $\Delta = \Delta(A)$.

The most important tools for studying lattices and integer solutions of systems of linear equations and inequalities are the Hermite and Smith normal forms [3, 4, 6, 37].

Theorem 1. *Any matrix* $A \in \mathbb{Q}^{m\times n}$ *of a rank* r *can be presented as a product* $A = HQ$, *where matrix* $Q \in \mathbb{Z}^{n\times n}$ *is unimodular, and the matrix* $H \in \mathbb{Q}^{m\times n}$, *referred to as the Hermite normal form (HNF), has the form*

$$\begin{pmatrix} H_{1\,1} & 0 & \dots & 0 & 0 & \dots & 0 \\ H_{2\,1} & H_{2\,2} & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ H_{r\,1} & H_{r\,2} & \dots & H_{r\,r} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ H_{m\,1} & H_{m\,2} & \dots & H_{m\,r} & 0 & \dots & 0 \end{pmatrix}.$$

For elements H *with numbers* $i, j \in \{1, 2, \dots, r\}$ *the following is true:* $H_{i\,i} > 0$, $0 \leqslant H_{i\,j} < H_{i\,i}$ *for* $i < j$. *For other elements of the matrix* H *these properties can be incorrect. If the source matrix* A *is integer, the matrix* H *is also an integer.*

Theorem 2. *Any matrix* $A \in \mathbb{Q}^{m\times n}$ *of a rank* r *can be presented as a product* $A = PSQ$, *where the matrices* $P \in \mathbb{Z}^{m\times m}$ *and* $Q \in \mathbb{Z}^{n\times n}$ *are unimodular, and the matrix* $S \in \mathbb{Q}^{m\times n}$, *referred to as the Smith normal form (SNF), has the form*

$$\begin{pmatrix} S_{1\,1} & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & S_{2\,2} & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & S_{r\,r} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{pmatrix}.$$

At the same time, $S_{i\,i} | S_{i+1\,i+1}$ *for* $1 \leqslant i \leqslant n-1$. *If the source matrix* A *is an integer, then the matrix* S *is an integer.*

The most time-efficient algorithms for computing HNF and SNF were given in [38, 39].

Theorem 3. *There are algorithms with the following complexity for computing the HNF and NFS of the matrix* A

$$\tilde{O}(n^{\Theta-1} m \cdot \mathrm{mult}(n \log ||A||_{\max})) = \tilde{O}(n^{\Theta-1} \cdot \mathrm{mult}(\mathrm{size} A)),$$

where Θ *is the matrix multiplication exponent.*

Proposition 5. *Let* $A \in \mathbb{Z}^{n\times n}$, $\Delta = |\det A| > 0$ *and* $A = HQ$, *where* Q *is an unimodular* $n \times n$ *matrix, and* H *is the HNF of* A. *The following equalities then hold:*

$$\mathrm{size}(H) = O(n^2 + n \log \Delta),$$

$$||H^{-1}||_{\max} \leqslant \Delta,$$

$$\mathrm{size}(||Q||_{\max}) = O(\log n + \log \Delta + \log ||A||_{\max}),$$

$$\mathrm{size}(||Q^{-1}||_{\max}) = O(\log n + \log \Delta + \mathrm{size}(A)).$$

Proof. Let us prove the first equality. The matrix H can be reduced to the following form through additional permutation of the rows and columns:

$$H_B = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 & 0 \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & 0 & 0 \dots & 0 \\ H_{s+1\,1} & H_{s+2\,2} & \dots & H_{s+1\,s} & H_{s+1\,s+1} & 0 \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ H_{n\,1} & H_{n\,2} & \dots & \dots & \dots & \dots & H_{n\,n} \end{pmatrix}, \quad (1)$$

where $H_{i\,i} > 1$ and $s+1 \le i \le n$. Let $k = n - s$ denote the number of diagonal elements that do not equal 1. It can be seen that $k \leqslant \log_2 \Delta$. The total size of the elements in the lines with numbers from 1 to s does not exceed $O(n^2)$. The total size of the elements in line i, where $s+1 \leqslant i \leqslant n$, does not exceed $O(n \log H_{i\,i})$. We obtain the required equality by summing all lines together.

