International Journal of Computer Mathematics, 76(1/2):59–74, 2000.

Complexity Results for Some Eigenvector Problems

Thomas Eiter and Georg Gottlob

Information Systems Department, TU Vienna
Paniglgasse 16, A-1040 Wien, Austria
eiter@kr.tuwien.ac.at,
gottlob@dbai.tuwien.ac.at
Fax: ++431-58801-18493

Abstract

We consider the computation of eigenvectors $\mathbf{x} = (x_1, \ldots, x_n)$ over the integers, where each component x_i satisfies $|x_i| \leq b$ for an integer b. We address various problems in this context, and analyze their computational complexity. We find that different problems are complete for the complexity classes NP, $\mathrm{P}^{\mathrm{NP}}_{\|}$, FNP//OptP[$O(\log n)$], $\mathrm{FP}^{\mathrm{NP}}$, P^{NP}, and $\mathrm{NP}^{\mathrm{NP}}$. Applying the results, finding bounded solutions of a Diophantine equation $\mathbf{v} \cdot \mathbf{x}^{\mathbf{T}} = 0$ is shown to be intractable.

Keywords: mathematical programming, eigenvectors, problem complexity, combinatorial optimization

Computing Reviews Categories: F.2.1. [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems – *computation on matrices*; G.1.6 [**Numerical Analysis**]: Optimization – *integer programming*.

1 Introduction

Eigenvalues and eigenvectors have important applications in many areas, e.g. to problems in structural analysis, quantum chemistry, power system analysis, stability analysis, VLSI design, and geophysics [2]. The computation of eigenvalues and eigenvectors is thus an important problem, which has been investigated intensively in the past; see e.g. [3, 5, 11] and references therein.

In this paper, we address the complexity of computing distinguished elements out of the in general infinite set of eigenvectors for a given eigenvalue λ of a matrix M over the integers $\mathbb{Z}$. In particular, we consider the computation of eigenvectors within a box of $\mathbb{Z}^n$, i.e., the set of vectors $\mathbf{v} = (v_1, \ldots, v_n)$ such that the absolute value $|v_i|$ of each component v_i is at most b; we call such vectors *b-bounded*. Observe that in programming languages, the range of integers is usually b-bounded for some constant $b \geq 1$.

As with the computation of eigenvectors, there is particular interest in computing shortest eigenvectors, i.e., a non-zero eigenvector $\mathbf{v}$ such that its length $\|\mathbf{v}\|$, which is understood in terms of the L_2 (euclidean) norm, is smallest. For this problem e.g. the algorithm of Håstad et al. [6] for finding integer relationships between real vectors can be employed, which is closely related to the Lovasz-Lenstra-Lenstra (L^3) algorithm [9]. Given linearly independent vectors $\mathbf{v}_1, \ldots, \mathbf{v}_s \in \mathbb{Z}^n$, and $k \geq 0$, the algorithm in [6] finds a vector $\mathbf{x} \in \mathbb{Z}^n$ in polynomial time such that $\mathbf{v}_i \cdot \mathbf{x}^{\mathbf{T}} = 0$ for all $i = 1, \ldots, s$ or reports that no such vector of length $\leq 2^k$ exists. The vector computed is not shortest, but usually shorter than a vector obtained by simple algorithm such as a standard Gaussian elimination. Furthermore, the algorithm does not return a b-bounded vector in general, and it is not clear whether the algorithm could be modified in this respect.

The main contributions of the present paper can be summarized as follows:

- We give a precise characterization of the computational complexity of different problems in the context of computing b-bounded eigenvectors over $\mathbb{Z}$. As we show, this problem is intractable in general. In particular, we show that computing a shortest b-bounded eigenvector is complete for $\mathrm{FP}^{\mathrm{NP}}$ and, if b is a constant, complete for the class $\mathrm{FNP}//\mathrm{OptP}[O(\log n)]$ introduced by Chen and Toda [1]. Few natural problems which are complete for this class are known so far.
- By means of this complexity characterization, appropriate algorithm schemes for the solution of these problems emerge.
- We provide several different problems, which can be used to establish similar hardness results for related problems.

To our knowledge, the complexity of these problems has not been considered before.

2 Preliminaries

Eigenvalues and Eigenvectors Let R be a ring with 1. Recall that $\lambda \in R$ is an eigenvalue of an $n \times n$ matrix $M = (m_{i,j})$ over R if the equation

$$M \cdot \mathbf{x}^{\mathbf{T}} = \lambda \mathbf{x}^{\mathbf{T}}$$

has nontrivial solutions, i.e., solutions $\mathbf{x} \neq \mathbf{0}$, where $\mathbf{x} = (x_1, \ldots, x_n)$, $\mathbf{x}^{\mathbf{T}} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$ is the transpose of $\mathbf{x}$, and $\mathbf{0} = (0, \ldots, 0)$ is the zero vector; all vectors $\mathbf{x}$ that satisfy this equation are eigenvectors (for the eigenvalue λ). It is well-known that for any eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$, all vectors $\sum_{i=1}^{r} a_i \mathbf{v}_i$, where $a_i \in \mathbb{Z}$, are eigenvectors; if R is a field, then the set of all eigenvectors for λ form a vector space, whose dimension is the multiplicity of λ as root of the characteristic polynomial of M. In this paper, we restrtc attention to $R = \mathbb{Z}$.

Recall that the L_2 norm of an integer vector $\mathbf{x}$, $\|\mathbf{x}\|$, is defined by $\|\mathbf{x}\| = \left(\sum_{i=1}^{n} x_i^2\right)^{1/2}$. Vector $\mathbf{x} \in V$ is *maximal* in a set of vectors V if and only if $\|\mathbf{y}\| \leq \|\mathbf{x}\|$, for every $\mathbf{y} \in V$. We say that vector $\mathbf{x}$ is *bounded* by a an integer $b \geq 0$ (*b-bounded*), if $|x_i| \leq b$, for every $i = 1, \ldots, n$.

