Two Choices Are Enough for P-LCPs, USOs, and Colorful Tangents

Michaela Borzechowski
Department of Mathematics and Computer Science, Freie Universität Berlin, Germany

John Fearnley
Department of Computer Science, University of Liverpool, UK

Spencer Gordon
Department of Computer Science, University of Liverpool, UK

Rahul Savani
The Alan Turing Institute, London, UK
Department of Computer Science, University of Liverpool, UK

Patrick Schnider
Department of Computer Science, ETH Zürich, Switzerland

Simon Weber
Department of Computer Science, ETH Zürich, Switzerland

Abstract

We provide polynomial-time reductions between three search problems from three distinct areas: the P-matrix linear complementarity problem (P-LCP), finding the sink of a unique sink orientation (USO), and a variant of the α-Ham Sandwich problem. For all three settings, we show that "two choices are enough", meaning that the general non-binary version of the problem can be reduced in polynomial time to the binary version. This specifically means that generalized P-LCPs are equivalent to P-LCPs, and grid USOs are equivalent to cube USOs. These results are obtained by showing that both the P-LCP and our α-Ham Sandwich variant are equivalent to a new problem we introduce, P-Lin-Bellman. This problem can be seen as a new tool for formulating problems as P-LCPs.

2012 ACM Subject Classification Theory of computation → Problems, reductions and completeness; Theory of computation → Continuous optimization; Mathematics of computing → Combinatoric problems; Theory of computation → Computational geometry

Keywords and phrases P-LCP, Unique Sink Orientation, α-Ham Sandwich, search complexity, TFNP, UEOPL

Digital Object Identifier 10.4230/LIPIcs.ICALP.2024.32

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: `https://arxiv.org/abs/2402.07683`

Funding *Michaela Borzechowski*: DFG within GRK 2434 *Facets of Complexity*
John Fearnley: EPSRC Grant EP/W014750/1
Spencer Gordon: EPSRC Grant EP/W014750/1.
Rahul Savani: EPSRC Grant EP/W014750/1
Simon Weber: Swiss National Science Foundation under project no. 204320

Acknowledgements We wish to thank Bernd Gärtner for introducing the authors to each other.

51st International Colloquium on Automata, Languages, and Programming (ICALP 2024).
Editors: Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson;
Article No. 32; pp. 32:1–32:18
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

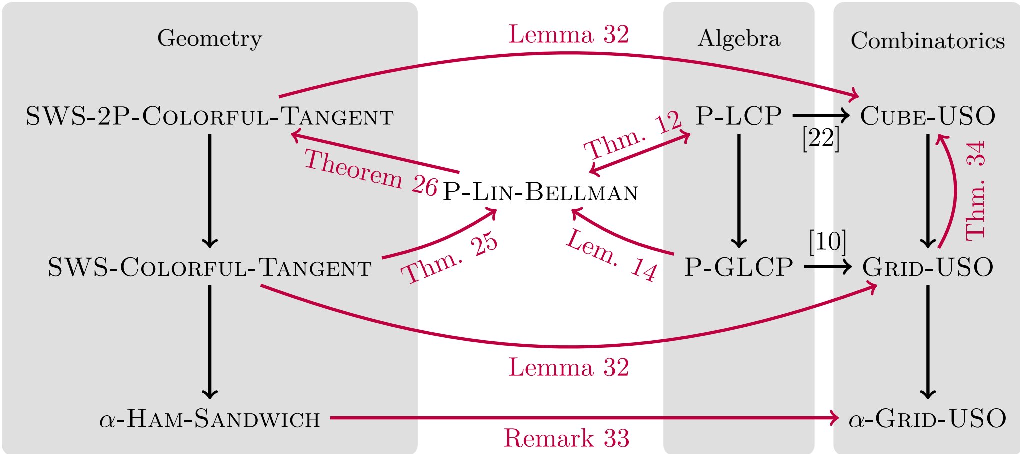

Figure 1 Red: Reductions we show in this paper. Black: Existing reductions and trivial inclusions.

In this paper we study three search problems from three distinct areas: the problem of solving a P-matrix linear complementarity problem (P-LCP), an algebraic problem, the problem of finding the sink of a unique sink orientation (USO), a combinatorial problem, and a variant of the α-Ham Sandwich problem, a geometric problem. Our results are a suite of reductions between these problems, which are shown in Figure 1.

There are two main themes for these reductions.

- **Two choices are enough.** For all three settings, the problems can be restricted to variants in which all choices are binary. In all three cases we show that the general non-binary problem can be reduced in polynomial time to the binary version.
- **A new tool for working with P-LCPs.** We introduce the P-Lin-Bellman problem, which serves as a crucial intermediate problem for our reductions with P-LCPs, and provides a new tool for showing equivalence to the P-LCP problem.

We now describe each of the three problems and our results.

P-Matrix LCPs. In the *Linear Complementarity Problem* (LCP), we are given an $n \times n$ matrix M and an n-dimensional vector q, and we are asked to find two n-dimensional vectors w, z such that $w = Mz + q$, both w and z are non-negative, and w and z satisfy the *complementarity* condition of $w^T z = 0$. In general, there is no guarantee that an LCP has a solution. However, if M is promised to be a P-matrix, that is, all its principal minors are positive, then the LCP problem always has a unique solution for every possible q [19], and we call this the P-LCP problem.

The P-LCP problem is important because many optimization problems reduce to it, for example, *Linear Programming* [12, 16] and *Strictly Convex Quadratic Programming* [5, 20], and solving a *Simple Stochastic Game (SSG)* [11, 23]. However, the complexity status of the P-LCP remains a major open question. The P-LCP problem is not known to be polynomial-time solvable, but NP-hardness (in the sense that an oracle for it could be used to solve SAT in polynomial time) would imply NP = co-NP [18].

The P-LCP problem naturally encodes problems that have *two choices*. This property arises from the complementarity condition, where for each index i, one must choose either $w_i = 0$ or $z_i = 0$. Thus the problem can be seen as making one of two choices for each of the

n possible dimensions of w and z. For example, this means that the direct encoding of a Simple Stochastic Game as a P-LCP [16] only works for *binary* games, in which each vertex has exactly two outgoing edges, and for each vertex the choice between the two outgoing edges is encoded by choosing between w_i and z_i.

To directly encode a non-binary SSG one must instead turn to *generalized* LCPs, which allow for more than two choices in the complementarity condition [23]. Generalized LCPs, which were defined by Habetler and Szanc [15], also have a P-matrix version, which we will refer to as P-GLCP.

