MATHEMATICAL ENGINEERING TECHNICAL REPORTS

Sparse Linear Complementarity Problems

Hanna SUMITA, Naonori KAKIMURA, and
Kazuhisa MAKINO

METR 2013–02 January 2013

DEPARTMENT OF MATHEMATICAL INFORMATICS
GRADUATE SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY
THE UNIVERSITY OF TOKYO
BUNKYO-KU, TOKYO 113-8656, JAPAN

WWW page: http://www.keisu.t.u-tokyo.ac.jp/research/techrep/index.html

The METR technical reports are published as a means to ensure timely dissemination of scholarly and technical work on a non-commercial basis. Copyright and all rights therein are maintained by the authors or by other copyright holders, notwithstanding that they have offered their works here electronically. It is understood that all persons copying this information will adhere to the terms and constraints invoked by each author's copyright. These works may not be reposted without the explicit permission of the copyright holder.

Sparse Linear Complementarity Problems

Hanna SUMITA* Naonori KAKIMURA† Kazuhisa MAKINO*

January 2013

Abstract

In this paper, we study the sparse linear complementarity problem, denoted by k-LCP: the coefficient matrix has at most k nonzero entries per row. It is known that 1-LCP is solvable in linear time, while 3-LCP is strongly NP-hard. We show that 2-LCP is strongly NP-hard, while it can be solved in $\mathrm{O}(n^3 \log n)$ time if it is sign-balanced, i.e., each row has at most one positive and one negative entries, where n is the number of constraints. Our second result matches with the currently best known complexity bound for the corresponding sparse linear feasibility problem. In addition, we show that an integer variant of sign-balanced 2-LCP is weakly NP-hard and pseudo-polynomially solvable, and the generalized 1-LCP is strongly NP-hard.

1 Introduction

Given a matrix $M \in \mathbb{R}^{n\times n}$ and a vector $q \in \mathbb{R}^n$, the *linear complementarity problem* (LCP) is to find vectors $w, z \in \mathbb{R}^n$ such that

$$w - Mz = q, \quad w, z \geq 0, \quad w^\top z = 0. \tag{1}$$

We denote a problem instance of LCP with M, q by LCP(M, q). We say that n is the *order* of LCP(M, q), where we note that the size of LCP(M, q) is $\mathrm{O}(n^2)$. The LCP, introduced by Cottle [10], Cottle and Dantzig [11], and Lemke [25], is one of the most widely studied mathematical programming problems, which, for example, contains linear and convex quadratic programming problems. Deciding whether LCP(M, q) has a solution for an arbitrary matrix M is NP-complete [7]. However, there are several classes of matrices M for which the associated LCP can be solved in polynomial time: for instance, positive semidefinite matrices [22], and *Z-matrices* (all off-diagonal entries are nonpositive) [3, 14, 26]. It is also known that M is a

*Department of Mathematical Informatics, University of Tokyo, Tokyo 113-8656, Japan. {Hanna_Sumita, makino}@mist.i.u-tokyo.ac.jp

†College of Arts and Sciences, University of Tokyo, Tokyo 153-8902, Japan. kakimura@global.c.u-tokyo.ac.jp

P-matrix, in which principal minors are all positive, if and only if LCP(M, q) has a unique solution for every q [29]. For details of theory of LCPs, see the books of Cottle, Pang, and Stone [13] and Murty [27].

In this paper, we focus on LCP with sparse coefficient matrix M. We denote by k-LCP the LCP whose coefficient matrix has at most k nonzero entries per row. For example, 2-LCP can have the following matrices:

$$M_1 = \begin{pmatrix} 0 & -1 & 3 \\ 0 & 1 & 1 \\ -2 & 0 & 0 \end{pmatrix},\ M_2 = \begin{pmatrix} 2 & -1 & 0 \\ 0 & 3 & 0 \\ 5 & 0 & -4 \end{pmatrix}.$$

Remark that the general LCP can be reduced to 3-LCP by introducing new variables, where the proof can be found in Appendix A.

Sparse LCP appears in the context of game theory. For example, mean payoff games can be formulated as 3-LCP [2]. Moreover, bimatrix games, which can be formulated as LCP, has been investigated in terms of sparsity in algorithmic game theory. A bimatrix game is *k-sparse* if each column and row in both payoff matrices of the game have at most k nonzero entries [6, 8, 16, 18].

Sparsity has also been attracting attention for the feasibility problem of systems of linear inequalities. A system of linear inequalities, i.e., a system of the form $Ax \le b$, where $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$, can be reformulated as a system of linear inequalities where each inequality involves at most three variables. If each inequality involves at most two variables, it is called a *TVPI system*[1]. A TVPI system can be naturally represented as a graph which has a vertex for each variable and an edge for each inequality, where an edge connects the vertices corresponding to the variables involved by the inequality. Shostak [31] proved that feasibility of a TVPI system can be decided by following paths and cycles in such a graph. This idea was used to design the first polynomial-time algorithm [1]. Cohen and Megiddo [9] and Hochbaum and Naor [20] proposed improved algorithms which run in $\mathrm{O}(mn^2(\log m + \log^2 n))$ time and $\mathrm{O}(mn^2 \log m)$ time, respectively, where m and n denote the number of constraints and variables, respectively. Any TVPI system can further be transformed to a *sign-balanced* TVPI system, where the two nonzero coefficients in each inequality have opposite signs. A sign-balanced TVPI system is also called a *monotone* TVPI system.

We say that 2-LCP is *sign-balanced* if the coefficient matrix M has at most one positive and negative entries per row. The matrix M_2 above is such an example. We note that sign-balanced TVPI systems with nonnegativity constraints can be formulated as sign-balanced 2-LCP.

The first main result of this paper is to present a polynomial-time combinatorial algorithm for sign-balanced 2-LCP.

[1]TVPI stands for "two variables per inequality."

Theorem 1. *Sign-balanced 2-LCP of order n can be solved in $\mathrm{O}(n^3 \log n)$ time.*

We remark that the complexity of Theorem 1 matches with the currently best known bound, due to Hochbaum and Naor [20], for the feasibility problem of sign-balanced TVPI systems with nonnegativity constraints. This implies that in order to improve the complexity of Theorem 1, we need to have a faster algorithm for the feasibility problem of sign-balanced TVPI systems with nonnegativity constraints.

It should also be noted that Theorem 1 is not obtained from the results for the other well-known subclasses of LCP that focus on the sign pattern of M, such as *Z-LCP* (i.e., the coefficient matrix is restricted to be a Z-matrix) and sign-solvable LCP introduced by Kakimura [21].

On the other hand, it turns out that 2-LCP seems to be intractable.

Theorem 2. *2-LCP is NP-hard in the strong sense.*

Since 1-LCP can easily be solved in linear time, Theorems 1 and 2 completely reveal computational complexity of the LCP in terms of sparsity. Note that 3-LCP is clearly NP-hard, since LCP can be reduced to 3-LCP in polynomial time. The first row in Table 1 summarizes our results for LCP.