Computational Complexity We assume that the reader has some knowledge about computational complexity. Excellent sources are [4, 7], which we refer to for background information.

Computational problems are encoded over the alphabet $\Sigma = \{0, 1\}$, for which a standard one-to-one polynomial-time invertible pairing function $\langle x, y \rangle$ is available. A language is a subset of Σ^*, and a function is a partial map $f : \Sigma^* \rightarrow \Sigma^*$. A multi-valued function g is a map $g : \Sigma^* \rightarrow 2^{\Sigma^*}$, where $g(x)$ is considered undefined if $g(x) = \emptyset$.

The complexity classes considered are defined using variants of standard (possibly nondeterministic) Turing machines (TMs), and are either acceptors or transducers. On a given input x, a branch of a nondeterministic TM M may halt in an accepting or rejecting state. The language accepted by M is the set of all strings which are accepted by M. A transducer T computes a string y on input x, if some branch halts in an accepting state and y is on the output tape of T. Every deterministic (resp. nondeterministic) transducer T computes a function f (resp., multi-valued function g) such that $f(x) = y$ (resp., $y \in g(x)$) iff M computes y on input x, for every $x \in \Sigma^*$.

P (resp. NP) is the class of decision problems (identified with languages) solved by a polynomial-time deterministic TM (resp. nondeterministic TM), and FP is the functional version of P. The class P^{NP} (resp., $\mathrm{NP}^{\mathrm{NP}}$) contains the decision problems solved in polynomial time by a deterministic (resp., nondeterministic) TM with an oracle for NP. The class $\mathrm{FP}^{\mathrm{NP}}$ is the functional version of P^{NP}. The class $\mathrm{P}_{||}^{\mathrm{NP}}$ is the variant of P^{NP} in which all oracle calls must be run in parallel, i.e., no subseqeunt call of the oracle is possible.

A NP *metric Turing machine* [8] is a polynomial-time bounded TM T, such that on input x every computation branch halts and outputs a binary number; the result of T on x is the maximum over all these numbers. The class OptP contains all (total) integer functions f which are computable by an NP metric TM. The class $\mathrm{OptP}[O(\log n)]$ is the subclass of OptP in which the output $f(x)$ has $O(|x|)$ many bits, where $|x|$ is the length of x, i.e., $f(x)$ is polynomial in $|x|$. The class FNP//OptP$[O(\log n)]$, introduced in [1], contains all (partial) multi-valued functions g for which a polynomial-bounded nondeterministic transducer T and a function $h \in \mathrm{OptP}[O(\log n)]$ exist such that for every x, $g(x) = T(\langle x, h(x)\rangle)$.

A function f_1 (resp., multi-valued function g_1) is (polynomial-time) reducible to a function f_2 (resp., multi-valued function g_2) if there is a pair of polynomial functions h_1, h_2 such that, for every x, $h_1(x)$ is defined, and $f_1(x)$ = $h_2(x, w)$ where $w = f_2(h_1(x))$ (resp., $h_2(x, w) \in g_1(x)$ for every $w \in g_2(h_1(x))$, and some w exists if $f_1(x)$ is defined.) A (single- or multi-valued) function f is hard for a class of (single- or multi-valued) functions $\mathbf{F}$, if every $g \in \mathbf{F}$ is reducible f, and is complete for $\mathbf{F}$, if it is hard for $\mathbf{F}$ and belongs to $\mathbf{F}$.

A computational problem Π is modeled (or "solved") by a function f (resp., multivalued function g) if given any instance I of Π encoded by a string x, $f(x)$ is defined (resp., $g(x) \neq \emptyset$) iff I has some solution, and $f(x)$ is the solution (resp., each $w \in g(x)$ is a solution) for instance I. Furthermore, a problem is hard (resp. complete) for a class of functions $\mathbf{F}$, if it is modeled by some function which is hard (resp. complete) for $\mathbf{F}$. E.g., computing some optimal tour in the Traveling Salesman Problem, as well as the cost of an optimal tour, is complete for $\mathrm{FP}^{\mathrm{NP}}$.

We remark that a class FNP//OptP can be defined analogous to FNP//OptP$[O(\log n)]$ by replacing "$h \in \mathrm{OptP}[O(\log n)]$" with "$h \in \mathrm{OptP}$" in the definition. It is easy to show that every (multi-valued) function $g \in$ FNP//OptP has a refinement (single-valued) function $f \in \mathrm{FP}^{\mathrm{NP}}$, i.e., for every x it holds that $g(x)$ is defined iff $f(x)$ is defined and $f(x) \in g(x)$. Thus, a problem (with possibly multiple solutions for a given instance) is solvable in FNP//OptP iff it is solvable in $\mathrm{FP}^{\mathrm{NP}}$. Even if problems in $\mathrm{FP}^{\mathrm{NP}}$ that have multiple possible solutions (e.g., computing an optimal tour in the Traveling Salesman Problem) may be more naturally modeled by functions in FNP//OptP, we use here the class $\mathrm{FP}^{\mathrm{NP}}$, which is more widely known and reflects more appropriately the nature of deterministic algorithms used in practice.