Two choices are enough for P-LCPs. Our first main result is to show that P-GLCP and P-LCP are polynomial-time equivalent problems. Every P-LCP is a P-GLCP by definition, so our contribution is to give a polynomial-time reduction from P-GLCP to P-LCP, meaning that any problem that can be formulated as a P-GLCP can also be efficiently recast as a P-LCP. Such a result was already claimed in 1995, but later a counterexample to a crucial step in the proof was found by the same author [6, 7].

To show this result we draw inspiration from the world of infinite games. SSGs are a special case of *stochastic discounted games* (SDGs), which are known to be reducible in polynomial-time to the P-LCP problem [16, 17]. This reduction first writes down a system of *Bellman equations* for the game, which are a system of linear equations using min and max operations. These equations are then formulated as a P-LCP using the complementarity condition to encode the min and max operations.

We introduce a generalization of the Bellman equations for SDGs, which we call P-Lin-Bellman. We show that the existing reduction from SDGs to P-LCP continues to work for P-Lin-Bellman, and we show that we can polynomial-time reduce P-LCP to P-Lin-Bellman. Thus, we obtain a Bellman-equation type problem that is equivalent to P-LCP.

Then we use P-Lin-Bellman as a tool to reduce P-GLCP to P-LCP. Specifically we show that a P-GLCP can be reduced to P-Lin-Bellman. Here our use of P-Lin-Bellman as an intermediary shows its usefulness: while it is not at all clear how one can reduce P-GLCP to P-LCP directly, when both problems are formulated as P-Lin-Bellman instances, their equivalence essentially boils down to the fact that $\max(a, b, c) = \max(a, \max(b, c))$.

Unique Sink Orientations. A Unique Sink Orientation (USO) is an orientation of the n-dimensional hypercube such that every subcube contains a unique sink. An example of an USO can be found in Figure 2. The goal is to find the unique sink of the entire hypercube. USOs were introduced by Stickney and Watson [22] as a combinatorial abstraction of the candidate solutions of the P-LCP and have been studied ever since Szabó and Welzl formally defined them in 2001 [24].

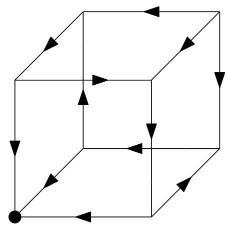

Figure 2 A 3-dimensional USO. The marked vertex denotes its unique global sink.

USOs have received much attention because many problems are known to reduce to them. This includes linear programming and more generally convex quadratic programming, simple stochastic games, and finding the smallest enclosing ball of given points.

As with P-LCPs, USOs naturally capture problems in which there are two choices. Each dimension of the cube allows us to pick between one of two faces, and thus each vertex of the cube is the result of making n binary choices. To remove this restriction, prior work has studied unique sink orientations of products of complete graphs [10], which the literature somewhat confusingly calls *grid USOs*. For example, as with P-LCP, the direct reduction from SSGs to USOs only works for binary games, while a direct reduction from non-binary games yields a grid USO.

Two choices are enough for USOs. Our second main result is to show that grid USOs and cube USOs are polynomial-time equivalent problems. Every cube USO is a grid USO by definition, so our contribution is to provide a polynomial-time reduction from grid USOs to cube USOs. Despite many researchers suspecting that grid USOs are a strict generalization of cube USOs, our result shows that at least in the promise setting, the two problems are computationally equivalent. For the reduction we embed a k-regular grid into a (also k-regular) k-cube. The main challenge to overcome is that the k-cube contains many more vertices and significantly more edges. These edges need to be oriented in such a way to be consistent with the orientation of the grid, and this orientation also needs to be computable locally, i.e., without looking at the whole orientation of the grid.

It should be noted that while P-LCP is known to reduce to cube USOs, and P-GLCP is known to reduce to grid USOs, neither of our "two choices is enough" results imply each other. This is because there is no known reduction from an USO-type problem to an LCP-type problem.

Equivalence between P-LCP and Colorful Tangent Problems. The Ham Sandwich theorem is a classical theorem in computational geometry: given d sets of points in $\mathbb{R}^d$, we can find a hyperplane that simultaneously *bisects* all of the sets. Steiger and Zhao [21] have shown that in a restricted input setting we can not only bisect, but cut off arbitrary fractions of each point set. This is the so-called α-Ham Sandwich theorem: if the point sets are *well-separated*, then for each vector α there exists a unique choice of one point per set, such that the hyperplane spanned by these points has exactly α_i of the points from set i above. The α-Ham Sandwich problem asks to determine this unique choice of one point per set.

In this paper, we consider a restricted variant of the α-Ham Sandwich problem. Firstly, we slightly strengthen the assumption of well-separation to *strong* well-separation. Secondly we restrict the input vector α such that each entry takes either be the minimum or maximum possible value. This means that for every point set, either all points of the set must lie above, or all below the desired α-cut. Thus, the α-cuts we search for are tangents to all the point sets. Combining the assumption of strong well-separation and the solutions being tangents, we call this problem SWS-Colorful-Tangent. We also consider the binary variant of the problem, which we call SWS-2P-Colorful-Tangent, where we restrict the size of each point set to 2; finding an α-cut then corresponds to a series of binary choices, one per set.

Here our contribution is to show that SWS-Colorful-Tangent is polynomial-time equivalent to the P-LCP problem. Our new intermediate problem P-Lin-Bellman plays a crucial role in this result: we give a polynomial-time reduction from SWS-Colorful-Tangent to P-Lin-Bellman, and a polynomial-time reduction from P-Lin-Bellman to SWS-2P-Colorful-Tangent. This also shows that two choices are enough for this problem as well.

For these reductions, we consider the point-hyperplane dual of the colorful tangent problems. In the dual, every input point becomes a hyperplane, and the solution we search for is a point lying on one hyperplane of each color, and lying either above or below all other hyperplanes. This can be encoded in the min and max operations of the P-Lin-Bellman problem. This demonstrates the usefulness of P-Lin-Bellman as an intermediate problem, since it has allowed us to show what is – to the best of our knowledge – the first polynomial-time equivalence between P-LCP and another problem.

Related Work. All problems we consider in this paper are promise search problems which lie in the complexity class PromiseUEOPL [2, 3], which is the promise setting analogue of the class UEOPL (short for Unique End of Potential Line) [9]. The latter has recently attracted the attention of many researchers trying to further the understanding of the hierarchy of total search problem classes, with a breakthrough result separating UEOPL from EOPL [14]. There is only one known natural[1] complete problem for UEOPL, called One Permutation Discrete Contraction (OPDC). OPDC also admits a natural binary variant, and it has already been shown implicitly that the binary variant is polynomial-time equivalent to the general variant [9]. Our reductions may help in finding another natural UEOPL-complete problem, even though our results for now only hold in the promise setting.