Toward proving Theorem 1, we first design a simple combinatorial algorithm for sign-balanced 2-LCP. For a given instance of sign-balanced 2-LCP(M, q), consider the TVPI system S obtained by dropping the complementarity condition in (1). The algorithm computes the least element of S to find one of given constraints that needs to be satisfied with equality. By repeating this at most $n + 1$ times, we can find in polynomial time a solution of the instance or conclude that it is infeasible. To reduce the running time, we exploit deep results for sign-balanced TVPI systems. Cohen and Megiddo [9] presented an efficient procedure to decide whether a given feasible TVPI system is still feasible by adding new upper and lower bounds for each variable. We apply the procedure to S, which is not necessarily feasible, to find a constraint that needs to be satisfied with equality. Note that the obtained result by applying the Cohen–Megiddo's procedure might be wrong, since we might apply it to an infeasible system S. Thus after finishing all the iterations, we check if the obtained result is correct or not.

The LCP is said to be *unit* if the coefficient matrix M is restricted to belong to $\{0, \pm 1\}^{n \times n}$.

Theorem 3. *Unit sign-balanced 2-LCP of order n can be solved in $\mathrm{O}(n^2 \log n)$ time.*

The result is based on the framework of the simple algorithm for sign-balanced 2-LCP, in which we compute the least element by reduction to the shortest-path problem.

In addition, we discuss an integer variant of LCP. Given a matrix $M \in \mathbb{Z}^{n \times n}$, a vector $q \in \mathbb{Z}^n$, and a positive integer d, the *integer LCP* is the problem to find two integer vectors w, z satisfying (1) with $z \in \{0, 1, \ldots, d-1\}^n$. Integer LCP was first considered by Du Val [17] and Chandrasekaran [4] in the context of least element theory. Chandrasekaran, Kabadi and Sridhar [5] and Cunningham and Geelen [15] independently proposed sufficient conditions on a matrix M such that for every q, LCP(M, q) has an integer solution.

In this paper, we obtain the following result on integer sparse LCP. See also Table 1.

Theorem 4. *Integer sign-balanced 2-LCP is weakly NP-hard, and can be solved in pseudo-polynomial time.*

The weak NP-hardness follows from the fact that finding an integer solution to a sign-balanced TVPI system is weakly NP-hard [24]. The algorithm in Theorem 4 has a similar framework to the algorithm in Theorem 1. We here need to find the least element of integer solutions in the sign-balanced TVPI systems obtained from a given LCP instance, which can be done in pseudo-polynomial time [19, 20]. Note that the proof of Theorem 2 immediately implies that integer 2-LCP is NP-hard in the strong sense.

Finally, we investigate a generalization of LCPs in terms of sparsity. The *generalized* LCP (GLCP), which was introduced by Cottle and Dantzig [12], is a generalization of LCP from a square coefficient matrix to a vertical rectangular one.

Theorem 5. 1-*GLCP* (*i.e., the GLCP whose coefficient matrix that has at most one nonzero entry per row*) *is NP-hard in the strong sense.*

Table 1: Computational complexity of k-LCPs.

k	1	sign-balanced 2	$2 \leq$
LCP	$\mathrm{O}(n)$	$\mathbf{O(n^3 \log n)}$	**NP-hard**
integer LCP	$\mathrm{O}(n)$	**pseudo-polynomial**	**NP-hard**
GLCP		**NP-hard**	

This paper is organized as follows. Section 2 describes existing results of sign-balanced TVPI systems. Section 3 proposes a simple polynomial-time algorithm for sign-balanced 2-LCP. Section 4 improves the algorithm in Section 3 using the Cohen–Megiddo's procedure. Section 5 analyses the running time bound of Algorithm 1 for unit sign-balanced 2-LCP. Section 6 shows the NP-hardness of 2-LCP. Section 7 discusses computational complexity of integer 2-LCPs. Section 8 shows the NP-hardness of 1-GLCP.

2 Sign-balanced TVPI systems

Let F be the feasible region of a sign-balanced TVPI system. It is well known that if $x, y \in F$ then the *meet* $z = x \wedge y$ is contained in F, where $(x \wedge y)_i = \min(x_i, y_i)$. Indeed, for each inequality $a_j x_j + a_k x_k \geq b_i$ $(a_j > 0,\ a_k < 0)$, we may assume that $z_j = y_j$, and we have $a_j z_j + a_k z_k \geq a_j y_j + a_k y_k \geq b_i$, which implies $z \in F$. If F is bounded below, then there is a vector $u \in F$ such that for any $z \in F$, we have $z \geq u$. Such a vector is called the *least element* of F. Moreover, $F \cap \mathbb{Z}^n$ also has these properties.

The remaining of this section is organized as follows. In Section 2.1, we present Shostak's characterization of infeasibility of a TVPI system by a graph. The characterization is used by Cohen and Megiddo to design a combinatorial algorithm for solving TVPI systems. Their algorithm decides $\mathrm{O}(n(\log^2 n + \log m))$ times whether a given feasible TVPI system is still feasible by adding a bound for one variable, where m and n are the number of constraints and variables, respectively. Cohen and Megiddo presented a procedure (Algorithm 2.18 in [9]) which can be used to solve a more general decision problem based on Shostak's characterization, which runs in $\mathrm{O}(n^2)$ time. We describe the procedure in Section 2.2, which will be used in Section 4.

2.1 Characterization by a graph

Shostak [31] introduced a representation of a TVPI system by a graph and gave a characterization of infeasibility of the system in terms of the graph. The characterization is a generalization of a negative cycle in the shortest-path problem.

Let S be a TVPI system over variables $x_1, \ldots, x_n$. Shostak [31] represented S as an undirected graph $G = (V, E)$ as follows. For each variable x_i, the graph G has the vertex v_i. Moreover, G has an additional vertex v_0. For each inequality $a x_j + b x_k \leq c$ $(a, b \neq 0)$, the graph G has the edge $\{v_j, v_k\}$. For each single-variable inequality $x_i \geq \alpha$ (or $x_i \leq \beta$), the graph G has the edge $\{v_i, v_0\}$. For notational convenience, we introduce a new variable x_0 corresponding to v_0, and regard $x_i \geq \alpha$ (resp., $x_i \leq \beta$) as $x_i + b x_0 \geq \alpha$ (resp., $x_i + b x_0 \leq \beta$) with $b = 0$.

Let $P = (e_1, \ldots, e_l)$ be a path in G, where $e_i = \{v_{p_i}, v_{p_{i+1}}\}$ represents an inequality $a_i x_{p_i} + b_i x_{p_{i+1}} \leq c_i$ for $i = 1, \ldots, l$. If b_i and a_{i+1} have opposite signs for $i = 1, \ldots, l-1$, that is, one is positive and the other is negative, then P is said to be *admissible*. Note that the reverse of an admissible path is also admissible, and v_0 cannot be an intermediate vertex of an admissible path. An admissible path P *induces* a new inequality $a_P x_{p_1} + b_P x_{p_{l+1}} \leq c_P$ by eliminating common variables $x_{p_2}, \ldots, x_{p_l}$. For example, two inequalities $a_i x_{p_i} + b_i x_{p_{i+1}} \leq c_i$ and $a_{i+1} x_{p_{i+1}} + b_{i+1} x_{p_{i+2}} \leq c_{i+1}$ imply $a_i |a_{i+1}| x_{p_i} + b_{i+1} |b_i| x_{p_{i+2}} \leq c_i |a_{i+1}| + c_{i+1} |b_i|$. Any feasible solution to S satisfies all new inequalities induced by admissible paths in G.