Let us prove the second equality. It can be seen that $\Delta_{n-1}(H) \leqslant \Delta^2$. We find that $||H^{-1}||_{\max} \leqslant \Delta$, since the elements of matrix H^{-1} are fractions of the $n-1$ order minors to Δ.

The third equality follows directly from Proposition 2, since $Q = H^{-1}A$. The fourth equality follows from Proposition 2 and the equality after Proposition 3.

Remark 1. Let A be an integer nondegenerate $n \times n$ matrix. The unimodular matrix Q performing the decomposition $A = HQ$ is then unique and can be easily found using the formula $Q = H^{-1}A$. Unfortunately, this is not true for unimodular matrices P, Q from the decomposition $A = PSQ$, where S is the SNF of A, since P and Q are not uniquely defined. The problem of finding the smallest possible matrices P and Q thus arises. It was noted in [38] that an appropriate matrix P can be found using an algorithm with the same complexity as in Theorem 2. In addition, $\mathrm{size}(P) = O(\mathrm{size}(A))$ is true. This result suffices for us, since we did not construct the matrix Q in this work.

A similar problem arises when constructing the decomposition $A = HQ$ if the matrix $A \in \mathbb{Z}^{m \times n}$ is not square. The matrix Q is in this case not unique either. An algorithm for constructing Q with the same complexity as in Theorem 3 was provided in [39]. The matrix Q^{-1} produced by this algorithm has no more than $O(nm)$ nonzero elements and $\mathrm{size}(||Q||_{\max}) = O(m \log ||A||_{\max})$.

2.2. A Square Inequality System with Two-Sided Constraints

Let $A \in \mathbb{Z}^{n \times n}$, $\Delta = |\det(A)| > 0$ and $a, b, c \in \mathbb{Z}^n$. We then consider the problem

$$c^\top x \to \min \quad (2)$$

$$\begin{cases} a \leqslant Ax \leqslant b \\ x \in \mathbb{Z}^n. \end{cases} \quad (3)$$

The original system is converted to one below after introducing residuals $y = b - Ax$ and replacing $x = A^{-1}(b - y)$:

$$c^\top A^{-1} y - c^\top A^{-1} b \to \max$$

$$\begin{cases} Ax + y = b \\ 0 \leqslant y \leqslant b - a \\ x, y \in \mathbb{Z}^n. \end{cases}$$

Let $A = PSQ$, where S is the SNF of A, and P and Q are the unimodular matrices. The original system is converted into an equivalent system after multiplying the original system by P^{-1} and replacing $x \to Q^{-1}x$:

$$w^\top y \to \max \tag{4}$$

$$\begin{cases} Gy \equiv g \,(\text{mod S}) \\ 0 \leqslant y \leqslant \tau \\ y \in \mathbb{Z}^n, \end{cases} \tag{5}$$

where $w^\top = c^\top A^*$, $G = P^{-1} \bmod S$, $g = P^{-1}b \bmod S$ and $\tau = b - a$. Here, $A^* = \det(A)A^{-1}$ is the matrix adjoint to A. The latter problem is actually the Gomory group minimization problem [7–9] and can easily be solved via dynamic programming.

Remark 2. It can be seen that the matrix S has no more than $\log_2 \Delta$ diagonal elements unequal to one. Thus, we can assume that the matrix G has no more than $\log_2 \Delta$ rows, and the elements of its ith row do not exceed $S_{i\,i} \leqslant \Delta$. The same can be said about the vector g.

Let us consider the auxiliary problem $Prob(k, \eta)$ to solve the problem (3) using dynamic programming:

$$(w_{1:k})^\top y \to \max,$$

$$\begin{cases} G_{*\ 1:k}\, y \equiv \eta \,(\text{mod S}) \\ 0 \leqslant y \leqslant \tau_{1:k} \\ y \in \mathbb{Z}^k \end{cases} ,$$

where $1 \leqslant k \leqslant n$, $\eta \in \mathbb{Z}^n \bmod \text{S}$.