2 A New Intermediate Problem: P-Lin-Bellman

Our new P-Lin-Bellman problem is motivated by discounted games. A stochastic discounted game (SDG) is defined by a set of states S, which are partitioned into S_{Max} and S_{Min}, a set of actions A, a transition function $p : S \times A \times S \to \mathbb{R}$, a reward function $r : S \times A \to \mathbb{R}$, and a discount factor λ. The value of a SDG is known to be the solution to the following system of *Bellman equations.* For each state s we have an equation

$$x_s = \begin{cases} \max_{a \in A} \left(r(s,a) + \lambda \cdot \sum_{s' \in S} p(s,a,s') \cdot x_{s'} \right) & \text{if } s \in S_{\text{Max}}, \\ \min_{a \in A} \left(r(s,a) + \lambda \cdot \sum_{s' \in S} p(s,a,s') \cdot x_{s'} \right) & \text{if } s \in S_{\text{Min}}. \end{cases}$$

Prior work has shown that, if the game is binary, meaning that $|A| = 2$, then these Bellman equations can be formulated as a P-LCP [17, 16].

For our purposes, we are interested in the format of these equations. Note that each equation is a max or min over affine functions of the other variables. We capture this idea in the following generalized definition.

▶ **Definition 1.** *A* Lin-Bellman *system $G = (L, R, q, S)$ is defined by two matrices $L, R \in \mathbb{R}^{n \times n}$, a vector $q \in \mathbb{R}^n$, and a set $S \subseteq \{1, 2, \ldots, n\}$. These inputs define the following system of equations over a vector of variables $x \in \mathbb{R}^n$:*

$$x_i = \begin{cases} \max(\sum_{1 \le j \le n} L_{ij} x_j + q_i, \ \sum_{1 \le j \le n} R_{ij} x_j) & \text{if } i \in S, \\ \min(\sum_{1 \le j \le n} L_{ij} x_j + q_i, \ \sum_{1 \le j \le n} R_{ij} x_j) & \text{if } i \notin S. \end{cases} \tag{1}$$

Observe that this definition captures systems of equations in which a min or max operation is taken over two affine expressions in the other variables. Note that here we have included the additive q term only in the first of the two expressions (the second is thus linear), whereas the SDG equations have additive terms in all of the affine expressions. This is because, for our reductions, we do not need the second additive term. Otherwise, this definition captures the style of Bellman equations that are used in SDGs and other infinite games.

[1] By natural we mean a problem that is not a variant of the Unique End of Potential Line problem which naturally characterizes the class.

We say that $x \in \mathbb{R}^n$ is a solution of a Lin-Bellman system G if Equation (1) is satisfied for all i. In general, however, such an x may not exist or may not be unique. To get a problem that is equivalent to P-LCP, we need a restriction that ensures that the problem always has a unique solution, like the P-matrix restriction on the LCP problem.

To make sure that a unique solution exists, we introduce a promise to yield the promise problem P-Lin-Bellman. As the promise we guarantee the same property that is implied by the promise of the P-LCP: For every $q' \in \mathbb{R}^n$, the given Lin-Bellman system shall have a unique solution. We use the following restriction.

▶ **Definition 2.** *P-Lin-Bellman*
Input: *A Lin-Bellman system $G = (L, R, q, S)$.*
Promise: *The Lin-Bellman system (L, R, q', S) has a unique solution for every $q' \in \mathbb{R}^n$.*
Output: *A solution $x \in \mathbb{R}^n$ of G.*

This promise insists that the linear equations not only have a unique solution for the given vector q, but that they also have a unique solution no matter what vector q is given. This is analogous to a property from SDGs: an SDG has a unique solution no matter the rewards for the actions, and this corresponds to the additive q vector in the problem above.

Similarly to the LCP, where unique solutions for any right-hand side q imply that the matrix M is a P-matrix (and vice versa), this promise of P-Lin-Bellman implies something about the involved matrices. However, for P-Lin-Bellman we do not have a full characterization of the systems $(L, R, \cdot, S)$ that fulfill the promise. We only have the following rather weak necessary condition (proven in the full version of the paper), which is however useful for several of our reductions.

▶ **Lemma 3.** *In any P-Lin-Bellman instance, $R - I$ is invertible.*

Since P-Lin-Bellman is so similar to the P-LCP, we will prove the equivalence of P-LCP and P-Lin-Bellman first, in the next section.

3 Linear Complementarity Problems

We begin by giving the definitions of the (binary) linear complementarity problems.

▶ **Definition 4.** *LCP(M, q)*
Input: *An $n \times n$ matrix M and a vector $q \in \mathbb{R}^n$.*
Output: *Two vectors $w, z \in \mathbb{R}^n$ such that*
- $w = Mz + q$,
- $w, z \geq 0$, *and*
- $w^T \cdot z = 0$.

We are particularly interested in the case where the input matrix M is a P-matrix.

▶ **Definition 5.** *An $n \times n$ matrix M is a P-matrix if every principal minor of M is positive.*

One particularly interesting feature of the P-matrix Linear Complementarity Problem is that it always has a unique solution.

▶ **Theorem 6** ([5])**.** *M is a P-matrix if and only if LCP(M, q) has a unique solution for every $q \in \mathbb{R}^n$.*

There are two problems associated with P-matrix LCPs. The first problem is a total search problem in which we must either solve the LCP, or show that the input matrix M is not a P-matrix by producing a non-positive principal minor of M. The second problem is a promise problem, where the promise is that the input is a P-matrix.

▶ **Definition 7.** *P-LCP*(M, q)
Input: *An* $n \times n$ *matrix* M *and a vector* $q \in \mathbb{R}^n$.
Promise: M *is a P-matrix.*
Output: *Two vectors* $w, z \in \mathbb{R}^n$ *that are solutions of LCP*(M, q).

In the next two sections, we will show that P-Lin-Bellman and P-LCP are equivalent under polynomial-time many-one reductions.

3.1 P-LCP to P-Lin-Bellman

Let $M \in \mathbb{R}^{n\times n}$ and $q \in \mathbb{R}^n$ be an instance of P-LCP. We will build a P-Lin-Bellman instance $G(M, q)$ in the following way. For each i in the range $1 \leq i \leq n$, we set

$$x_i = \min((Mx + x)_i + q_i, 2x_i). \tag{2}$$

In other words, we set $L = M + I$, $R = 2I$, and $S = \emptyset$. We first show that every solution of $G(M, q)$ corresponds to a solution of the LCP. We do not need to use any properties of the P-matrix in this part of the proof.

▶ **Lemma 8.** *A vector* $z \in \mathbb{R}^n$ *is a solution of the Lin-Bellman system* $G(M, q)$ *if and only if* z *and* $w = Mz + q$ *are a solution of LCP*(M, q).