A path is called a *loop* if the initial and last vertices are identical. An admissible loop L with initial vertex v_{p_1} induces a single-variable inequality $(a_L + b_L)x_{p_1} \leq c_L$. Note that if $v_{p_1} = v_0$, then $a_L = b_L = 0$ holds. We define the *extended graph* $\bar{G}$ of G by adding for each simple admissible loop L in G with initial vertex $v_i\,(v_i \neq v_0)$, a new edge which represents the single-variable inequality induced by L. If $\bar{G}$ has an admissible loop L that induces a new inequality $(a_L+b_L)x_i \leq c_L$ such that $a_L+b_L = 0$ and $c_L < 0$, then the loop L is called *infeasible*, in the sense that there is no vector satisfying the new inequality. Shostak showed infeasibility of S is equivalent to existence of a simple infeasible loop in $\bar{G}$.

Theorem 6 ([31])**.** *A TVPI system S is feasible if and only if the extended graph $\bar{G}$ has no simple infeasible loop.*

2.2 Cohen–Megiddo's procedure

In this subsection, we present a procedure of Cohen and Megiddo, which corresponds to Algorithm 2.18 in [9].

Let S be a feasible TVPI system, which may contain a single-variable linear inequality. By Theorem 6, $\bar{G}$ has no simple infeasible loop. Let T be a set of single-variable inequalities, and G_T be the graph associated with $S \cup T$. Theorem 6 implies that infeasibility of $S \cup T$ is equivalent to existence of a simple infeasible loop L in the extended graph $\bar{G}_T$ of G_T. Since $\bar{G}$ has no simple infeasible loop, L contains the vertex v_0, and at least one of the two edges incident to v_0 is an edge of T. Let $T' \subseteq T$ be the set of single-variable inequalities corresponding to the one or two edges. Then $|T'| \leq 2$ and $S \cup T'$ is infeasible by definition.

Given a feasible TVPI system S and a set T of single-variable inequalities, the Cohen–Megiddo's procedure decides whether $S \cup T$ is feasible or not, by detecting a simple infeasible loop in $\bar{G}_T$ if exists. By above discussion, the procedure can be equivalently written as follows:

Cohen–Megiddo's procedure

Input: a feasible TVPI system S and a set T of single-variable linear inequalities.

Output: find a nonempty set $T' \subseteq T$ such that $|T'| \leq 2$ and $S \cup T'$ is infeasible, or return that $S \cup T$ is feasible.

In particular, when S is a feasible sign-balanced TVPI system, the output T' of the Cohen–Megiddo's procedure has at most one upper and lower bounds. This is implicitly shown in [9], but we give a proof for correctness.

Lemma 1. *Let S be a feasible sign-balanced TVPI system. Let T be a set of single-variable linear inequalities such that $S \cup T$ is infeasible. Then the* *output $T' \subseteq T$ of the Cohen–Megiddo's procedure contains at most one upper and lower bounds.*

Proof. Let L be a simple infeasible loop with initial vertex v_0 and e_1, e_2 be edges of L incident to v_0. Let P be the path obtained by $L \setminus \{e_1, e_2\}$. Since S is sign-balanced, the inequality induced by P has one positive and negative coefficients. Since L is admissible, this means that either e_1 or e_2 represents an upper bound and the other represents a lower bound. At least one of e_1 and e_2 is due to T, and hence the statement holds. □

Cohen and Megiddo achieved the following running time.

Theorem 7 ([9])**.** *Let S be a feasible TVPI system with m inequalities and n variables. The Cohen–Megiddo's procedure terminates in* $\mathrm{O}(mn)$ *time.*

3 Simple algorithm for sign-balanced 2-LCP

In this section, we present an $\mathrm{O}(n^4 \log n)$ time algorithm for sign-balanced 2-LCP. The main idea of our algorithm is reduction of the problem to a sign-balanced TVPI system.

LCP(M, q) can be regarded as a problem to find a vector z satisfying $z^\top(Mz + q) = 0$ in $F := \{z \mid Mz + q \geq 0,\ z \geq 0\}$. Once we obtain such a vector z, the pair (w, z), where $w = Mz + q$, is a solution to LCP(M, q). We denote $SOL(M, q) := \{z \mid Mz + q \geq 0,\ z \geq 0,\ z^\top(Mz + q) = 0\}$.

If $F = \emptyset$, then LCP(M, q) has no solution. Suppose that $F \neq \emptyset$. Since F is the feasible region of a sign-balanced TVPI system bounded below, F has the least element u. If u satisfies $u^\top(Mu+q) = 0$, then u is clearly a solution to LCP(M, q). Otherwise, i.e., if u does not satisfy $u^\top(Mu + q) = 0$, then there is an index $i \in [n] := \{1, \dots, n\}$ such that $u_i > 0$ and $(Mu + q)_i > 0$. This implies that any $z \in SOL(M, q)$ satisfies $z_i \geq u_i > 0$, and hence z satisfies $(Mz + q)_i = 0$. Thus $SOL(M, q) \subseteq (F \cap \{z \mid (Mz + q)_i \leq 0\})$, which means that we can restrict F with a constraint $(Mz+q)_i \leq 0$, that is, replace F by $F \cap \{z \mid (Mz+q)_i \leq 0\}$. Since the inequality $(Mz+q)_i \leq 0$ has at most one positive and negative coefficients, F is still the feasible region of a sign-balanced TVPI system. Moreover, any $z \in F$ satisfies $z_i(Mz+q)_i = 0$.

We repeat the procedure mentioned above until the least element of F satisfies $z^\top(Mz + q) = 0$ or F turns out to be empty, i.e., LCP(M, q) is infeasible. Consequently, sign-balanced 2-LCP(M, q) is solved. Note that the number of repetition is at most $n + 1$.

The algorithm is summarized as follows.

Algorithm 1

Step 1. $F := \{z \mid Mz + q \geq 0, z \geq 0\}$.

Step 2. For $j = 0, \ldots, n$

Find the least element u of F. If u does not exist, then return that $\mathrm{LCP}(M, q)$ is infeasible.

If u satisfies $u^\top(Mu+q) = 0$, then return u. If u does not satisfy $u^\top(Mu + q) = 0$, then find an index i such that $u_i > 0$ and $(Mu + q)_i > 0$, update $F \leftarrow \{z \in F \mid (Mz + q)_i \leq 0\}$, and go to the next iteration.

It remains to discuss how to find the least element of F at Step 2. The least element of F is obtained by solving the linear programming problem $\min\{\mathbf{1}^\top z \mid z \in F\}$, where $\mathbf{1}$ is the vector whose elements are all one. Since Algorithm 1 requires to find the least element at most $n+1$ times, Algorithm 1 can find a solution to $\mathrm{LCP}(M, q)$ in polynomial time if exists.

The least element can be found more efficiently in a combinatorial way. Hochbaum and Naor [20] noted that their algorithm for TVPI systems can compute the least element of a sign-balanced TVPI system. Their algorithm runs in $\mathrm{O}(n^3 \log n)$ time, where n is the order of a given LCP instance, and hence the running time of Algorithm 1 reduces to $\mathrm{O}(n^4 \log n)$ time in total.