We denote the optimum value of the problem $Prob(k, \eta)$ by $\psi(k, \eta)$. For infeasible problems, we set $\psi(k, \eta) = -\infty$. If $w_k \leqslant 0$. The values of $\psi(k, \eta)$ then satisfy the recurrence relation

$$\psi(k, \eta) = \max_{0 \leqslant z \leqslant \min\{\tau_k, \Delta - 1\}} z w_k + \psi(k - 1, (\eta - zG_{*\ k}) \bmod S), \tag{6}$$

otherwise, if $w_k > 0$, we have

$$\psi(k, \eta) = \max_{\max\{0, \tau_k - \Delta + 1\} \leqslant z \leqslant \tau_k} z w_k + \psi(k - 1, (\eta - zG_{*\ k}) \bmod S). \tag{7}$$

The values of $\psi(1, \eta)$ for $w_1 \leqslant 0$ can be computed using the formula

$$\psi(1, \eta) = \max\{w_1 z : \text{for such } 0 \leqslant z \leqslant \min\{\tau_1, \Delta - 1\}, \text{ that } G_{*\ 1} z \equiv \eta \,(\text{mod S})\}, \tag{8}$$

if relation $G_{*\ 1} z \equiv \eta \,(\text{mod S})$ does not hold for any z, the problem $\psi(1, \eta)$ is infeasible, and we may assume $\psi(1, \eta) = -\infty$. A similar formula is true for the case, when $w_1 > 0$.

Lemma 1. *When solving problem* (3), *there is an algorithm with the complexity* $O(\log \Delta \cdot n\Delta^2 \cdot \text{mult}(s))$, *where* $s = \log ||\tau||_{\max} + \log ||w||_{max} + \log \Delta + \log n$.

Proof. we must determine the value of $\psi(n, g)$ to solve problem (3). Let us find the complexity of one recursion step for computing $\psi(k, \eta)$ using formulas (5) and (6). Due to Remark 2, the bit complexity of the operation $(\eta - zG_{*\ k}) \bmod S$ does not exceed $O(\log \Delta \cdot \text{mult}(\log \tau_k + \log \Delta))$. Since $\psi(k, \eta) \leqslant k \max\{\Delta, \tau_k\} ||w||_\infty$, the complexity of the subsequent addition does not exceed $O(\log n + \log \Delta + \log \tau_k + \log ||w||_\infty)$. Allowing for the enumeration of the values of z. The total complexity of one recursive step does not exceed

$$O(\log \Delta \cdot \Delta \cdot \text{mult}(\log ||\tau||_\infty + \log \Delta) + \Delta \cdot (\log ||w||_\infty + \log n)).$$

Thus, the total complexity of the algorithm thus does not exceed the last value multiplied by $n\Delta$. We obtain resulting estimate $O(\log \Delta \cdot n\Delta^2 \cdot \text{mult}(s))$ by estimating the complexity of addition and multiplication.

Theorem 4. *When solving problem* (2) *there is an algorithm with the complexity*

$$O((n^2 + \log \Delta \cdot n\Delta^2) \cdot \text{mult}(s)),$$

where $s = \log ||b-a||_\infty + \log ||c||_\infty + \log \Delta + \mathrm{size}(A)$. *The size of the elements of the optimum vector* x^* *does not exceed* $O(\mathrm{size}(A) + \log ||b-a||_\infty)$.

Proof. As was shown above, the system (2) can be transformed into the system (3). Let us determine the complexity of this transformation, after which it remains to apply Lemma 1.