Proof. If z solves Equation (2) then we have $\min(Mz + z + q, 2z) - z = 0$, and hence that $\min(Mz + q, z) = 0$. This implies that both z and w are non-negative:

$$\begin{aligned} z &\geq \min(Mz + q, z) = 0, \\ w = Mz + q &\geq \min(Mz + q, z) = 0. \end{aligned}$$

Moreover, since $\min(Mz + q, z) = 0$, the complementarity condition also holds. Hence w and z are a solution of the LCP.

In the other direction, if w and z are solutions of the LCP, then we argue that z must be a solution of $G(M, q)$. This is because any solution of the LCP satisfies $\min(Mz + q, z) = 0$ due to the complementarity condition, and so we must have that z solves Equation (2). ◀

Next we show that the promise of P-Lin-Bellman is also satisfied.

▶ **Lemma 9.** *If* M *is a P-matrix, then* $G(M, q)$ *satisfies the promise of P-Lin-Bellman.*

Proof. We must show that the Lin-Bellman system $G(M, q')$ has a unique solution for every q'. This follows from Lemma 8, which shows that $G(M, q')$ has a unique solution if and only if the LCP defined by M and q' has a unique solution, which follows from the fact that M is a P-matrix. ◀

3.2 P-Lin-Bellman to P-LCP

For this direction, we follow and generalize the approach used by Jurdziński and Savani [17]. Suppose that we have a P-Lin-Bellman instance defined by $G = (L, R, q, S)$.

The reduction is based on the idea of simulating the operation $x = \max(a, b)$ by introducing two non-negative slack variables w and z:

$$x - w = a, \qquad x - z = b, \qquad w, z \geq 0, \qquad w \cdot z = 0.$$

Since w and z are both non-negative, any solution to the system of equations above must satisfy $x \geq \max(a, b)$. Moreover, since the complementarity condition insists that either $w = 0$ or $z = 0$, we have that $x = \max(a, b)$.

An operation $x = \min(a, b)$ can likewise be simulated by

$$x + w = a, \qquad x + z = b, \qquad w, z \geq 0, \qquad w \cdot z = 0.$$

We use this technique to rewrite the system of equations given in (1) in the following way. For each i in $1 \leq i \leq n$ we have the following.

$$x_i + w_i = \sum_{1 \leq j \leq n} L_{ij} \cdot x_j + q_i, \qquad \text{if } i \notin S \tag{3}$$

$$x_i - w_i = \sum_{1 \leq j \leq n} L_{ij} \cdot x_j + q_i, \qquad \text{if } i \in S \tag{4}$$

$$x_i + z_i = \sum_{1 \leq j \leq n} R_{ij} \cdot x_j, \qquad \text{if } i \notin S \tag{5}$$

$$x_i - z_i = \sum_{1 \leq j \leq n} R_{ij} \cdot x_j, \qquad \text{if } i \in S \tag{6}$$

$$w_i, z_i \geq 0 \tag{7}$$

$$w_i \cdot z_i = 0. \tag{8}$$

We have already argued that this will correctly simulate the max and min operations in Equation (1), and so we have the following lemma.

▶ **Lemma 10.** *$x \in \mathbb{R}^n$ is a solution of G if and only if x is a solution of the system defined by Equations* (3)*–*(8)*.*

We shall now reformulate Equations (3)–(8) as an LCP. To achieve this we first introduce some helpful notation: For every $n \times n$ matrix A, we define $\widehat{A}$ in the following way. For each $i, j \in \{1, 2, \ldots, n\}$ we let

$$\widehat{A}_{ij} := \begin{cases} A_{ij} & \text{if } i \in S, \\ -A_{ij} & \text{if } i \notin S. \end{cases}$$

Equations (3)–(6) can be rewritten as the following matrix equations:

$$\widehat{I}x = w + \widehat{L}x + \widehat{I}q,$$

$$\widehat{I}x = z + \widehat{R}x,$$

Eliminating x yields $(\widehat{I} - \widehat{L})(\widehat{I} - \widehat{R})^{-1}z = w + \widehat{I}q$, which is equivalent to $w = Mz + q'$ for $M = (\widehat{I} - \widehat{L})(\widehat{I} - \widehat{R})^{-1}, \ q' = -\widehat{I}q$.

We rely here on the fact that by Lemma 3, $R - I$ and thus also $(\widehat{I} - \widehat{R})$ is invertible. We have so far shown the following lemma.

▶ **Lemma 11.** *A vector $x \in \mathbb{R}^n$ is a solution of G if and only if there are vectors w, z such that w, z are a solution of $LCP(M, q)$ and x, w, and z are a solution of Equations* (3)*–*(8)*.*

To show that M is a P matrix, we must show that $\text{LCP}(M, q'')$ has a unique solution for every $q'' \in \mathbb{R}^n$. By Lemma 11, this holds if and only if $G(L, R, q'', S)$ has a unique solution for every q'', which holds due to the P-Lin-Bellman promise. Our reduction is thus correct, and we have established our first main result:

▶ **Theorem 12.** *P-Lin-Bellman and P-LCP are equivalent under polynomial-time many-one reductions.*

3.3 P-matrix Generalized LCP to P-LCP

Generalized LCPs were originally introduced by Cottle and Dantzig [4]. A generalized LCP instance is defined by a vertical block matrix M, and a vertical block vector q where

$$M = \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_n \end{bmatrix} \qquad\qquad q = \begin{pmatrix} q_1 \\ q_2 \\ \vdots \\ q_n \end{pmatrix}$$

Each matrix M_i has dimensionality $b_i \times n$, while each vector q_i has b_i dimensions. Thus, if $N = \sum_i b_i$, we have that $M \in \mathbb{R}^{N \times n}$ and $q \in \mathbb{R}^N$. Given a vertical block vector x with n blocks, we use x_i^j to refer to the jth element of the ith block of x.

Given such an M and q, the *generalized linear complementarity problem* (GLCP) is to find vectors $w \in \mathbb{R}^N$ and $z \in \mathbb{R}^n$ such that

- $w = Mz + q$,
- $w \geq 0$,
- $z \geq 0$, and
- $z_i \cdot \prod_{j=1}^{b_i} w_i^j = 0$ for all i.

Given a tuple $p = (p_1, p_2, \ldots, p_n)$ such that each p_i lies in the range $1 \leq i \leq b_i$, the representative submatrix of M defined by p is given by $M(p) \in \mathbb{R}^{n \times n}$ and is constructed by selecting row p_i from each block-matrix M_i. We then say that M is a P-matrix if all of its representative submatrices are P-matrices.