Theorem 8. *Algorithm 1 solves sign-balanced 2-LCP of order n in $\mathrm{O}(n^4 \log n)$ time.*

For example, consider $\mathrm{LCP}(M, q)$, where

$$M = \begin{pmatrix} -1 & 1 \\ -2 & 1 \end{pmatrix}, \quad q = \begin{pmatrix} -2 \\ 3 \end{pmatrix}.$$

The least element of F is $u = (0\ 2)^\top$, which does not satisfy $z^\top(Mz+q) = 0$ since $Mu + q = (0\ 5)^\top$. In this case, we have $u_2 > 0$ and $(Mu + q)_2 > 0$. We update F to $F \cap \{z \mid -2z_1 + z_2 + 3 \leq 0\}$. Then the least element shifts to $u' = (5\ 7)^\top$. Since $Mu' + q = 0$, we have $u' \in \mathit{SOL}(M, q)$.

4 Improved algorithm for sign-balanced 2-LCP

Recall that the main step of Algorithm 1 is finding the least element of the feasible region of a sign-balanced TVPI system S to detect an index i such that $z_i > 0$ for any solution z of a given $\mathrm{LCP}(M, q)$. In our improved algorithm, we directly detect such an index i without finding the least element.

For that purpose, we execute the Cohen–Megiddo's procedure described in Section 2.2 by setting T to be a set of single-variable inequalities in the form of $z_i \leq 0$. However, the procedure is guaranteed to run correctly only for feasible TVPI systems, while S is not necessarily feasible in our algorithm. In fact, the procedure may return that $S \cup T$ is "feasible" when

S is infeasible. For example, let S be the TVPI system consisting of the following eight constraints:

$$\begin{array}{lll} e_1 : z_1 - 2z_2 \le -1, & e_2 : 2z_2 - z_3 \le 3, & e_3 : z_3 - z_1 \le -3, \\ e_4 : -z_1 + 2z_2 \le 1, & e_5 : -2z_2 + z_3 \le -3, & \\ e_6 : z_1 \ge 0, & e_7 : z_2 \ge 0, & e_8 : z_3 \ge 0, \end{array}$$

and $T = \{e_9 : z_3 \le 0\}$. Let G be the graph associated with S as described in Section 2.1, whose edges are $e_1, \dots, e_8$. The simple admissible loop (e_1, e_2, e_3) induces $0 \le -1$, which implies infeasibility of S by Theorem 6. However, the extended graph $\bar{G}_T$ of $S \cup T$, which coincides with the graph G_T associated with $S \cup T$, has only two simple admissible loops with initial vertex v_0, namely, (e_6, e_1, e_2, e_9) and (e_7, e_2, e_9). Since neither of the two loops is infeasible, the Cohen–Megiddo's procedure decides that $S \cup T$ is "feasible," which is a contradiction.

Nevertheless, if the Cohen–Megiddo's procedure finds a nonempty subset T' of size at most two, then the system $S \cup T'$ is known to be infeasible without regard to feasibility of S. Moreover, we will show in Lemma 2 below that T' has the form $\{z_i \le 0\}$, which corresponds to an index i such that $z_i > 0$ for any solution z to LCP(M, q). Then, in a similar way to Algorithm 1, we can add a new constraint $(Mz + q)_i \le 0$ to S by complementarity, and repeat this until the Cohen–Megiddo's procedure returns that $S \cup T$ is "feasible." During and at the end of the repetition, we do not require that the sign-balanced TVPI system S is feasible. Instead, we need to solve a sign-balanced TVPI system at the last step in order to verify the feasibility of S.

A formal description of our algorithm is given as follows. For a set $I \subseteq [n]$, let $\bar{I} := [n] \setminus I$. We denote by z_I a subvector of z which consists of entries with coordinates in $I \subseteq [n]$.

Algorithm 2

Step 1. $I := [n],\ F := \{z \mid Mz + q \ge 0, z \ge 0\}$.

Step 2. While $I \ne \emptyset$, do Step 3.

Step 3. Let S be the sign-balanced TVPI system $Mz + q \ge 0, z \ge 0, (Mz + q)_{\bar{I}} \le 0$ and $T = \{z_i \le 0 \mid i \in I\}$.
Execute the Cohen–Megiddo's procedure with inputs S and T. If the procedure returns $T' = \{z_i \le 0\}$ (for some $i \in I$), then update $I \leftarrow I \setminus \{i\},\ F \leftarrow F \cap \{z \mid (Mz + q)_i \le 0\}$ and go to the next iteration. Otherwise, go to Step 4.

Step 4. Find a feasible vector of $F \cap \{z \mid z_I \le 0\}$, that is, solve

$$z_I = 0,\ z_{\bar{I}} \ge 0,\ (Mz + q)_I \ge 0,\ (Mz + q)_{\bar{I}} = 0. \qquad (2)$$

If a feasible vector z^* exists, then return z^*. Otherwise, return that LCP(M, q) is infeasible.

For correctness of Algorithm 2, we show the following lemma. Note that throughout Algorithm 2, F remains to be the feasible region of a sign-balanced TVPI system.

Lemma 2. *Let LCP(M, q) be a sign-balanced 2-LCP instance. LCP(M, q) has a solution if and only if the sign-balanced TVPI system* (2) *is feasible.*

Proof. The if-part is easy to see. It suffices to show the only-if-part. Suppose that LCP(M, q) has a solution. At Step 1, F is not empty because $SOL(M, q) \subseteq F$.

We will show that throughout the execution of Step 3, it holds that

1. any $z \in SOL(M, q)$ satisfies $z_i > 0$ for all $i \in \bar{I}$, and

2. $SOL(M, q) \subseteq F$.

These claims hold at the beginning, that is, when $I = [n]$. Suppose that the claims hold for some $I \subseteq [n]$. We may assume that the Cohen–Megiddo's procedure returns a nonempty subset $T' \subseteq T$. Since T contains only upper bounds, T' contains only one upper bound, that is, $z_i \leq 0$ for some $i \in I$, by nonemptiness of F and Lemma 1. Since $S \cup \{z_i \leq 0\}$ is infeasible, any $z' \in F$ satisfies $z'_i > 0$, and hence any $z \in SOL(M, q)$ satisfies $(Mz + q)_i = 0$ by $SOL(M, q) \subseteq F$. This implies that $SOL(M, q) \subseteq (F \cap \{z \mid (Mz + q)_i \leq 0\})$. Thus the claims hold.

Therefore, S always has a solution during Step 3 by the second claim. Hence, when we go to Step 4, $S \cup \{z_i \leq 0 \mid i \in I\}$ is feasible, because the Cohen–Megiddo's procedure works correctly. Thus (2) has a solution. □

We discuss the time complexity of Algorithm 2.