The most complicated step of the transformation is to compute the matrix S (the SNF of of the vertex v in S) and the unimodular matrix P from the decomposition $A = PSQ$. The matrix Q is not needed for computations. In accordance with Theorem 3 and Remark 1, the complexity of this step does not exceed $O(n^{\Theta-1} \cdot \mathrm{mult}(\mathrm{size} A))$, and $\mathrm{size}(P) = O(\mathrm{size}(A))$. From Proposition 3, it follows that $\mathrm{size}(||P^{-1}||_{\max}) = O(\mathrm{size}(A))$. The complexity of computing $G = P^{-1} \bmod S$ and $g = P^{-1} b \bmod S$ does not exceed the complexity of computing the NFS. In accordance with Proposition 3, the complexity of computing $w^\top = c^\top A^*$ does not exceed $O(n^2 \mathrm{mult}(\log ||c||_\infty + \mathrm{size}(A)))$, and $\log(||w||_\infty) = O(\log ||c||_\infty + \mathrm{size}(A))$.

By combining the complexity of the transformation and the complexity of Lemma 1 computations, we find the overall complexity does not exceed

$$O(n^{\Theta-1} \cdot \mathrm{mult}(\mathrm{size}(A)) + n^2 \cdot \mathrm{mult}(\log ||c||_\infty) + \log \Delta \cdot n\Delta^2 \cdot \mathrm{mult}(s)).$$

We obtain the required complexity as an upper bound of this estimate.

Finally, the optimum vector x^* is computed using the formula $x^* = A^{-1}(b - y^*)$, where y^* is the optimum vector of the problem (3). Since $||y^*||_\infty \leqslant ||b-a||_\infty$, we find $\mathrm{size}(||x^*||_\infty) = O(\mathrm{size}(A) + \log ||b-a||_\infty)$.

3. COMPUTING SYMPLEX WIDTH

Let $V \in \mathbb{Z}^{n \times (n+1)}$, $\Delta = \Delta(V)$ and $S = \mathrm{conv.hull}(V)$ be a simplex of a dimension n. The aim of this section was to construct an FPT algorithm parameterized by the $\Delta(V)$ parameter to compute the value of $\mathrm{width}(S)$ and the flat direction of the simplex S. The corresponding result is formulated in Theorem 5.

We repeatedly need the following lemma for further reasoning. Its proof follows from the standard theory of convex polyhedra. (See, e.g., [4, 5, 40].)

Lemma 2. *Let* $A \in \mathbb{Z}^{n \times n}$, $b \in \mathbb{Z}^n$ *and* $\det A \neq 0$. Then $P(A, 0) = \mathrm{cone.hull}(-A^*)$ and $P(A, b) = v + \mathrm{cone.hull}(-A^*)$, *where* $v = A^{-1} b$ *is the apex of the polyhedral cone* $P(A, b)$.

Lemma 2 implies that there exists a dual representation for the simplex S. In other words, there exist a matrix $A \in \mathbb{Z}^{(n+1) \times n}$ and a vector $b \in \mathbb{Z}^{n+1}$ such that $S = P(A, b)$. Note too that the coordinate columns of the vertices of simplex S coincide with the columns of the matrix V. Let us identify them.

Let v be the vertex of S. Then $v = V_{*\,i}$ for some $1 \leqslant i \leqslant n+1$. Let us define the matrix $B(v)$ by the following formula

$$B(v) = (V_{*\,1} - V_{*\,i}, \ldots, V_{*\,(i-1)} - V_{*\,i}, V_{*\,(i+1)} - V_{*\,i}, \ldots, V_{*\,(n+1)} - V_{*\,i}).$$

The columns of $B(v)$ are the radius vectors of the edges of S coming from the vertex v. We also denote $A(v)x \leqslant b(v)$ as a subsystem of the system $Ax \leqslant b$ such that $A(v)v = b(v)$. The system $A(v)x \leqslant b(v)$ has exactly n lines. The excluded inequality holds strictly for v. Let us denote $C(v) = v + \mathrm{cone.hull}(B(v))$ as the angular cone of the vertex v in S. According to Lemma 2, the equalities $\mathrm{cone.hull}(B(v)) = P(A(v), 0)$ and $C(v) = P(A(v), b(v))$ hold true. Let us denote $N(v)$ as the cone of normals generated by normal vectors of facets incident to the vertex v. By definition, the equality

$N(v) = \text{cone.hull}(A(v)^\top)$ holds. Therefore, $N(v) = P(-(A(v)^*)^\top, 0)$ according to Lemma 2. It then follows from the equality $C(v) = P(A(v), b(v))$ that $N(v) = P(B(v)^\top, 0)$.