We define the P-matrix GLCP (P-GLCP) as a promise problem, analogously to the P-LCP: The input is a GLCP instance (M, q) with the promise that M is a P-matrix. This promise again guarantees unique solutions:

▶ **Theorem 13** (Habetler, Szanc [15]). *A vertical block matrix M is a P-matrix if and only if the GLCP instance (M, q') has a unique solution for every $q' \in \mathbb{R}^N$.*

We are now ready to show our reduction of P-GLCP to P-Lin-Bellman.

▶ **Lemma 14.** *There is a poly-time many-one reduction from P-GLCP to P-Lin-Bellman.*

Proof. We can turn a P-matrix GLCP instance into a P-Lin-Bellman instance in much the same way as we did for P-LCPs in Section 3.1. For each i in the range $1 \leq i \leq n$ we set

$$z_i = \min\Big(\min\big\{(Mz+q)_i^j + z_i \;\; : 1 \leq j \leq b_i\big\}, 2z_i\Big). \tag{9}$$

We claim that any solution of the system of equations defined above yields a solution of the GLCP. In any solution to the system we have

$$\min\Big(\min\big\{(Mz+q)_i^j \;\; : 1 \leq j \leq b_i\big\}, z_i\Big) = 0.$$

So if we set $w_i^j = (Mz+q)_i^j$ then we have the following non-negativities

$$\begin{aligned} w_i^j = (Mz+q)_i^j \geq \min\Big(\min\big\{(Mz+q)_i^j \;\; : 1 \leq j \leq b_i\big\}, z_i\Big) = 0. \\ z_i \geq \min\Big(\min\big\{(Mz+q)_i^j \;\; : 1 \leq j \leq b_i\big\}, z_i\Big) = 0. \end{aligned}$$

We can also see that the complementarity condition is satisfied, since either $z_i = 0$ or $w_i^j = 0$ for some j.

To write this as a Lin-Bellman system, we must carefully write Equation (9) using two-input min operations. We will introduce a variable x_i^j for each i in the range $1 \leq i \leq n$ and each j in the range $1 \leq j \leq b_i$. We use $\mathbf{x}^1$ to denote the vector $(x_1^1, x_2^1, \ldots, x_n^1)$. For each i, and each j in the range $1 \leq j < b_i$ we use the following.

$$x_i^j = \min\Big((M\mathbf{x}^1 + q)_i^j + x_i^1,\ x_i^{j+1}\Big) \tag{10}$$

We also use the following when $j = b_i$.

$$x_i^{b_i} = \min\Big((M\mathbf{x}^1 + q)_i^{b_i} + x_i^1,\ 2x_i^1\Big) \tag{11}$$

Let $G = (L, R, q, S)$ denote the resulting Lin-Bellman system. It is now easy to see that on the one hand, for every solution $\mathbf{x}$ of G the vectors $z := \mathbf{x}^1$ and w with $w_i^j = (Mz + q)_i^j$ form a solution of the input P-matrix GLCP. On the other hand, from any solution to the GLCP we can extract the vector $\mathbf{x}^1 := z$. Given this vector $\mathbf{x}^1$ there is only one way to extend this to an assignment for all x_i^j that is a solution to G. We thus get a one-to-one correspondence between solutions to G and solutions to the GLCP. Since the GLCP defined by M is promised to have a unique solution for every q', this thus implies that the Lin-Bellman system $G' = (L, R, q', S)$ also has a unique solution for every q'. Thus, this is indeed a P-Lin-Bellman instance. ◀

Combining this with the previously proven Theorem 12, we get the following corollary:

▶ **Corollary 15.** *P-GLCP and P-LCP are polynomial-time equivalent.*

4 Colorful Tangents and P-Lin-Bellman

Let us start by introducing the definitions and prior results on the α-Ham Sandwich theorem.

▶ **Definition 16.** *A family of point sets $P_1 = \{p_{1,1}, \ldots, p_{1,s_1}\}, P_2, \ldots, P_d \subset \mathbb{R}^d$ is said to be* well-separated *if for any non-empty index set $I \subset [d]$, there exists a hyperplane h such that h separates the points in $\bigcup_{i \in I} P_i$ from those in $\bigcup_{i \in [d] \setminus I} P_i$.*

In the setting of well-separated point sets, the classical Ham Sandwich theorem can be strengthened to the more modern α-Ham Sandwich theorem due to Steiger and Zhao [21], for which we need the definition of an $(\alpha_1, \ldots, \alpha_d)$-cut:

▶ **Definition 17.** *Given positive integers $\alpha_i \leq |P_i|$ for all $i \in [d]$, an $(\alpha_1, \ldots, \alpha_d)$-cut is a hyperplane h such that $h \cap P_i \neq \emptyset$ and $|h^+ \cap P_i| = \alpha_i$ for all $i \in [d]$.*

In this definition and in the rest of this section, h^+ and h^- denote the two closed halfspaces bounded by a hyperplane h. When h is a colorful hyperplane (i.e., a hyperplane containing a colorful point set, which is a set $\{p_1, \ldots, p_d\}$ with $p_i \in P_i$), its positive side is determined by the orientation of the points $p_i \in P_i$ spanning it.

▶ **Theorem 18** (α-Ham Sandwich theorem [21])**.** *Given d point sets $P_1, \ldots, P_d \subset \mathbb{R}^d$ that are well-separated and in weak general position*[2]*, for every $(\alpha_1, \ldots, \alpha_d)$ where $1 \leq \alpha_i \leq |P_i|$, there exists a unique $(\alpha_1, \ldots, \alpha_d)$-cut.*

[2] Weak general position is a weaker version of general position. We will not go further into this since we will always ensure classical general position when invoking this theorem.

The α-Ham Sandwich theorem guarantees us existence and uniqueness of an $(\alpha_1, \ldots, \alpha_d)$-cut, and it is thus not too surprising that the problem of finding such a cut is contained in UEOPL [3].

In this section, we wish to show equivalence of this problem to P-Lin-Bellman, however, we only manage this with some slight changes. Firstly, we wish to drop the assumption of weak general position, since it is not easily guaranteed when reducing from P-Lin-Bellman. Because of this we need to weaken the definition of $(\alpha_1, \ldots, \alpha_d)$-cuts accordingly:

▶ **Definition 19.** *Given a well-separated family of point sets $P_1, \ldots, P_d \subset \mathbb{R}^d$ and $(\alpha_1, \ldots, \alpha_d)$ such that $1 \leq \alpha_i \leq |P_i|$, a hyperplane h is an $(\alpha_1, \ldots, \alpha_d)$-cut if for all $i \in [d]$, $h \cap P_i \neq \emptyset$ and $|(h^+ \setminus h) \cap P_i| + 1 \leq \alpha_i \leq |h^+ \cap P_i|$.*

In other words, we allow a hyperplane to contain more than one point per color, and all the additional points may be "counted" for either side of the hyperplane. Note that even without any general position assumption, the affine hull of every colorful set of points must have dimension at least $d-1$, and thus span a unique hyperplane (see the full version of the paper). We further show in the full version that every colorful subset of points on the same hyperplane h must be oriented the same way, and thus h^+ is uniquely defined even without the weak general position assumption.