Theorem 9. *Algorithm 2 solves sign-balanced 2-LCP of order n in* $\mathrm{O}(n^3 \log n)$ *time.*

Proof. The number of repetitions in Step 3 is at most n since $|I|$ decreases by one at each repetition. The execution time of each repetition is $\mathrm{O}(n^2)$ time by Theorem 7. Therefore, Algorithm 2 takes $\mathrm{O}(n^3)$ time to go through Step 3. At Step 4, since the TVPI system (2) has at most $3n$ inequalities, it is solvable in $\mathrm{O}(n^3 \log n)$ time by the algorithm of Hochbaum and Naor [20]. This concludes the proof. □

Thus Theorem 1 immediately holds from Theorem 9.

Remark 1. *The running time of Algorithm 2 can be written as* $\mathrm{O}(n^3 + T_{\mathrm{LI}}(n, n))$ *time, where* $T_{\mathrm{LI}}(m, n)$ *denotes the time complexity for solving a TVPI system with m constraints and n variables. In other words, Algorithm 2 reduces sign-balanced 2-LCP to sign-balanced TVPI system in* $\mathrm{O}(n^3)$ *time.*

Remark 2. *A sign-balanced TVPI system $Ax \leq b, x \geq 0$ where $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^n$ can be formulated as a sign-balanced 2-LCP instance with*

$$M = \begin{array}{c} \\ m \\ n \end{array} \!\!\begin{array}{c} \begin{array}{cc} m & n \end{array} \\ \begin{pmatrix} 0 & -A \\ 0 & 0 \end{pmatrix} \end{array}, \quad q = \begin{pmatrix} b \\ 0 \end{pmatrix}.$$

This implies that sign-balanced 2-LCP cannot be solved faster than sign-balanced TVPI system with nonnegativity constraints, whose current best running time is $\mathrm{O}(n^3 \log n)$ [20] when $m = \mathrm{O}(n)$. Theorem 9 shows that Algorithm 2 achieves the same running time.

5 Unit sign-balanced 2-LCP

In this section, we discuss the time complexity of unit sign-balanced 2-LCP. Recall that $\mathrm{LCP}(M, q)$ is called *unit* if $M \in \{0, \pm 1\}^{n\times n}$. We will show that the running time of Algorithm 1 reduces to $\mathrm{O}(n^2 \log n)$ time.

Let $F = \{z \mid Mz + q \geq 0, z \geq 0\}$. The least element of F is the unique optimal solution of the linear programming problem

$$\begin{array}{ll} \text{min.} & \mathbf{1}^\top z \\ \text{s.t.} & Mz + q \geq 0, \\ & z \geq 0, \end{array} \tag{3}$$

where recall that $\mathbf{1}$ is the vector whose elements are all one.

If $M \in \{0, \pm 1\}^{n\times n}$, then the linear programming problem (3) can be solved more efficiently. First, we introduce a new variable z_0, and rewrite every single-variable constraint in (3) as follows:

- if $z_i \leq \alpha$, then $-z_i + z_0 \geq -\alpha$,
- if $z_i \geq \alpha$, then $z_i - z_0 \geq \alpha$.

Let D be a directed graph which has a vertex v_i for each variable z_i, and an edge (v_j, v_i) with length $-\alpha$ for each inequality $z_i - z_j \geq \alpha$.

Then, the dual of (3) is the shortest-path problem from a vertex v_0 to all vertices on the directed graph D. Let $d \in \mathbb{R}^{n+1}$ be the vector where d_i is the shortest distance from v_0 to v_i on D. Note that $d_0 = 0$. The vector $(-d_1 \ \cdots \ -d_n)^\top \in \mathbb{R}^n$ is the optimal solution to the original problem (3). Hence, by solving the shortest-path problem using the Bellman–Ford algorithm, we can find the least element of F in $\mathrm{O}(n^2)$ time.

Algorithm 1 finds the least element of F every time F is restricted to $F \cap \{z \mid (Mz + q)_i \leq 0\}$ for some $i \in [n]$. We can construct the least element of $F \cap \{z \mid (Mz + q)_i \leq 0\}$ from that of F in $\mathrm{O}(n \log n)$ time by an algorithm of Ramalingam, Song, Joskowicz and Miller [28]. If we apply their algorithm to find the least element on and after the second iteration at Step 2 in Algorithm 1, then unit sign-balanced 2-LCP can be solved more efficiently.

Theorem 10. *Algorithm 1 solves unit sign-balanced 2-LCP of order n in* $\mathrm{O}(n^2 \log n)$ *time.*

This means that Theorem 3 holds.

6 NP-hardness for 2-LCP

In this section, we prove Theorem 2, which says that 2-LCP is NP-hard. This is contrast to the fact that a TVPI system can be solved in polynomial time even if the system is not sign-balanced. The NP-hardness can be proved by reduction of monotone one-in-three 3SAT to 2-LCP.

Given a monotone 3CNF formula $\psi = \bigwedge_{j=1}^{m}(x_{j_1} \vee x_{j_2} \vee x_{j_3})$ with n literals, the *monotone one-in-three 3SAT* is a problem to decide whether there exists an assignment to $(x_1, \ldots, x_n)$ so that for each clause, exactly one literal is true. The monotone one-in-three 3SAT is introduced and proved to be NP-complete by Schaefer [30].

We now restate Theorem 2 and present the proof.

Theorem 2. *2-LCP is NP-hard in the strong sense.*

Proof. Let $\psi = \bigwedge_{j=1}^{m}(x_{j_1} \vee x_{j_2} \vee x_{j_3})$ be a monotone one-in-three 3SAT instance with n literals. We construct an instance of 2-LCP of order $n + 9m$ from ψ as follows: for each literal $i = 1, \ldots, n$, define

$$w_i + z_i = 1. \tag{4}$$

Moreover, for each clause $j = 1, \ldots, m$, letting $p_j = n + 9(j-1)$, set

$$w_{p_j+1} + z_{j_2} + z_{j_3} = 1,\ z_{j_1} + w_{p_j+2} + z_{j_3} = 1,\ z_{j_1} + z_{j_2} + w_{p_j+3} = 1, \tag{5}$$

and in addition, set

$$\begin{aligned} w_{p_j+4} - z_{p_j+1} - z_{j_1} &= -1, & w_{p_j+5} + z_{p_j+1} + z_{j_1} &= 1,\\ w_{p_j+6} - z_{p_j+2} - z_{j_2} &= -1, & w_{p_j+7} + z_{p_j+2} + z_{j_2} &= 1,\\ w_{p_j+8} - z_{p_j+3} - z_{j_3} &= -1, & w_{p_j+9} + z_{p_j+3} + z_{j_3} &= 1. \end{aligned} \tag{6}$$

Consider the instance of 2-LCP consisting of the above constraints (4), (5) and (6). Note that (6) is equivalent to

$$z_{p_j+1} + z_{j_1} = 1,\ z_{p_j+2} + z_{j_2} = 1,\ z_{p_j+3} + z_{j_3} = 1, \tag{7}$$

since $w_{p_j+\ell} \geq 0$ for $\ell = 4, 5, \ldots, 9$.

We denote by M and q the coefficient matrix and the constant vector of the above instance of LCP. We will show that LCP(M, q) has a solution if and only if the monotone one-in-three 3SAT instance ψ is a true instance.

First assume that LCP(M, q) has a solution (w, z). By (4), for any $i = 1, \ldots, n$, it holds that $(w_i, z_i) = (0, 1)$ or $(w_i, z_i) = (1, 0)$. Assign each literal x_i true if $z_i = 1$ and false otherwise.