An equality $\mathbb{R}^n = \bigcup\limits_{v \in \text{vert}(S)} N(v)$ is true due to the boundedness of simplex S, so the equality $\mathbb{R}^n = \bigcup\limits_{v,u \in \text{vert}(S)} N(v) \cap (-N(u))$ also holds.

Let $M(v,u) = N(v) \cap (-N(u)) \cap \mathbb{Z}^n \setminus \{0\}$. The formula below then holds for width(S)

$$\begin{aligned} \text{width}(S) &= \min_{v,u \in \text{vert}(S)} \min_{c \in M(v,u)} (\max_{x \in S} c^\top x - \min_{x \in S} c^\top x) \\ &= \min_{v,u \in \text{vert}(S)} \min_{c \in M(v,u)} c^\top (v-u). \end{aligned} \tag{9}$$

The original problem is thus equivalent to solving $n(n-1)/2$ problems of the form $\min\limits_{c \in M(v,u)} c^\top(v-u)$ for different pairs of vertices v, u of S.

We fix an arbitrary pair of vertices v and u of S. The vertices v, u have $n-1$ common facets, since any two vertices of the simplex are adjacent. Let the matrix $F^\top$ consist of rows of the matrix A which correspond to these $n-1$ common facets. We may therefore assume that $N(u) = \text{cone.hull}(F\ a_u)$ and $N(v) = \text{cone.hull}(F\ a_v)$, where $a_u^\top$, $a_v^\top$ are the rows of A, for which the corresponding inequalities of the system $Ax \leqslant b$ are strictl for v and u, respectively. In addition, the relations $-a_u \in N(v)$, $-a_v \in N(u)$ are true, so the relation $a_v - a_u \in M(v,u)$ holds as well.

Let us consider the hyperplane $H(k) = \{x \in \mathbb{R}^n : w^\top x = k\}$, where $w = (v-u)/d$ and d are the greatest common divisors of the elements of the vector $v - u$. Since $\forall c \in M(v,u)$ the inequality $(v-u)c^\top \geqslant 0$ holds, then the equality $M(v,u) = \bigcup\limits_{k \in \mathbb{Z}_+} (M(v,u) \cap H(k))$ is true. Thus,

$$\min_{c \in M(u,v)} c^\top(v-u) = \min\{k \in \{1,2,\ldots,r\} : M(v,u) \cap H(k) \neq \varnothing\}, \tag{10}$$

for some finite r.

The latter means that the problem $\min\limits_{c \in M(v,u)} c^\top(v-u)$ is reduced to r problems of checking the nonemptiness of sets of the form $M(v,u) \cap H(k)$. It is easy to see that $r \leq (n+1)\Delta$. The latter enables us to develop an algorithm for checking the nonemptyness of $M(v,u) \cap H(k)$.

The following lemma describes the structure of sets $M(v,u) \cap H(k)$.

Lemma 4. *Let $k \in \mathbb{R}_+$, then $N(v) \cap (-N(u)) \cap H(k) = (p_v(k) + \text{cone.hull}(F)) \cap (p_u(k) - \text{cone.hull}(F))$, where $p_v(k)$ is the intersection point of the ray $L_v = \{a_v t : t \in \mathbb{R}_+\}$ with the hyperplane $H(k)$ and $p_u(k)$ is the intersection point of the ray $L_u = \{-a_u t : t \in \mathbb{R}_+\}$ with the hyperplane $H(k)$.*