Note that the α-Ham Sandwich theorem (Theorem 18) generalizes directly to the setting where the assumption of weak general position is dropped and Definition 17 is replaced by Definition 19.

▶ **Theorem 20.** *Given a well-separated family of point sets $P_1, P_2, \ldots, P_d \subset \mathbb{R}^d$ and $(\alpha_1, \ldots, \alpha_d)$ for $1 \leq \alpha_i \leq |P_i|$, there exists a unique hyperplane h that is an $(\alpha_1, \ldots, \alpha_d)$-cut according to Definition 19.*

Proof. This simply follows from perturbing the points and applying Theorem 18. ◀

Secondly, we have to strengthen well-separation to *strong well-separation.* We illustrate well-separation and strong well-separation in Figure 3.

▶ **Definition 21.** *A family of point sets $P_1 = \{p_{1,1}, \ldots, p_{1,s_1}\}, P_2, \ldots, P_d \subset \mathbb{R}^d$ is said to be* strongly well-separated *if the point sets $P'_1, \ldots, P'_d$ obtained by projecting $P_1, \ldots, P_d$ to the hyperplane spanned by $p_{1,1}, \ldots, p_{d,1}$ are well-separated.*

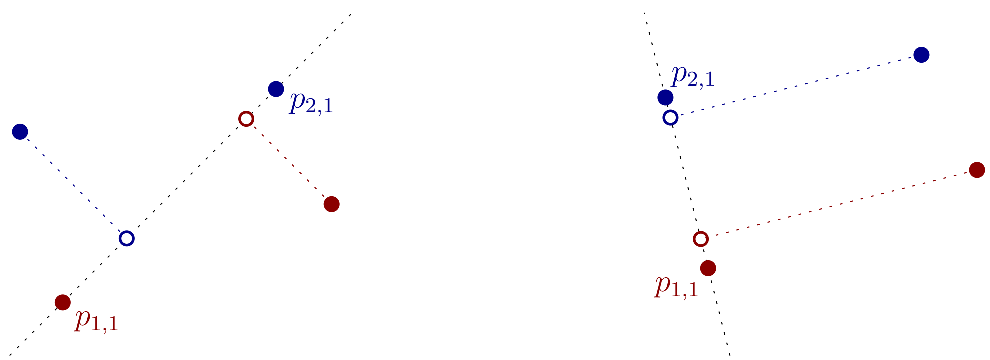

Figure 3 The point sets (dots) on the left are well-separated but not strongly well-separated, since their projections (circles) are not well-separated. The point sets on the right are strongly well-separated.

We are now ready to introduce the problem we wish to study.

▶ **Definition 22.** *SWS-Colorful-Tangent*$(P_1, \dots, P_d, \alpha_1, \dots, \alpha_d)$
Input: *d point sets $P_1, \dots, P_d \subset \mathbb{R}^d$ and a vector $(\alpha_1, \dots, \alpha_d)$, where $\alpha_i \in \{1, |P_i|\}$.*
Promise: *The point sets $P_1, \dots, P_d$ are strongly well-separated.*
Output: *An $(\alpha_1, \dots, \alpha_d)$-cut.*
We depart from the *Ham Sandwich* name for this problem since for $\alpha_i \in \{1, |P_i|\}$, the α-"cuts" are really just colorful tangent hyperplanes, as can be seen in Figure 4.

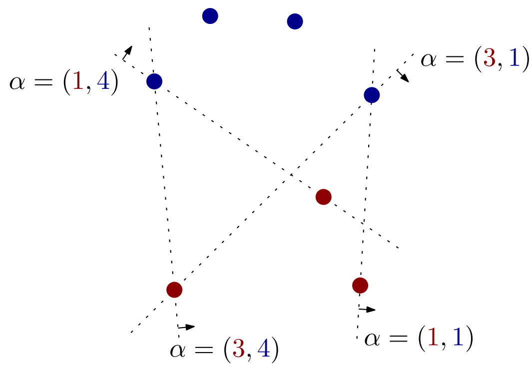

Figure 4 Two well-separated point sets and the solution α-cuts to all four possible α-vectors. Note that the positive side of a colorful line is considered to be the right side when orienting the line from the red point to the blue point.

Similarly to P-GLCP and its binary variant P-LCP, we also define SWS-2P-Colorful-Tangent as the restriction of SWS-Colorful-Tangent to inputs where $|P_i| = 2$ for all $i \in [d]$. We now wish to prove these two problems polynomial-time equivalent to P-Lin-Bellman. To achieve this, we reduce SWS-Colorful-Tangent to P-Lin-Bellman, and P-Lin-Bellman to SWS-2P-Colorful-Tangent, with the reduction by inclusion from SWS-2P-Colorful-Tangent to SWS-Colorful-Tangent closing the cycle.

For our reductions, we will use the concept of *point-hyperplane duality*, as described by Edelsbrunner in [8, p.13]. Point-hyperplane duality is a bijective mapping from all points in $\mathbb{R}^d$ to all non-vertical hyperplanes in $\mathbb{R}^d$. A point $p = (p_1, \dots, p_d)$ is mapped to the hyperplane $p^* = \{x \in \mathbb{R}^d \mid 2p_1x_1 + \dots 2p_{d-1}x_{d-1} - x_d = p_d\}$. The hyperplane p^* is called *the dual of* p, and for a non-vertical hyperplane h, its dual h^* is the unique point p such that $p^* = h$.

Point-hyperplane duality has the nice properties of *incidence preservation* and *order preservation*: for any point p and non-vertical hyperplane h, we have that $p \in h$ if and only if $h^* \in p^*$, and furthermore we have that p lies above h if and only if h^* lies above p^*.

We are now ready to begin presenting our reductions. For all of these reductions, we use the following crucial yet simple observation on strongly well-separated point sets:

▶ **Observation 23.** *If a set of point sets $P_1, \dots, P_d$ is strongly well-separated, then every set of point sets $P'_1, \dots, P'_d$ obtained by moving points $p_{i,j}$ $(j \neq 1)$ orthogonally to the hyperplane spanned by $p_{1,1}, \dots, p_{d,1}$ is also strongly well-separated.*

The general idea of the reduction is to represent the linear inputs to a min or max operation in a Lin-Bellman system by a point $p_{i,1}$ and the affine inputs by a point $p_{i,j}$ $(j \neq 1)$. As a warm-up, we first only reduce from the two-point version.