We will claim that x is a truth assignment for ψ, that is, each clause has exactly one true literal. Indeed, for each clause j, if $z_{j_1} = 0$ then $w_{p_j+1} = 0$ by (7) and the complementarity, and hence exactly one of z_{j_2} and z_{j_3} is equal to one by the first equation in (5). If $z_{j_1} = 1$ then $z_{j_2} = z_{j_3} = 0$ by the second and third equations in (5). Thus each clause has exactly one true literal.

Conversely, assume that ψ is a true instance. Let $x = (x_1, \ldots, x_n)$ be a truth assignment of ψ. Define $z \in \mathbb{R}^{n+9m}$ as follows: For $i = 1, \ldots, n$, set $z_i = 1$ if x_i is true, and $z_i = 0$ if x_i is false. For $j = 1, \ldots, m$, set $z_{p_j+\ell} = 1 - z_{j_\ell}$ for $\ell = 1, 2, 3$ and $z_{p_j+\ell} = 0$ for $\ell = 4, \ldots, 9$. Define $w = Mz + q$. Then the pair (w, z) satisfies (4), (5) and (6), and $w, z \geq 0$ holds.

We claim that the pair (w, z) is a solution of LCP(M, q). To prove this, it remains to show that the pair (w, z) satisfies $w^\top z = 0$. For $i = 1, \ldots, n$, it clearly holds that $w_i z_i = 0$ by (4). Let $j \in \{1, \ldots, m\}$. Since the clause j has exactly one true literal, we may suppose by symmetry that $z_{j_1} = 1$ and $z_{j_2} = z_{j_3} = 0$. By (5), it holds that $w_{p_j+1} = 1$ and $w_{p_j+2} = w_{p_j+3} = 0$. On the other hand, we have $z_{p_j+1} = 0$ and $z_{p_j+2} = z_{p_j+3} = 1$ by (7), which means that $w_{p_j+\ell} z_{p_j+\ell} = 0$ for $\ell = 1, 2, 3$. For $\ell = 4, \ldots, 9$, we have $w_{p_j+\ell} z_{p_j+\ell} = 0$ since $z_{p_j+\ell} = 0$. Thus the complementarity condition is satisfied.

Therefore, LCP(M, q) has a solution if and only if ψ is a true instance, and thus the statement holds. □

7 Integer solutions of sparse LCP

In this section, we discuss the computational complexity of integer 2-LCP and prove Theorem 4. Recall that the integer LCP is the problem to find two integer vectors w, z satisfying (1) with $z \in \{0, 1, \ldots, d-1\}^n$. As well as the LCP, it is equivalent to the problem to find a vector $z \in \{0, \ldots, d-1\}^n$ satisfying the three constraints $Mz + q \geq 0, z \geq 0$ and $z^\top(Mz + q) = 0$.

We first show that integer sign-balanced 2-LCP is weakly NP-hard. This is proved by reduction from integer sign-balanced TVPI system, which is shown to be NP-hard by Lagarias [24]. In the proof, we use the fact that if the integer sign-balanced TVPI system $Ax \leq b,\ x \in \mathbb{Z}^n$ is feasible, then there exists a feasible vector with components of size at most $13(\mathrm{size}(A) + \mathrm{size}(b))$ [23], where size(A) and size(b) denote the sizes of A and b in the binary representation, respectively.

Lemma 3. *Integer sign-balanced 2-LCP is NP-hard.*

Proof. Let $Ax \leq b,\ x \in \mathbb{Z}^n$ be an integer sign-balanced TVPI system. Let $l \in \mathbb{Z}^n$ be the vector whose elements are all $-2^{13(\mathrm{size}(A)+\mathrm{size}(b))}$. We add lower bounds $x \geq l$. Then, by replacing $x - l$ with y, the integer sign-balanced TVPI system is rewritten as $Ay + Al - b \leq 0,\ y \geq 0,\ y \in \mathbb{Z}^n$, which can be formulated as an integer sign-balanced 2-LCP(M, q), where

$$M = \begin{pmatrix} 0 & -A \\ 0 & 0 \end{pmatrix}, \quad q = \begin{pmatrix} -Al + b \\ 0 \end{pmatrix}.$$

LCP(M, q) is polynomially equivalent to the integer sign-balanced TVPI system. □

We then obtain a pseudo-polynomial time algorithm for integer sign-balanced 2-LCP(M, q) based on the framework of Algorithm 1. Let $F = \{z \mid Mz + q \geq 0, z \geq 0, (Mz + q)_I \leq 0\}$ for some $I \subseteq [n]$. It is known that $F \cap \mathbb{Z}^n$ has the least element as mentioned in Section 2. Instead of finding the least element of F in Algorithm 1, we find the least element of $F \cap \mathbb{Z}^n$, which can be computed in $\mathrm{O}(nd)$ time [19] by using transformation to 2SAT. By repeating this at most $n + 1$ times, we can find a solution to LCP(M, q). Therefore, the following theorem holds.

Theorem 11. *Algorithm 1 solves integer sign-balanced 2-LCP of order n in $\mathrm{O}(n^2 d)$ time, where d is the upper bound of each component of a solution.*

Theorem 4 immediately holds by Lemma 3 and Theorem 11.

Note that it remains open to find an integer solution to sign-balanced 2-LCP, i.e., when we do not have upper bounds, in pseudo-polynomial time. It is known that it is still open even to find an integer solution for TVPI system without finite bounds in pseudo-polynomial time.

We conclude this section with finding an integer solution to unit sign-balanced 2-LCP. In this case, we use the same algorithm as unit sign-balanced 2-LCP. Let us review the linear programming problem (3) in Section 5, whose optimal solution is the least element of F. Since the vector q is integer, the shortest distance d is integer, and so is the optimal solution to (3). Therefore, the following corollary holds.

Corollary 1. *Algorithm 1 solves integer unit sign-balanced 2-LCP of order n in $\mathrm{O}(n^2 \log n)$ time.*

On the other hand, the proof of Theorem 2 implies that integer unit 2-LCP is NP-hard even if $d = 2$.

Corollary 2. *Integer unit 2-LCP is NP-hard.*

8 Extension to the generalized LCP

In this section, we discuss a well-studied generalization of LCPs and its computational complexity in terms of sparsity. One of generalized problems of the LCP is the generalized LCP (GLCP), introduced by Cottle and Dantzig [12].

We denote a matrix $N \in \mathbb{R}^{p\times n}$ and vectors $q, w \in \mathbb{R}^p$ of type $(p_1, \dots, p_n)$ by

$$N = \begin{pmatrix} N^1 \\ N^2 \\ \vdots \\ N^n \end{pmatrix} \in \mathbb{R}^{p\times n},\ q = \begin{pmatrix} q^1 \\ q^2 \\ \vdots \\ q^n \end{pmatrix},\ w = \begin{pmatrix} w^1 \\ w^2 \\ \vdots \\ w^n \end{pmatrix} \in \mathbb{R}^p,$$

where $p = \sum_{i=1}^n p_i$, $N^i \in \mathbb{R}^{p_i \times n}$ $(i = 1, \dots, n)$ and $q^i, w^i \in \mathbb{R}^{p_i}$ $(i = 1, \dots, n)$.