3 Problem Statements

We assume tacitly that vectors and matrices are over the integers $\mathbb{Z}$. We consider the following problems:

Problem P1: Given an $n \times n$ matrix M, an integer eigenvalueλ of M, a real number K, and a bound $b \geq 1$, does there exist a b-bounded non-zero eigenvector $\mathbf{x}$ for λ such that $\|\mathbf{x}\| \leq K$?[1]

This problem is the decision problem naturally associated with the problem of computing a shortest b-bounded eigenvector $\mathbf{x}$. It is related to integer and quadratic programming problems (see [4]). We show that P1 is NP-complete, and hardness holds even if $K = \sqrt{n}b$, i.e., deciding whether any b-bounded eigenvector exists is NP-complete. Thus, the algorithm of Håstad et al. [6] can not be modified to find a b-bounded nonzero integer relationship among vectors $\mathbf{v}_1, \ldots, \mathbf{v}_s$ in polynomial time. As shown in Section 5, this holds even if $s = 1$, i.e., for a single vector.

Problem P2: Given an $n \times n$ matrix M, an integer eigenvalue λ of M, and an integer b, compute a shortest eigenvector $\mathbf{x}$ among the b-bounded eigenvectors for λ.

Intuitively, solving this problem requires computing the length $\|\mathbf{x}\|$ of a shortest b-bounded eigenvector, and generating an eigenvector of that norm. This problem is complete for $\mathrm{FP}^{\mathrm{NP}}$ in general, and for FNP//OptP$[O(\log n)]$ if b is fixed to any constant $c \geq 1$.

Problem P3: Given an $n \times n$ matrix M, an integer eigenvalue λ of M, and integers b and z, decide if there is any shortest eigenvector $\mathbf{x}$ among the b-bounded eigenvectors for λ with $x_1 = z$, i.e., the first component of $\mathbf{x}$ is z.

This problem is P^{NP}-complete in general, and $\mathrm{P}^{\mathrm{NP}}_{\|}$-complete if b is fixed to $c \geq 1$.

Problem P4: Given an $n \times n$ matrix M and an integer eigenvalue λ of M, compute the lexicographically first among the shortest b-bounded eigenvectors for λ.

Selection of the first vector under lexicographical ordering or a similar ordering is a natural choice. This problem is $\mathrm{FP}^{\mathrm{NP}}$-complete, reardless of fixing b to a constant $c \geq 1$ or not.

Problem P5: Given an $n \times n$ matrix M, an integer eigenvalue λ of M, a subset I of the components, and integers b, z, does there exist a b-bounded $\preceq$-minimal nonzero eigenvector $\mathbf{x}$ for λ such that $x_1 = z$, where $\mathbf{x} \preceq \mathbf{y}$ if and only if $\mathbf{x}$ and $\mathbf{y}$ coincide on the components in I and $\|\mathbf{x}\| \leq \|\mathbf{y}\|$.

[1] Note that P1-P5 are trivial if λ is irrational, and can be easily reduced to the integer case if it is a rational number.

This problem subsumes P3 as a special case if $I = \emptyset$. Here, the comparability between different vectors $\mathbf{x}$ and $\mathbf{y}$ is restricted to vectors which coincide on a given part I of the components. As we will see, however, this restriction on comparability does not decrease the complexity; on the contrary, it increases the complexity from P^{NP} to $\mathrm{NP}^{\mathrm{NP}}$.

Notice that in all problems P1–P5, correct problem instances can be recognized in polynomial time, since deciding whether λ is an eigenvalue of M can be done in polynomial time (e.g. using linear programming or Gaussian elimination).

4 Complexity Results

For determining the complexity of problems P1–P5, we refer to variants of problems involving the classical satisfiability problem SAT. Let $\varphi = \{C_1, \ldots, C_m\}$ be a set of propositional clauses C_i on variables X. A truth assignment τ to X satisfies φ, if each clause $C \in \varphi$ contains at least one literal (i.e., variable of negated variable) with value *true*. An assignment τ is *not-all-equal satisfying* (*nae-satisfying*) for φ, if each clause in φ contains two literals that have different value according to τ; clearly, each nae-satisfying assignment for φ satisfies φ in the standard sense. Moreover, if σ is an nae-satisfying assignment, then also the complementary assignment $\overline{\sigma}$, in which each variable has opposite truth value, is nae-satisfying.

Let $\varphi = \{C_1, \ldots, C_m\}$ be an instance of 3SAT, i.e., a set of propositional clauses $C_i = \alpha_{i,1} \vee \alpha_{i,2} \vee \alpha_{i,3}$, $i = 1, \ldots, m$ on variables $X = \{x_1, \ldots, x_n\}$. Then denote by φ' the set of the following clauses:

- $x_j \vee x_j^* \vee z_j$ and $x_j \vee x_j^* \vee \neg z_j$, for each $j = 1 \ldots, n$,
- $\alpha_{i,1}^* \vee \alpha_{i,2}^* \vee w_i$ and $\alpha_{3,i}^* \vee x_0 \vee \neg w_i$, for each $i = 1, \ldots, m$

where x_0, all z_j, all x_j^*, and all w_i are fresh variables and $\alpha_{i,j}^* = x_\ell$, if $\alpha_{i,j} = x_\ell$,and $\alpha_{i,j}^* = x_\ell^*$ if $\alpha_{i,j} = \neg x_\ell$.

The following is easily verified. Let τ be an nae-satisfying assignment for φ'. If $\tau(x_0) = \mathit{false}$, then τ, restricted to X, satisfies φ; if $\tau(x_0) = \mathit{true}$, then the complementary assignment $\overline{\tau}$, restricted to X, satisfies φ. On the other hand, if an assignment σ satisfies φ, then σ is extendible to at least one nae-satisfying assignment of φ' in which $x_0 = \mathit{false}$. Thus, we obtain the following.

Lemma 4.1 *Let φ be any 3SAT instance on variables X. Then, the nae-satisfying assignments τ of φ' such that $\tau(x_0) = \mathit{false}$, correspond on the variables X 1-1 to the satisfying assignments of φ.*

As a consequence, deciding whether a SAT instance is satisfiable under nae-satisfaction (NAESAT) is NP-hard [4], even if all clauses have size 3 (NAE3SAT).