Proof. Let $x \in N(v) \cap (-N(u)) \cap H(k)$, then $x = F\alpha + a_v t_v = -F\beta - a_u t_u$ for some $\alpha, \beta \in \mathbb{Q}_+^{n-1}$ and $t_v, t_u \in \mathbb{Q}_+$. In addition, $w^\top x = k$ is true. $t_v = \frac{k}{w^\top a_v}$ and $t_u = -\frac{k}{w^\top a_u}$ holds because $v^\top F = u^\top F$. Let us consider points $p_v(k)$ and $p_u(k)$. It can be seen that $p_v(k) = a_v t_v$ and $p_u(k) = -a_u t_u$. The inclusion $x \in (p_v(k) + cone(F)) \cap (p_u(k) - cone(F))$ is true because $x = F\alpha + a_v t_v = -F\beta - a_u t_u$.

Let $x \in (p_v(k) + cone(F)) \cap (p_u(k) - cone(F))$. $w^\top x = w^\top p_v(k) = k$ holds, so $x \in H(k)$, since $w^\top F = 0$. Finally, $x \in N(v)$ and $x \in -N(v)$, since points $p_v(k), p_u(k)$ lie on the rays L_v, L_u, which generate rays for the normal cones $N(v)$ and $-N(u)$.

The problem of checking the non-emptyness of the set $M(v,u) \cap H(k)$ is equivalent to checking the feasibility of the system

$$\begin{cases} w^\top x = k \\ B^\top(v)x \leqslant 0 \\ B^\top(u)x \geqslant 0 \\ x \in \mathbb{Z}^n. \end{cases} \tag{11}$$

The elements of the vector w are mutually simple. There is then an unimodular matrix Q such that $w^\top Q = e_1^\top$, where e_1 is the vector for which only the first coordinate is 1 and the others are 0. The transformation exists, since $e_1^\top$ is the HNF for $w^\top$. According to Theorem 3, the matrix Q can be found for time $O(n^{\Theta-1}\mathrm{mult}(n \log ||w||_\infty))$, and $\mathrm{size}(||Q||_{\max}) = O(\log ||w||_\infty)$ is true, in accordance with Remark 1. The system (9) becomes the system below after the transformation $x \to Qx$ and rearranging the rows in the matrices $B(v)$ and $B(u)$

$$\begin{cases} (1, 0, \dots, 0)x = k \\ \begin{pmatrix} -d & 0 \\ \alpha & \hat{B} \end{pmatrix} x \leqslant 0 \\ \begin{pmatrix} d & 0 \\ beta & hatB \end{pmatrix} x \geqslant 0 \\ x \in \mathbb{Z}^n. \end{cases} \tag{12}$$

Let us recall that d is the largest common divisor of the elements of the vector $v - u$ and $w = (v-u)/d$. Let us explain why the system (10) takes this form. First, the matrix $B(v)$ contains the column $u - v$ and the matrix $B(u)$ contains the column $v - u$ by construction. The corresponding rows $B^\top(v)$ and $B^\top(u)$ become $-de_1^\top$ and $de_1^\top$ after substituting $x \to Qx$. It is easy to see from Lemma 4 that there are identical blocks containing the matrix $\hat{B} \in \mathbb{Z}^{(n-1)\times(n-1)}$. Note too that $|\det \hat{B}| \leqslant \Delta$, $\mathrm{size}(||\hat{B}||_{\max}) = O(\log ||V||_{\max})$, $\mathrm{size}(||\alpha||_\infty) = O(\log ||V||_{\max})$ and $\mathrm{size}(||\beta||_\infty) = O(\log ||V||_{\max})$.

After the substitution $x_1 = k$, the system (10) becomes

$$\begin{cases} -k\beta \leqslant \hat{B}x \leqslant -k\alpha, \\ x \in \mathbb{Z}^n. \end{cases} \tag{13}$$

The consistency of this system can be checked using Theorem 4.