Given a matrix N and a vector q of type $(p_1, p_2, \dots, p_n)$, the GLCP is to find vectors $w \in \mathbb{R}^p, z \in \mathbb{R}^n$ such that

$$w - Nz = q, \quad w, z \geq 0, \quad z_i \prod_{j=1}^{p_i} w_j^i = 0\ (i = 1, \dots, n).$$

We similarly define *k-GLCP* by the GLCP whose coefficient matrix N has at most k nonzero entries per row.

We now prove Theorem 5 stated in the Introduction.

Theorem 5. *1-GLCP is NP-hard in the strong sense.*

Proof. We show this theorem by reduction from the 3SAT problem. Let $\psi = \bigwedge_{j=1}^m (y_{j_1} \vee y_{j_2} \vee y_{j_3})$, where $y_i \in \{x_i, \bar{x}_i\}$, be an instance of the 3SAT problem with n literals. We construct an instance of GLCP of order $(n + 5m) \times (n+2m)$ from ψ as follows. For each literal $i = 1, \dots, n$, the ith block is defined to be only one equation:

$$w^i + z_i = 1. \tag{8}$$

For each clause $j = 1, \dots, m$, let $k_j = n + 2(j-1)$. The $(k_j + 1)$th block is defined to have three equations:

$$w_\ell^{k_j+1} = \begin{cases} 1 - z_{j_\ell} & \text{if } y_{j_\ell} = x_{j_\ell} \\ z_{j_\ell} & \text{if } y_{j_\ell} = \bar{x}_{j_\ell} \end{cases} \quad (\ell = 1, 2, 3), \tag{9}$$

and the $(k_j + 2)$th block is to have two equations:

$$w_1^{k_j+2} - z_{k_j+1} = -1 \text{ and } w_2^{k_j+2} + z_{k_j+1} = 1. \tag{10}$$

Observe that, for $i = 1, \dots, n$, a pair (w, z) of nonnegative vectors satisfies (8) and $w^i z_i = 0$ if and only if $(w_i, z_i) = (0, 1)$ or $(w_i, z_i) = (1, 0)$.

For $j = 1, \ldots, m$, note that $z_{k_j+1} = 1$ holds by (10) and $w \geq 0$, and hence the complementarity condition with respect to $k_j + 1$ means that at least one of $w_1^{k_j+1}$, $w_2^{k_j+1}$, and $w_3^{k_j+1}$ is equal to zero. Hence, if we set x_i to be true if $z_i = 1$ and false if $z_i = 0$, then this is equivalent to that the clause j is satisfied by (9). Therefore, the GLCP has a solution if and only if ψ is satisfiable, which shows the statement. □

Acknowledgement

This work is partially supported by a Grant-in-Aid for Scientific Research from the Ministry of Education, Culture, Sports, Science and Technology of Japan, by the Global COE "The Research and Training Center for New Development in Mathematics," and by JST, ERATO, Kawarabayashi Large Graph Project.

References

[1] B. Aspvall and Y. Shiloach. A polynomial time algorithm for solving systems of linear inequalities with two variables per inequality. *SIAM Journal on Computing*, **9** (1980), pp. 827–845.

[2] H. Björklund, O. Svensson and S. Vorobyov. Linear complementarity algorithms for mean payoff games. *Technical Report 2005-05*, DIMACS: Center for Discrete Mathematics and Theoretical Computer Science, 2005.

[3] R. Chandrasekaran. A special case of the complementary pivot problem. *Opsearch*, **7** (1970), pp. 263–268.

[4] R. Chandrasekaran. Integer programming problems for which a simple rounding type algorithm works. *Combinatorial Optimization*, **8** (1984), pp. 101–106.

[5] R. Chandrasekaran, S. N. Kabadi and R. Sridhar. Integer solution for linear complementarity problem. *Mathematics of Operations Research*, **23** (1998), pp. 390–402.

[6] X, Chen, X. Deng and S. Teng. Sparse games are hard. *Proceedings of the 2nd International Workshop on Internet and Network Economics*, 2006, pp. 262–273.

[7] S. J. Chung. NP-completeness of the linear complementarity problem. *Journal of Optimization Theory and Applications*, **60** (1989), pp. 393–399.

[8] B. Codenotti, M. Leoncini and G. Resta. Efficient Computation of Nash equilibria for very sparse win-lose bimatrix games. *Proceedings of the 14th Annual European Symposium on Algorithms*, 2006, pp. 232–243.

[9] E. Cohen and N. Megiddo. Improved algorithms for linear inequalities with two variables per inequality. *SIAM Journal on Computing*, **23** (1994), pp. 1313–1347.

[10] R. W. Cottle. The principal pivoting method of quadratic programming. In: G. B. Dantzig, A. F. Veinott (Eds.), *Mathematics of Decision Sciences*, Part 1, American Mathematical Society, Providence R. I., 1968, pp. 142–162.

[11] R. W. Cottle and G. B. Dantzig. Complementary pivot theory of mathematical programming. *Linear Algebra and Its Applications*, **1** (1968), pp. 103–125.

[12] R. W. Cottle and G. B. Dantzig. A generalization of the linear complementarity problem. *Journal on Combinatorial Theory*, **8** (1970), pp. 79–90.

[13] R. W. Cottle, J. S. Pang and R. E. Stone. *The Linear Complementarity Problem.* Academic Press, Boston, 1992.

[14] R. W. Cottle and A. F. Veinott. Polyhedral sets having a least element. *Mathematical Programming*, **3** (1972), pp. 238–249.

[15] W. H. Cunningham and J. F. Geelen. Integral solutions of linear complementarity problems. *Mathematics of Operations Research*, **23** (1998), pp. 61–68.

[16] C. Daskalakis and C. H. Papadimitriou. On oblivious PTAS's for Nash equilibrium. *Proceedings of the 41st Annual ACM Symposium on Theory of computing*, 2009, pp. 75–84.

[17] P. Du Val. The unloading problem for plane curves. *American Journal of Mathematics*, **62** (1940), pp. 307–311.

[18] D. Hermelin, C. Huang, S. Kratsch and M. Wahlström. Parameterized two-player Nash equilibrium. *Algorithmica*, 2012, pp. 1–15.

[19] D. S. Hochbaum, N. Megiddo, J. Naor and A. Tamir. Tight bounds and 2-approximation algorithms for integer programs with two variables per inequality. *Mathematical Programming*, **62** (1993), pp. 69–83.

[20] D. S. Hochbaum and J. Naor. Simple and fast algorithms for linear and integer programs with two variables per inequality. *SIAM Journal on Computing*, **23** (1994), pp. 1179–1192.

[21] N. Kakimura. Sign-solvable linear complementarity problems. *Linear Algebra and Its Applications*, **429** (2008), pp. 606–616.

[22] M. Kojima, N. Megiddo, T. Noma and A. Yoshise. A unified approach to interior point algorithms for linear complementarity problems. *Lecture Notes in Computer Science*, **538**, Springer, Berlin, 1991.

[23] B. Korte and J. Vygen. *Combinatorial Optimization: Theory and Algorithms*. Fifth Edition, Springer-Verlag, Berlin, Heidelberg, 2012.