We now turn to Problem P1 from above, and obtain our first result.

Theorem 4.2 *Problem P1 is* NP*-complete. Hardness holds if b is fixed to an arbitrary constant $c \geq 1$.*

Proof. Membership in NP is clear, since a guess for a suitable eigenvector $\mathbf{x}$ has polynomially many bits in the size of the input and can be verified in polynomial time.

We show hardness by a reduction from NAE3SAT. Let $\varphi = \{C_1, \ldots, C_m\}$ be a 3CNF on variables $X = \{x_1, \ldots, x_n\}$.

We will describe the matrix M in terms of the equations emerging for each component x_i from the equation $M \cdot \mathbf{x}^\mathbf{T} = \lambda \mathbf{x}^\mathbf{T}$; the eigenvalue λ is 1.[2] That is, we state for each x_i the equation

$$\sum_j m_{i,j} \cdot x_j = x_i$$

Unless stated otherwise, the equation is $1 \cdot x_i = x_i$ (i.e., the i-th row of M has 1 at column i and 0 everywhere else).

We construct M as a $k \times k$ matrix, $k = 3n + m$, as follows. The components $x_1, \ldots, x_n$ of an eigenvector $\mathbf{x} = (x_1, \ldots, x_k)$ are supposed to correspond to the variables $x_1, \ldots, x_n$ of φ, and the component values to (partial) truth assignments. For each $i = 1, \ldots, n$, we have the equation

$$c \cdot x_{n+i} = x_i; \tag{1}$$

any c-bounded solution requires that (x_i, x_{n+i}) is one of $(0,0)$, $(c,1)$, $(-c,-1)$. Intuitively, this corresponds to a partial truth assignment, where $(c,1)$ means that variable x_i of φ is set true, $(-c,-1)$ that x_i is set false, and $(0,0)$ that x_i is undefined.

For each clause $C_i = x_{i_1}^{s_{i,1}} \vee x_{i_2}^{s_{i,2}} \vee x_{i_3}^{s_{i,3}}$ in φ, where $s_{i,j} \in \{0,1\}$ (as usual x^0 is the literal x and x^1 the literal $\neg x$), we have the equation

$$(-1)^{s_{i,1}} \cdot x_{i_1} + (-1)^{s_{i,2}} \cdot x_{i_2} + (-1)^{s_{i,3}} \cdot x_{i_3} \quad = \quad x_{3n+i} \tag{2}$$

The value of x_{3n+i} in any c-bounded eigenvector $\mathbf{x}$ is either 0 or $\pm c$. If all literals in C_i have assigned a truth value, i.e., $x_{i_j} = \pm c$, then x_{3n+i} must have value $\pm c$; this is only possible if two terms $(-1)^{s_{i,j}} x_{i_j}$ add up to 0, i.e., the corresponding literals L_j have opposite value under the truth assignment represented by $\mathbf{x}$.

Now we add further equations, for all $i = 1, \ldots, n$:

$$x_{n+1} + x_{n+i} - x_{2n+i} \quad = \quad x_{2n+i} \tag{3}$$

They have the following effect. If $x_{n+1} = 0$, i.e., x_1 is not assigned a value, then also $x_{n+i} = 0$ must hold in any c-bounded eigenvector $\mathbf{x}$, which implies $\mathbf{x} = \mathbf{0}$. On the other hand, if $x_{n+1} = \pm 1$, then also x_{n+i} must have a value from $-1, 1$, otherwise Equation (3) can not hold.

It should be clear from above how the matrix M is completed. Notice that $\lambda = 1$ is an eigenvalue of M (e.g., set $x_i = c$, for all $i = 1, \ldots, n$, and choose the other components appropriately).

It holds that in any non-zero c-bounded eigenvector $\mathbf{x}$ for M and λ, every x_i, $i = 1, \ldots, n$ must have value $\pm c$. Furthermore, the nae-satisfying assignments of φ correspond 1-1 to the non-zero eigenvectors. Thus, for $K = \sqrt{(3n + m)c^2}$ the maximum possible length of a c-bounded vector, it holds that M, λ, c, and K is a Yes-instance of P1 if and only if φ is nae-satisfiable. This proves the result. ∎

[2] Here and in the other proofs, we take $\lambda = 1$; it is easy to see that 1 is indeed always an eigenvalue of M.

Theorem 4.3 *Problem P2, i.e., computing a shortest non-zero b-bounded eigenvector, is complete for* FNP//OptP$[O(\log n)]$*, if b is fixed to $c \geq 1$.*

Proof. The problem is in FNP//OptP$[O(\log n)]$: The length ℓ of a shortest c-bounded eigenvector is at most $\sqrt{n}c$, which means that it has $O(\log n)$ bits. Moreover, it can be computed by a NP metric TM, and thus computing ℓ is in $\mathrm{OptP}[O(\log n)]$. An eigenvector of length ℓ can be guessed and verified in polynomial time; hence, computing a shortest eigenvector is in FNP//OptP$[O(\log n)]$.

For the hardness part, we employ a reduction from the following problem. Given an NAE3SAT instance φ on variables X, a subset $X' \subseteq X$ and a variable $x_i \in X'$, call a nae-satisfying assignment σ of φ $x_i; X'$*-minimal*, if the set $\{x_j \in X' \mid \sigma(x_j) = \sigma(x_i)\}$ is minimal over all nae-satisfying assignments σ with respect to inclusion.

Lemma 4.4 *Given an NAE3SAT instance φ on variables X, a subset X', and a variable $x_i \in X'$, computing a $x_i; X'$-minimal nae-satisfying assignment of φ is* FNP//OptP$[O(\log n)]$*-hard.*

Proof. This can be shown by a reduction from the following problem, which was proved FNP//log-complete in [1] (X-MAXIMAL MODEL): Given a CNF φ on variables X and a subset $X' \subseteq X$, compute an assignment σ to the variables in X' such that $C\sigma$ is satisfiable and for no assignment τ to X' such that $\sigma < \tau$ under usual truth ordering, $C\tau$ is satisfiable.