CONCLUSIONS

Theorem 5. *Let* $V \in \mathbb{Z}^{n\times(n+1)}$, $\Delta = \Delta(V)$ *and* $S = \mathrm{conv.hull}(V)$ *be a simplex of a dimension* n. *The problem of computing the simplex width* S *and its flat direction can then be solved using an algorithm with the complexity*

$$O(r \cdot n^2 \cdot (n^2 + \log \Delta \cdot n\Delta^2) \cdot \mathrm{mult}(s)),$$

where r *is an estimate of the simplex width, for which the inequality* $r \leq (n+1)\Delta$, *and* $s = n^2 \log ||V||_{\max} + \log \Delta$ holds. *This algorithm is an FPT algorithm with respect to the parameter* Δ.

Proof. The formula (7) shows that the original problem is equivalent to $O(n^2)$ problems of form $\min\limits_{c \in M(u,v)} c^\top(v-u)$ for different pairs of vertices of the simplex S. The formula (8) in turn shows that each problem of the the form $\min\limits_{c \in M(v,u)} c^\top(v-u)$ is equivalent to r problems of checking the non-emptyness of the sets $M(v, u) \cap H(k)$, where $k \in 1 : r$. The problem of checking the non-emptyness of the set $M(v, u) \cap H(k)$ is equivalent to checking the feasibility of the system (11). We obtain an algorithm with the required estimate of a complexity by applying Theorem 4 and estimates of the determinant and the size of the system (11).

4. ACKNOWLEDGMENTS

This work was supported by the Russian Science Foundation, project no. 17-11-01336.

REFERENCES

1. M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh, *Parameterized Algorithms* (Springer, Switzerland, 2015).
2. R. G. Downey and M. R. Fellows, *Parameterized Complexity* (Springer, New York, 1999).
3. B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms* (Springer, Berlin, 2006).
4. A. Schrijver, *Theory of linear and integer programming* (Wiley, Chichester, 1998).
5. V. A. Emelichev, M. M. Kovalev, and M. K. Kravtsov, *Polyhedra, Graphs, Optimization* (Nauka, Moscow, 1981) [in Russian].
6. F. Eisenbrand, "Integer programming and algorithmic geometry of numbers," in *50 Years of Integer Programming,* Ed. by M. Jünger, T. Liebling, D. Naddef, W. Pulleyblank, G. Reinelt, G. Rinaldi, and L. Wolsey (Springer, New York, 2010), pp. 1958–2008.
7. R. E. Gomory, "On the relation between integer and non-integer solutions to linear programs," Proc. Natl. Acad. Sci. U.S.A. **53**, 260–265 (1965).
8. R. E. Gomory, "Integer faces of a polyhedron," Proc. Natl. Acad. Sci. U.S.A. **57**, 16–18 (1967).
9. R. E. Gomory, "Some polyhedra related to combinatorial problems," J. Linear Algebra Appl. **2**, 451–558 (1969).
10. T. C. Hu, *Integer Programming and Network Flows* (Addison-Wesley, Reading, Mass, 1970).
11. V. N. Shevchenko, *Qualitative Integral Programming Issues* (Fizmatlit, Moscow, 1995) [in Russian].
12. S. Artmann, F. Eisenbrand, C. Glanzer, O. Timm, S. Vempala, and R. Weismantel, "A note on non-degenerate integer programs with small subdeterminants," Operat. Res. Lett. **44**, 635–639 (2016).
13. D. V. Gribanov, D. S. Malyshev, P. M. Pardalos, and S. I. Veselov, "FPT-algorithms for some problems related to integer programming," Comb. Optim. **35**, 1128–1146 (2018).
14. N. Karmarkar, "A new polynomial time algorithm for linear programming," Combinatorica **4**, 373–391 (1984).
15. L. G. Khachiyan, "Polynomial algorithms in linear programming," Comput. Math. Math. Phys. **20**, 53–72 (1980).
16. Y. E. Nesterov and A. S. Nemirovsky, *Interior Point Polynomial Methods in Convex Programming* (Soc. Ind. Appl. Math., Philadelphia, 1994).
17. P. M. Pardalos, C. G. Han, and Y. Ye, "Interior point algorithms for solving nonlinear optimization problems," COAL Newslett. **19**, 45–54 (1991).
18. S. I. Veselov and A. J. Chirkov, "Integer program with bimodular matrix," Discrete Optimiz. **6**, 220–222 (2009).
19. S. Artmann, R. Weismantel, and R. Zenklusen, "A strongly polynomial algorithm for bimodular integer linear programming," in *Proceedings of 49th Annual ACM Symposium on Theory of Computing* (ACM, New York, 2017), pp. 1206–1219.
20. V. V. Alekseev and D. Zakharova, "Independent sets in the graphs with bounded minors of the extended incidence matrix," J. Appl. Ind. Math. **5**, 14–18 (2011).
21. D. Gribanov and D. Malyshev, "The computational complexity of dominating set problems for instances with bounded minors of constraint matrices," Discrete Optimiz. (2018, in press).
22. D. V. Gribanov and D. S. Malyshev, "The computational complexity of three graph problems for instances with bounded minors of constraint matrices," Discrete Appl. Math. **227**, 13–20 (2017).
23. D. V. Gribanov and D. S. Malyshev, "The complexity of some problems on graphs with bounded minors of their constraint matrices," Zh. Srednevolzh. Mat. Ob-va **18** (3), 19–31 (2016).
24. D. Malyshev, "A complexity dichotomy and a new boundary class for the dominating set problem," J. Combin. Optimiz. **32**, 226–243 (2016).
25. D. Malyshev, "A dichotomy for the dominating set problem for classes defined by small forbidden induced subgraphs," Discrete Appl. Math. **203**, 117–126 (2016).
26. D. Malyshev and P. M. Pardalos, "Critical hereditary graph classes: a survey," Optimiz. Lett. **10**, 1593–1612 (2016).
27. D. Malyshev and P. M. Pardalos, "The clique problem for graphs with a few eigenvalues of the same sign," Optimiz. Lett. **9**, 839–843 (2015).
28. D. S. Malyshev, "Critical elements in combinatorially closed families of classes of graphs," Diskret. Anal. Issled. Operatsii **24** (1), 81–96 (2017).
29. D. S. Malyshev, "Critical classes of graphs for the edge list ranking problem," Diskret. Anal. Issled. Operatsii **20** (6), 59–76 (2013).
30. F. Eisenbrand, "Fast Integer Programming in Fixed Dimension," Lect. Notes Comput. Sci. **2832**, 196–207 (2003).
31. D. Dadush, "Integer programming, lattice algorithms, and deterministic volume estimation," PhD Thesis (Georgia Inst. Technol., Ann Arbor, MI, 2012).
32. S. Heinz, "Complexity of integer quasiconvex polynomial optimization," J. Complexity **21**, 543–556 (2005).