▶ **Lemma 24.** *SWS-2P-*Colorful-Tangent *poly-time reduces to P-*Lin-Bellman*.*

Proof. We first linearly transform our point sets such that the plane through $p_{1,1}, \ldots, p_{d,1}$ is mapped to the horizontal plane $\{(x_1, \ldots, x_d) \in \mathbb{R}^d \mid x_d = 0\}$. Without loss of generality, we assume that after this transformation the colorful hyperplane h spanned by $p_{1,1}, \ldots, p_{d,1}$ is oriented upwards, i.e., its positive halfspace contains $(0, \ldots, 0, +\infty)$.

Then, we apply point-hyperplane duality. Each point $p_{i,j}$ becomes a hyperplane $h_{i,j} = p_{i,j}^*$, which can be described by some function $h_{i,j} : \mathbb{R}^d \to \mathbb{R}$ such that a point $x \in \mathbb{R}^d$ lies strictly above $h_{i,j}$ if $h_{i,j}(x) > 0$, and on $h_{i,j}$ if $h_{i,j}(x) = 0$. For the hyperplanes $h_{i,1}$ dual to our points $p_{i,1}$, we furthermore have that they go through the origin, i.e., $h_{i,j}$ is a linear function (and not only an affine function; it has no additive constant term).

Before applying duality, the desired $(\alpha_1, \ldots, \alpha_d)$-cut hyperplane is a hyperplane h containing at least one point p_i' per set P_i, such that the other point of P_i lies on or above h if and only if $\alpha_i = |P_i|$, and on or below h if and only if $\alpha_i = 1$. In the dual, this now means that we need to find a point p which lies on at least one of the hyperplanes $h_{i,1}, h_{i,2}$ for each i, and such that it lies above (below) or on both of these hyperplanes if $\alpha_i = |P_i|$ ($\alpha_i = 1$). This can be described by the equations $\min\{h_{i,1}(x), h_{i,2}(x)\} = 0$ if $\alpha_i = |P_i|$ (and max instead of min, otherwise). This can of course be rewritten as $x_i = \min\{h_{i,2}(x) + x_i, h_{i,1}(x) + x_i\}$. Note again that the second input to this minimum is *linear* in x (there is no additive term). With one such constraint each per point set P_i, we thus get a system of d equations over d variables, which together form a Lin-Bellman system.

It remains to check whether this system is also a P-Lin-Bellman instance. We first see that changing q simply corresponds to moving the hyperplanes $h_{i,2}$ in a parallel fashion (i.e., without changing their normal vectors), which in the primal corresponds to moving them in direction x_d. By Observation 23 and Theorem 20, we always have unique $(\alpha_1, \ldots, \alpha_d)$-cuts in this modified family of point sets, and thus the Lin-Bellman instance has a unique solution for all q'. ◀

▶ **Theorem 25.** *SWS-*Colorful-Tangent *poly-time reduces to P-*Lin-Bellman*.*

Proof (sketch). The proof works exactly the same way as the previous proof of Lemma 24. The only difference is that the equations that need to be encoded are of the form

$$x_i = \min\{h_{i,1}(x) + x_i, \ldots, h_{i,|P_i|}(x) + x_i\}, \tag{12}$$

where still only $h_{i,1}(x)$ is guaranteed to be a linear function. To encode this as a Lin-Bellman system, we apply the same trick as in the proof of Lemma 14 to split the multi-input minimum into multiple two-input minima, with the remaining details found in the full version of the paper. ◀

To complete our cycle of reductions, we need to reduce from P-Lin-Bellman to SWS-2P-Colorful-Tangent. For this reduction we basically perform the process from the proof of Lemma 24 in reverse. The details are found in the full version of the paper.

▶ **Theorem 26.** *P-*Lin-Bellman *poly-time reduces to SWS-2P-*Colorful-Tangent*.*

To end this section, we want to note that the assumption of strong well-separation is not too strong, at least combinatorially:

▶ **Theorem 27.** *For every family of well-separated point sets, there exists a family of strongly well-separated point sets with the same combinatorial structure, i.e., the two families have the exact same order type.*

The proof can be found in the full version of the paper. Note that the proof is merely existential and it is unlikely that the translation can be performed efficiently.

5 Grid-USO and Cube-USO

We saw in the two previous sections that both P-GLCP and SWS-Colorful-Tangent can be reduced to their respective "binary" variants, P-LCP and SWS-2P-Colorful-Tangent. In this section we discuss grid USOs and cube USOs, which are a combinatorial framework that can be used to model all the problems studied in the sections above as well as many more algebraic and geometric problems. Similarly to the previous sections, cube USOs are a restriction of grid USOs to the "binary" case.

5.1 Definitions

We define C as a d-dimensional hypercube. We say $V(C) := \{0,1\}^d$ and $\{J, K\} \in E(C)$ iff $|J \oplus K| = 1$, where $\oplus$ is the bit-wise xor operation (i.e., addition in $\mathbb{Z}_2^d$) and $|\cdot|$ counts the number of 1-entries. For notational simplicity, in this section we use the same name both for a bitvector in $\{0,1\}^d$ and for the set of dimensions $i \in [d]$ in which the vector is 1. For example for $J, K \in \{0,1\}^d$ we write $J \subseteq K$ if for all $i \in [d]$ with $J_i = 1$ we also have $K_i = 1$.

The orientation of the edges of the cube C is given by an *orientation* function $O : V(C) \to \{0,1\}^d$, where $O(J)_i = 0$ means J has an incoming edge in dimension i and $O(J)_i = 1$ is an outgoing edge in dimension i.

▶ **Definition 28.** *An orientation is a* unique sink orientation (USO) *if and only if every induced subcube has a unique sink.*

The common search problem version of this problem is to find the global sink of the cube, given the function O as a boolean circuit. Note that it is co-NP-complete to test whether a given orientation is USO [13]. We consider the promise version Cube-USO, which was one of the first search problems proven to lie in the complexity class Promise-UEOPL [9].

▶ **Definition 29.** *Cube-USO*
Input: *A circuit computing the orientation function O on a d-dimensional cube C.*
Promise: *O is a Unique Sink Orientation.*
Output: *A vertex $J \in V(C)$ which is a sink, i.e., $\forall i \in [d] : O(J)_i = 0$.*

While the d-dimensional hypercube is the product of d copies of K_2 (the complete graph on two vertices), a *grid graph* is the product of complete graphs of arbitrary size: A d-dimensional grid graph Γ is given by $n_1, \dots, n_d \in \mathbb{N}^+$:

$$V(\Gamma) := \{0, \dots, n_1\} \times \cdots \times \{0, \dots, n_d\},$$
$$E(\Gamma) := \{\{v, w\} \mid v, w \in V(\Gamma), \exists! i \in [d] : v_i \neq w_i\}.$$

We say the grid has d *dimensions* and each dimension i has $n_i + 1$ *directions*.