Without loss of generality, φ is only satisfiable if a distinguished variable $x_i \in X'$ is set to true. Using fresh variables, φ can be rewritten by splitting clauses in the standard way (replace $C = C_1 \vee C_2$ by $C_1 \vee y$ and $\neg y \vee C_2$) to a 3CNF φ^* such that the X'-maximal models of φ and φ^* coincide. We then apply to φ^* the transformation from 3SAT to NAE3SAT outlined at the beginning of Section 4, and obtain a NAE3SAT instance φ'. By Lemma 4.1 and the observations preceding it, each $x_i; X'$-minimal nae-satisfying assignment τ of φ^* corresponds to a X'-maximal (partial) model σ of φ, given by $\sigma(x_j) \equiv \tau(x_i) \neq \tau(x_j)$, for all $x_j \in X'$, and conversely for every σ at least one such τ exists. Since φ^* and σ are constructible in polynomial time from φ, X', x_i and φ, X', x_i, τ, respectively, the result follows. ■

The proof is an extension to the construction in the proof of Theorem 4.2. Let φ, $X' = \{x_1, \ldots, x_k\}$, and $x_i = x_1$ be an instance of the problem in Lemma 4.4. Suppose the equations (1)–(3) have already been established for φ. We introduce for each variable $x_j \in X \setminus X'$ further components z_j for a vector, $j = k+1, \ldots, n$, and set up the following equation:[3]

$$x_{n+1} - x_{n+j} - z_j \quad = \quad z_j \tag{4}$$

This equation is similar to Equation (3), which assigns in any non-zero c-bounded eigenvector x_{2n+j} the value ± 1, if $x_{n+1} = x_{n+j}$, and the value 0 otherwise (i.e., if $x_{n+1} = -x_{n+j}$); Equation (4) does just the opposite.

Let M and $\lambda = 1$ ($b = c$) be the resulting instance of P2. Then, similar as in proof of Theorem 4.2, the non-zero c-bounded eigenvectors $\mathbf{x}$ of M correspond 1-1 to the nae-satisfying assignments. Each such

[3]Here and in the rest of the paper, for better readability we use component names z_i, y_i etc which can be easily transformed to names $x_1, \ldots, x_n$ as stated in problems P1–P5.

vector $\mathbf{x}$ satisfies $\|\mathbf{x}\|^2 = (m+n)c^2 + n + n - k + 1 + eq(\mathbf{x})$, where $eq(\mathbf{x}) = |\{j \mid x_1 = x_j$ and $1 \leq j \leq k\}|$ is the number of components among $x_1, \ldots, x_k$ which coincide with x_1.

Thus, a shortest c-bounded eigenvector $\mathbf{x}$ of M corresponds to an nae-satisfying assignment σ of φ in which $|\{x_j \in X \mid \sigma(x_1) = \sigma(x_j)\}|$ is minimum. Clearly, every such σ is an $x_1; X'$-minimal nae-satisfying assignment for φ.

The matrix M and $\lambda = 1$ can be constructed in polynomial time from φ, and from any shortest c-bounded eigenvector of M, the corresponding $x_1; X'$-minimal nae-satisfying assignment σ is constructible in polynomial time; this proves hardness for FNP//OptP$[O(\log n)]$. ■

For an assignment σ to Boolean variables X, denote by $\sigma[X']$ the restriction of σ to $X' \subseteq X$. Recall that a truth assignment τ to an ordered set of variables $X' = \{x_1, \ldots, x_n\}$ is lexicographically smaller than a truth assignment σ, denoted $\tau <_{lex} \sigma$, if $\tau(x_i) = \mathit{false}$ for the least index i such that $\tau(x_i) \neq \sigma(x_i)$.

Theorem 4.5 *Problem P2, i.e., computing a shortest non-zero b-bounded eigenvector, is* $\mathrm{FP}^{\mathrm{NP}}$*-complete.*

Proof. The membership part follows from Theorem 4.11 below.

For the hardness part, we employ a reduction from the following problem. Given an NAE3SAT instance φ on variables X, $X' \subseteq X$, and a variable $x_i \in X$, call any nae-satisfying assignment σ of X *lexicographic* $x_i; X'$*-maximal*, if the assignment σ' to X given by $\sigma'(x_i) \equiv \sigma(x_1) = \sigma(x_i)$, for all $x_i \in X' = \{x_1, \ldots, x_k\}$, is lexicographic maximal over all such σ'. Similar as in the proof of Lemma 4.4, and using the result that computing the lexicographic maximal satisfying assignment of a propositional CNF φ is $\mathrm{FP}^{\mathrm{NP}}$-complete [8], the following can be shown.

Lemma 4.6 *Given an NAE3SAT instance φ on variables X, a subset $X' \subseteq X$, and a variable $x_i \in X'$, computing a lexicographic $x_i; X'$-maximal nae-satisfying assignment of φ is* $\mathrm{FP}^{\mathrm{NP}}$*-hard.*

We reduce the problem in Lemma 4.6 to computing a b-bounded eigenvector by adapting the proof of Theorem 4.3 as follows.

(a) Equation (4) is set up for all $j = 1, \ldots, n$;

(b) for each $j = 1, \ldots, k$, where $X' = \{x_1, \ldots, x_k\}$, add the equation

$$2^{n-j+1} x_{n+1} - 2^{n-j+1} x_{n+j} \quad = \quad w_j. \tag{5}$$

We then set $b = 2^n$. The effect of these changes is the following. In any non-zero 2^n-bounded eigenvector $\mathbf{x}$, each component x_i must be set to $\pm 2^n$, and x_{n+i} to ± 1. The vectors $\mathbf{x}$ correspond 1-1 to the nae-satisfying assignments of φ. Moreover, the shortest eigenvectors $\mathbf{x}$ correspond 1-1 to the lexicographic $x_1; X'$-maximal nae-satisfying assignments σ of φ, and some σ is easily obtained from any such $\mathbf{x}$.