33. R. Hildebrand and M. Köppe, "A new lenstra-type algorithm for quasiconvex polynomial integer minimization with complexity $2^{O(n \log n)}$," Discrete Optimiz. **10**, 69–84 (2013).
34. H. W. Lenstra, "Integer programming with a fixed number of variables," Math. Operations Res. **8**, 538–548 (1983).
35. A. Sebö, "An introduction to empty lattice simplicies," Lect. Notes Comput. Sci. **1610**, 400–414 (1999).
36. D. V. Gribanov and A. J. Chirkov, "The width and integer optimization on simplices with bounded minors of the constraint matrices," Optimiz. Lett. **10**, 1179–1189 (2016).
37. V. V. Prasolov, *Problems and Theorems of Linear Algebra* (MTsNMO, Moscow, 2015) [in Russian].
38. A. Storjohann, "Near optimal algorithms for computing Smith normal forms of integer matrices," in *Proceedings of the 1996 International Symposium on Symbolic and Algebraic Computation* (ACM, New York, 1996), pp. 267–274.
39. A. Storjohann and G. Labahn, "Asymptotically fast computation of Hermite normal forms of integer matrices," in *Proceedings of the 1996 International Symposium on Symbolic and Algebraic Computation* (1996), pp. 259–266.
40. G. Ziegler, *Lectures on Polytopes* (Springer, New York, 1996).

Translated by O. Pismenov