[24] J. C. Lagarias. The computational complexity of simultaneous Diophantine approximation problems. *SIAM Journal on Computing*, **14** (1985), pp. 196–209.

[25] C. E. Lemke. Bimatrix equilibrium points and mathematical programming. *Management Science*, **11** (1965), pp. 681–689.

[26] O. L. Mangasarian. Linear complementarity problems solvable by a single linear program. *Mathematical Programming*, **10** (1976), pp. 263–270.

[27] K. G. Murty. *Linear Complementarity, Linear and Nonlinear Programming*. Internet Edition, 1997.

[28] G. Ramalingam, J. Song, L. Joskowicz and R. E. Miller. Solving systems of difference constraints incrementally. *Algorithmica*, **23** (1999), pp. 261–275.

[29] H. Samelson, R. M. Thrall and O. Wesler. A partition theorem for Euclidean n-space. *Proceedings of the American Mathematical Society*, **9** (1958), pp. 805–807.

[30] T. J. Schaefer. The complexity of satisfiability problems. *Proceedings of the 10th Annual ACM Symposium on Theory of Computing*, 1978, pp. 216–226.

[31] R. Shostak. Deciding linear inequalities by computing loop residues. *Journal of the ACM*, **28** (1981), pp. 769–779.

A Reduction of LCP to 3-LCP

In this section, we will show that any instance of LCP can be rewritten as an instance of 3-LCP with polynomial size.

Let LCP(M, q) be an instance of k-LCP ($k \geq 4$) of order n. We will construct an equivalent instance LCP(M', q') that is an instance of k'-LCP, where $k' < k$. By repeating the construction until M' has at most three nonzero entries per row, we obtain an instance of 3-LCP.

Suppose that the ith equation $w_i - (Mz)_i = q_i$ is in the form of

$$w_i - (a_1 z_1 + \cdots + a_p z_p - a_{p+1} z_{p+1} - \cdots - a_{p+r} z_{p+r}) = q_i, \qquad (11)$$

where $a_j > 0$ for $j = 1, \dots, p+r$. We may assume that $p + r > 3$, since if $p + r \leq 3$, we do not transform this equation. We also assume that $p \geq r$. The argument for the case when $p < r$ is similar. We transform the system (11) and $w, z \geq 0$ as follows:

Case 1: If $r > 1$, introducing new twelve variables $z^i_1, \dots, z^i_6, w^i_1, \dots w^i_6$, set

$$w_i - (z^i_1 + z^i_2 - z^i_3) = q_i, \quad w, z \geq 0, \quad w^i, z^i \geq 0, \tag{12}$$

and

$$\begin{gathered} w^i_1 - (z^i_1 - \sum_{j=1}^{\lceil p/2 \rceil} a_j z_j) = 0, \quad w^i_2 - (-z^i_1 + \sum_{j=1}^{\lceil p/2 \rceil} a_j z_j) = 0, \\ w^i_3 - (z^i_2 - \sum_{j=\lceil p/2 \rceil+1}^{p} a_j z_j) = 0, \ w^i_4 - (-z^i_2 + \sum_{j=\lceil p/2 \rceil+1}^{p} a_j z_j) = 0, \\ w^i_5 - (z^i_3 - \sum_{j=k+1}^{k+l} a_j z_j) = 0, \quad w^i_6 - (-z^i_3 + \sum_{j=k+1}^{k+l} a_j z_j) = 0. \end{gathered} \tag{13}$$

Case 2: If $r = 1$, introducing new eight variables $z^i_1, \dots, z^i_4, w^i_1, \dots w^i_4$, set

$$w_i - (z^i_1 + z^i_2 - z_{p+1}) = q_i, \quad w, z \geq 0, \quad w^i, z^i \geq 0, \tag{14}$$

and

$$\begin{gathered} w^i_1 - (z^i_1 - \sum_{j=1}^{\lceil p/2 \rceil} a_j z_j) = 0, \quad w^i_2 - (-z^i_1 + \sum_{j=1}^{\lceil p/2 \rceil} a_j z_j) = 0, \\ w^i_3 - (z^i_2 - \sum_{j=\lceil p/2 \rceil+1}^{p} a_j z_j) = 0, \ w^i_4 - (-z^i_2 + \sum_{j=\lceil p/2 \rceil+1}^{p} a_j z_j) = 0. \end{gathered} \tag{15}$$

Case 3: If $r = 0$, introduce the same new variables as in Case 2 and set

$$w_i - (z^i_1 + z^i_2) = q_i, \quad w, z \geq 0, \quad w^i, z^i \geq 0, \tag{16}$$

and (15).

The new system is equivalent to the system (11) and $w, z \geq 0$. Note that since $w^i \geq 0$, the system (13) is equivalent to

$$z^i_1 = \sum_{j=1}^{\lceil p/2 \rceil} a_j z_j, \quad z^i_2 = \sum_{j=\lceil p/2 \rceil+1}^{p} a_j z_j, \quad z^i_3 = \sum_{j=k+1}^{k+l} a_j z_j,$$

and the system (15) is equivalent to

$$z_1^i = \sum_{j=1}^{\lceil p/2 \rceil} a_j z_j, \quad z_2^i = \sum_{j=\lceil p/2 \rceil + 1}^{p} a_j z_j.$$

By these equations and the constraint $z \geq 0$, the constraint $z^i \geq 0$ is redundant.

We transform $w_i - (Mz)_i = q_i,\ w, z \geq 0$ for $i = 1, \ldots, n$ in the way above. Let the resulting system be denoted by

$$w' - M'z' = q', \quad w', z' \geq 0. \tag{17}$$

We claim that LCP(M', q') is equivalent to LCP(M, q). Indeed, the system $w - Mz = q,\ w, z \geq 0$ is equivalent to the system (17). Moreover, any feasible solution to the system (17) satisfies $w^i = 0$ for $i = 1, \ldots, n$. Therefore, it holds that LCP(M', q') has a solution if and only if LCP(M, q) has a solution.

The resulting LCP(M', q') is an instance of k'-LCP where $k' < k$. As in (12)–(16), each equation in $w' - M'z' = q'$ has at most $\max(3, 1 + \lceil p/2 \rceil, 1 + r) \leq \max(3, 1 + (p + r + 1)/2) < p + r$ nonzero coefficients, by the assumption that $p + r > 3$ and $p \geq r$.

It remains to show that the transformation is performed $\mathrm{O}(n^3)$ times until the resulting LCP(M', q') is an instance of 3-LCP, and that the row sizes of M' and q' are also $\mathrm{O}(n^3)$. Note that Case 1 is performed only at the first transformation. After the kth repetition, it is observed that the maximum number of nonzero elements per row in M' is at most $\max(3, 3(1 - 1/2^k) + n/2^k)$. For each equation in the original LCP, at most six new equations are generated at the first transformation, and then four equations are generated at the kth repetition, where $k > 1$. Thus, after transforming $\lceil \log n \rceil$ times, we obtain the 3-LCP(M', q'), whose order is equal to the sum of the order of the original LCP and the number of generated equations, that is, $n + n \times 6 \times 4^{\log n + 1} = \mathrm{O}(n^3)$.