Since M, $\lambda = 1$ and $b = 2^n$ are polynomial-time constructible from φ, X and x_1, the results follows. ■

Theorem 4.7 *Problem P3 is* P^{NP}*-complete.*

Proof. The length ℓ of a shortest non-zero b-bounded eigenvector $\mathbf{x}$ can be computed in polynomial time with a polynomial number of calls to an NP-oracle (query instances of P1), doing a binary search on for K on $[1, \sqrt{n}b]$. Given ℓ, deciding querying the NP oracle wehther a b-bounded nonzero eigenvector $\mathbf{x}$ exists such that $x_1 = z$. Hence, the problem is in P^{NP}.

The hardness part is established extending the reduction in the proof of Theorem 4.7: From the result that deciding a given bit of the lexicographic maximal satisfying assignment of a propositional CNF is P^{NP}-complete [8], deciding whether $\sigma(x_i) = \sigma(x_j)$ in the (unique) lexicographic $x_i; X'$-maximal nae-satisfying assignment of an NAE3SAT instance φ for $X' = X$ and $x_i, x_j \in X$ is P^{NP}-hard. The condition $\tau(x_i) = \tau(x_j)$ is equivalent to $w_j = 0$, where w_j is from Equation 5. This proves the result. ■

Theorem 4.8 *Problem P3 is* $\mathrm{P}_{\parallel}^{\mathrm{NP}}$*-complete, if* b *is fixed to* $c \geq 1$.

Proof. The length ℓ of a shortest c-bounded vector is at most $\sqrt{nc^2}$, and can be easily determined from the result of a polynomia number of parallel queries to an NP oracle whether $\ell \leq K$ where $K = \sqrt{1}, \sqrt{2}, \ldots, \sqrt{nc^2}$. Further parallel queries to NP oracles can determine if some c-bounded eigenvector of length K exists such that $x = z_1$. Given all queries results, problem P3 is easily answered. Hence, it is in $\mathrm{P}_{\parallel}^{\mathrm{NP}}$.

For the hardness part, we use the following lemma:

Lemma 4.9 *Given an NAE3SAT instance* φ *on variables* X*, a subset* $X' \subseteq X$*, and variables* $x_i \in X'$*,* $x_j \in X \setminus X'$*, deciding whether some nae-satisfying assignment of* φ *exist such that* $|\{x_j \in X' \mid \sigma(x_j) = \sigma(x_i)\}|$ *is smallest (call such a* σ $x_i; X'$*-minimum) and* $\sigma(x_i) = \sigma(x_k)$ *is* $\mathrm{P}_{\parallel}^{\mathrm{NP}}$*-complete. is* FNP//OptP$[O(\log n)]$*-hard.*

Proof. This can be shown by a reduction from problem MAX-3SAT-ODD, which asks whether $\max_{\sigma} |\{x_i \in X \mid \sigma(x_i) = \mathit{true}\}|$, where σ ranges over the satisfying assignments for a given 3SAT instance φ over X, is odd (see e.g. [10]). Using further variables y_i, whether $|\{x_i \in X \mid \sigma(x_i) = \mathit{true}\}|$ is odd can be expressed as y_n, where $y_1 \equiv x_1$ and $y_i \equiv \neg(y_{i-1} \equiv x_i)$, written in clausal form. Then, apply the reduction as in the proof of Lemma 4.4, and let x_i be as there and x_j be y_n. ■

Construct for the problem in Lemma 4.9 the instance of P2 as in the proof of Theorem 4.3 for the problem in Lemma 4.4 (where $x_i = x_1$). Then, add for each $i = 1, \ldots, k$ (recall that $X' = \{x_1, \ldots, x_k\}$) the equation

$$x_{n+1} + x_{n+i} - w_i = w_i \tag{6}$$

similar to Equation 3, where w_j is a new component, and drop for x_j from Lemma 4.9 Equation (4). These changes double the cost of components that have the same value as x_1, and add an extra cost for $x_1 = x_j$. It holds that $z_j = 0$ in some shortest c-bounded nonzero eigenvector iff an nae-satisfying assignment as in Lemma 4.9 exists. This proves the result. ■

For the analysis of problems P3 and P4, the following lemma is helpful.

Lemma 4.10 *Given an NAE3SAT instance φ, computing the lexicographically first nae-satisfying assignment of φ, τ^*, is* $\mathrm{FP}^{\mathrm{NP}}$*-hard. This holds even if φ is known to be nae-satisfiable.*

Proof. Reduce the analogous problem for 3SAT, and using the transformation of a 3SAT instance φ to a NAE3SAT instance φ' described above Lemma 4.1. Then, order the variables arbitrarily but such that the order starts with $x_0, x_1, \ldots, x_n$. By Lemma 4.1, the lexicographic first nae-satisfying assignment of φ' corresponds to the lexicographic first assignment of φ. As follows from [8], computing the latter, as well as a given bit thereof, is $\mathrm{FP}^{\mathrm{NP}}$-hard. ■

Theorem 4.11 *Problem P4 is* $\mathrm{FP}^{\mathrm{NP}}$*-complete. This holds even if b is fixed to $c \geq 1$.*

Proof. As described in the proof of Theorem 4.7, computing the length ℓ of a shortest non-zero b-bounded eigenvector $\mathbf{x}$ is in $\mathrm{FP}^{\mathrm{NP}}$. Computing the lexicographically first eigenvector $\mathbf{x}^*$ of length ℓ can be done with a polynomial number of NP oracle calls, computing x_1^*, x_2^* etc. in order; for each component x_i only $O(\log b)$ many values need be considered in a binary search on $[-b, b]$, and deciding if a partial vector $x_1, \ldots, x_k$ can be completed to a b-bounded eigenvector having length ℓ is in NP.

To show hardness, reuse the reduction from the proof of Theorem 4.3 and set $X' = \emptyset$. Then, the shortest non-zero c-bounded eigenvectors correspond to nae-satisfying assignments. In particular, the lexicographic first c-bounded eigenvector (in which $x_0 = -c$) corresponds to the lexicographically first nae-satisfying assignment of φ (in which $x_0 =$ *false*). By Lemma 4.10, this proves the result. ■

Corollary 4.12 *Let $c \geq 1$ be fixed. Given an $n \times n$ matrix M, an integer eigenvalue λ of M, and integers i, z, deciding if $x_i = z$ for the lexicographically first maximal eigenvector $\mathbf{x}$ among those that are c-bounded is* P^{NP}*-complete.*

Proof. Membership in P^{NP} is immediate from Theorem 4.11. As follows from [8], deciding a given bit of the lexicographically first satisfying truth assignment of a SAT instance is P^{NP}-complete. Thus, deciding a given bit $\tau^*(x_i)$ of the truth assignment τ^* in Lemma 4.10 is P^{NP}-hard, and the result follows from the reduction in the proof of Theorem 4.11. ■

Problem P5 turns out to be the hardest among the problems that we consider here, and is complete for $\mathrm{NP}^{\mathrm{NP}}$. In the proof, we employ that checking the validity of certain quantified boolean formulas (QBFs), based on the notion of nae-satisfaction, is as hard as for the standard notion of satisfaction. An NAESAT instance φ on variables X can be seen as a QBF $\Phi = \exists X.\varphi$, where φ is viewed as a conjunction of its clauses and the quantifier $\exists$ ranges over all truth assignments to X. Φ is valid under nae-satisfaction (briefly, nae-valid) if φ is a Yes-instance. Accordingly, a QBF $\forall Y \exists X.\varphi$ where φ is in conjunctive normal form (CNF) is nae-valid if for every assignment σ to Y, there is an assignment τ to X such that the combined assignment $\sigma \cup \tau$ nae-satisfies φ. The following lemma is used in the proof of the next theorem.

Lemma 4.13 *Given a QBF $\Phi = \forall Y \exists X.\varphi$, where φ is in CNF, deciding whether it is nae-valid is co-*$\mathrm{NP}^{\mathrm{NP}}$*-complete, and hard even if φ is in 3CNF.*

Proof. Membership in co-NP$^{\mathrm{NP}}$ is easy: A guess for τ such that no extension of τ by σ does nae-satisfy φ can be checked with a call to an NP oracle (the check is in co-NP).

Hardness follows from a reduction of checking the validity of a QBF of the given form in the standard sense, which utilizes the reduction from 3SAT to NAE3SAT given by the transformation from Lemma 4.1. I.e., construct for φ the formula φ', and consider the QBF

$$\Psi = \forall Y \exists X \cup X^* \cup \{z_j, x_0, w_i\}.\varphi'.$$

This formula is nae-valid if and only if Φ is valid in the standard sense. ■

Theorem 4.14 *Problem P5 is* NP$^{\mathrm{NP}}$*-complete, for every fixed* $c \geq 1$.

Proof. The problem is in NP$^{\mathrm{NP}}$, as a guess for a $\preceq$-minimal c-bounded nonzero eigenvector $\mathbf{x}$ can be verified with a call to a NP oracle (deciding whether some $\mathbf{y} \preceq \mathbf{x}$ with $\mathbf{x} \neq \mathbf{y}$ exists is in NP).

Hardness is shown by using the following variant of the problem in Lemma 4.13. Suppose for each assignment σ to Y, some assignment τ to X exists in which $\tau(x_1) = \tau(x_2)$ such that $\sigma \cup \tau$ nae-satisfies φ; it is asked whether a τ exists such that $\sigma \cup \tau$ nae-satisfies φ and $\tau(x_1) \neq \tau(x_2)$. This variant of the problem is also co-NP$^{\mathrm{NP}}$-hard; indeed, before applying the reduction from 3SAT to NAE3SAT, replace each clause in φ by the clauses $C \vee x_1$ and $C \vee x_2$, where x_1 and x_2 are fresh variables and split the clauses using further fresh variables to 3CNF form.

We reduce this problem to the complement of P5. Let the input formula be $\Phi = \forall X_2 \exists X_1.\varphi$. For the formula φ, seen as an NAE3SAT instance on $X_1 \cup X_2 = \{x_1, \ldots, x_n\}$, set up the equations (1)–(3) as in the reduction described in the proof of Theorem 4.2. Furthermore, introduce for each $j \neq 2$, $1 \leq j \leq n$, a new component z_j and set up the Equation (4) as for $X' = X_1 \setminus \{x_2\} \cup X_2$, i.e.,

$$x_{n+1} - x_{n+j} - z_j \quad = \quad z_j \tag{7}$$

Then, the c-bounded nonzero eigenvectors correspond 1-1 to the nae-satisfying assignments of φ. By the assertion on φ, for each assignment σ to X_2, a corresponding eigenvector $\mathbf{v}$ exists such that $v_1 = v_2 = c$. It holds that $||\mathbf{v}||^2 = (m+n)c^2+2n$, and $v_{2n+2} = 1$; $\mathbf{v}$ is $\preceq$-minimal, if and only if there is no assignment τ' to X_1 such that $\tau(x_1) \neq \tau(x_2)$ and σ, τ' satisfies φ. Thus, it follows that Φ is a not a Yes-instance for the problem described above, if and only if there exists some $\preceq$-minimal nonzero c-bounded eigenvector $\mathbf{v}$ which satisfies $v_{2n+2} = 1$. This proves the result. ■

5 Discussion and Conclusion

The results that we have derived in the previous section may be profitably used to derive similar complexity results for related problems. As an example, we consider the problem of finding integer relationships between numbers [6]. Given a real vector $\mathbf{v}$, find a vector of integers $\mathbf{x}$ such that $\mathbf{v}{\cdot}\mathbf{x}^{\mathbf{T}} = 0$. If $\mathbf{v}$ is an integer vector, then the resulting Diophantine equation always has nonzero solutions. Finding a b-bounded nonzero $\mathbf{x}$ which satisfies this equation is intractable, however.

Theorem 5.1 *Given an integer vector* $\mathbf{v} = (v_1, \dots, v_n)$ *and* $b \geq 0$*, deciding whether there is a nonzero* b*-bounded vector* $\mathbf{x} \in \mathbb{Z}$ *such that* $\mathbf{v} \cdot \mathbf{x}^{\mathbf{T}} = 0$ *is* NP*-complete. Hardness holds for* b *fixed to any* $c \geq 1$.

Proof. Obviously, a proper $\mathbf{x}$ can be guessed an checked in polynomial time. For the hardness part, we reduce problem P_1 to this problem. Rewrite $M \cdot \mathbf{x}^{\mathbf{T}} = \lambda \mathbf{x}^{\mathbf{T}}$ as $M' \cdot \mathbf{x}^{\mathbf{T}} = \mathbf{0}$, where $M' = M - \lambda \cdot I$ (I is the identity matrix). Let $m = \max_{i,j} |m'_{i,j}|$ be the largest absolute value in M'. Define $D = b \cdot n \cdot m + 1$, and let the vector $\mathbf{v} = (v_1, \dots, v_n)$ be

$$v_j = \sum_{i=1}^{n} D^{i-1} m'_{i,j}$$

(thus $\mathbf{v} \cdot \mathbf{x}^{\mathbf{T}} = \sum_{i=1}^{n} D^{i-1} \mathbf{y}_i \mathbf{x}^{\mathbf{T}}$, where $\mathbf{y}_i$ is the i-th row of M). Then, for every b-bounded vector $\mathbf{x}$ it holds that $\mathbf{v} \cdot \mathbf{x}^{\mathbf{T}} = 0$ iff $M' \cdot \mathbf{x}^{\mathbf{T}} = \mathbf{0}$. Observe that each D^i, $i \leq n$, has $O(i \cdot b \cdot n \cdot m)$ bits; thus, $\mathbf{v}$ is constructible in polynomial time from M', and thus from M and λ. This proves the result. ■

By the same reduction, similar complexity results as for problems P2-P5 can be established for analogous problems on a single Diophantine equation $\mathbf{v} \cdot \mathbf{x}^{\mathbf{T}} = 0$.

In this paper, we have considered the computational difficulty of problems that arise in the context of computing bounded integer eigenvectors for a given integer matrix M and eigenvalue λ.

As we have shown, computing some maximal b-bounded eigenvector is possible in polynomial time with the help of an NP oracle. Thus, practically speaking, this problem is not much harder than solving SAT. On the other hand, the proof of Theorem 4.3 suggests that parallelizing the computation of a maximal c-bounded eigenvector to NP problems is not evident; this follows from a similar property of computing a satisfying truth assignment (resp., an nae-satisfying truth assignment) of a SAT instance.

Some problems remain for further investigation. Other norms apart from L_2 for maximal vectors might be considered, as well as other domains such as the rationals, finite fields, or prime ideals of $\mathbb{Z}$. A further issue is approximation of shortest eigenvectors. This is left for future research.

Acknowledgment

This work has been partially supported by the Austrian Science Fund Project N Z29-INF.

References

[1] Z.-Z. Chen and S. Toda. The Complexity of Selecting Maximal Solutions. *Information and Computation*, 119:231–239, 1995. Extended Abstract in: *Proc. 8th IEEE Structure in Complexity Theory Conference*, pp. 313–325, 1993.

[2] J. Cullum and R. Willoughby *Lanczos Algorithms for Large Symmetric Eigenvalue Computations*, vol. 1. Birkhäuser, Boston, 1985.

[3] G. Del Corso and G. Manzini. On the Randomized Error of Polynomial Methods for Eigenvector and Eigenvalue Estimates. *Journal of Complexity*, 13:419–456, 1997.

[4] M. Garey and D. S. Johnson. *Computers and Intractability – A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, 1979.

[5] G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, Md, 1993.

[6] J. Håstad, B. Just, J. Lagarias, and C. Schnorr. Polynomial Time Relations for Finding Integer Relations Among Real Numbers. *SIAM Journal of Computing*, 18(5):859–881, 1989. Extended Abstract in: *Proc. STACS-86*, *LNCS* 210, pp. 105–118, 1986.

[7] D. S. Johnson. A Catalog of Complexity Classes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, vol. A. Elsevier, 1990.

[8] M. Krentel. The Complexity of Optimization Problems. *J. Computer and System Sciences*, 36:490–509, 1988.

[9] A. Lenstra, H. Lenstra, Jr., and L. Lovász. Factoring Polynomials with Rational Coefficients. *Mathematische Annalen*, 21:515–534, 1982.

[10] K. Wagner. Bounded Query Classes. *SIAM Journal of Computing*, 19(5):833–846, 1990.

[11] K. Yokoyama and T. Takeshima. On Hensel Construction of Eigenvalues and Eigenvectors of Matrices with Polynomial Entries. In *Proc. ACM-ISSAC '93, Kiev, Ukraine* (1993), ACM, pp. 218–224.