The subgraph Γ' of Γ induced by the vertices $V(\Gamma') = N_1 \times \cdots \times N_d$ for non-empty $N_i \subseteq \{0, \dots, n_i\}$ is called an *induced subgrid* of Γ. Note that if for some i we have $|N_i| = 1$, the induced subgrid loses a dimension. If $|N_i| = 1$ for all $i \in [d]$ except one, we say that the induced subgrid Γ' is a *simplex*. A simplex is a complete graph K_{n_j+1} for some $j \in [d]$.

The orientation of a grid is given by the *outmap function*, which assigns each vertex a binary vector that encodes whether its incident edges are incoming or outgoing. More formally, the outmap function is a function $\sigma : V(\Gamma) \to \{0,1\}^{n_1+\ldots+n_d}$, where $\sigma(v)_{n_1+\ldots+n_i+j} = 1$ denotes that the edge from v to its j-th neighbor w in dimension $i+1$ is outgoing, i.e., oriented from v to w. Note that any circuit computing σ has $n_1 + \ldots + n_d$ outputs, and is thus of size $\Omega(n_1 + \ldots + n_d)$.

For notational convenience, we denote the entry of $\sigma(v)$ relevant to the edge $\{v, w\}$ by $\sigma(v, w)$. In other words, for each pair of vertices v, w such that $\{v, w\} \in E(\Gamma)$, $\sigma(v, w) = 1$ iff the edge between v and w is oriented towards w.

▶ **Definition 30** (Gärtner, Morris, Rüst [10]). *An orientation of a grid graph is a* unique sink orientation (USO) *if and only if every induced subgrid has a unique sink.*

A unique sink orientation of a simplex with n vertices is equivalent to a permutation of n elements, i.e. every unique sink orientation of one simplex can be given by a total order of its vertices. The minimum element of the order is the source, the maximum element is the sink.

Since every cube is also a grid, the complexity of checking whether σ is USO is also co-NP-hard. So again, we consider the promise search version of this problem:

▶ **Definition 31.** *Grid-USO*
Input: *A d-dimensional grid $\Gamma = (n_1, \dots, n_d)$ and a circuit computing its outmap σ.*
Promise: *σ is a Unique Sink Orientation.*
Output: *A sink, i.e. a vertex $v \in V(\Gamma)$ s.t. $\forall w \in V(\Gamma)$ with $\{v, w\} \in E(\Gamma) : \sigma(v, w) = 0$.*

Just like Cube-USO, Grid-USO also lies in the search problem complexity class Promise-UEOPL [1]. It is well known that P-LCP reduces to Cube-USO [22] and P-GLCP reduces to Grid-USO [10]. Since SWS-2P-Colorful-Tangent and SWS-Colorful-Tangent can both be reduced to P-LCP and P-GLCP respectively, they also reduce to Cube-USO and Grid-USO respectively. However, we show a direct and straightforward reduction from SWS-2P-Colorful-Tangent to Cube-USO, and from SWS-Colorful-Tangent to Grid-USO. These direct reductions do not require strong well-separation, only classical well-separation. The proof can be found in the full version of the paper.

▶ **Lemma 32.** *Assuming general position of the input points, finding an α-cut in a well-separated point set family $P_1, \dots, P_d \subset \mathbb{R}^d$ for $\alpha_i \in \{1, |P_i|\}$ can be reduced to Grid-USO in polynomial time. The reduction goes to Cube-USO if for all i, $|P_i| = 2$.*

▶ Remark 33. If we instead want to find an α'-cut for arbitrary α', we can use the reduction from Lemma 32 with $\alpha = (1, \dots, 1)$, but instead of searching for a sink in the resulting grid USO Γ, we need to find a vertex with $\alpha'_i - 1$ outgoing edges in each dimension i.

By [10, Theorem 2.14], this vertex is guaranteed to exist and to be unique. We call the problem of searching for such a vertex α-Grid-USO. Note that α-Grid-USO is not known to reduce to regular Grid-USO, nor is it known to lie in Promise-UEOPL (compared to α-Ham Sandwich which does lie in Promise-UEOPL [3]).

5.2 Grid-USO to Cube-USO

In this section we show that every Grid-USO instance (Γ, σ) can be reduced to a Cube-USO instance (C,O) in polynomial time such that given the global sink of O, we can derive the global sink of σ in polynomial time.

We are given a d-dimensional grid $\Gamma = (n_1, \dots, n_d)$ and turn it into a hypercube of dimension $n := \sum n_i$. Note that Γ is n-regular and so is C. For every dimension of the grid spanned by $n_i + 1$ directions, we assign a block of n_i dimensions in C. For simplicity, we index into the dimensions of the hypercube using double indices: for every bitstring $J \in \{0, 1\}^n$, we write $J_{i,j}$ (where $j \in [n_i]$) as a shorthand for $J_{j+\sum_{l \in [i-1]} n_l}$.

Each vertex $J \in V(C)$ is assigned a corresponding vertex in the grid Γ as follows: For every grid dimension $i \in [d]$, we assign J a *color* j_i with

$$j_i := max(\{0\} \cup \{h \mid h \in [n_i], J_{i,h} = 1\}). \tag{13}$$

The tuple of colors $(j_1, \ldots, j_d)$ is the vertex in the grid that we associate with J. When we orient J, we orient each block i of dimensions of C independently, according to the corresponding simplex in dimension i in Γ that contains this vertex $(j_1, \ldots, j_d)$.

A single simplex of Γ with $n_i + 1$ directions is turned into an n_i-cube as shown in Figure 5. The vertex (0) of the simplex is mapped to the vertex 0^{n_i}, and all the other vertices of the grid are mapped to its neighbors. We call these $n_i + 1$ vertices the *grid-vertices*, all others are called *non-grid-vertices*. The paths of length one and two between these grid-vertices encode the orientation of the grid. This completely orients the grid-vertices. The rest of the cube is oriented according to the colors of the non-grid-vertices; edges between vertices of two different colors are oriented as the corresponding edge in the grid. Vertices of the same color are oriented in such a way that in the subcube corresponding to all vertices of the same color all the edges of the same dimension are oriented the same way. This orientation is uniquely defined since the unique grid-vertex of this color is already completely oriented.

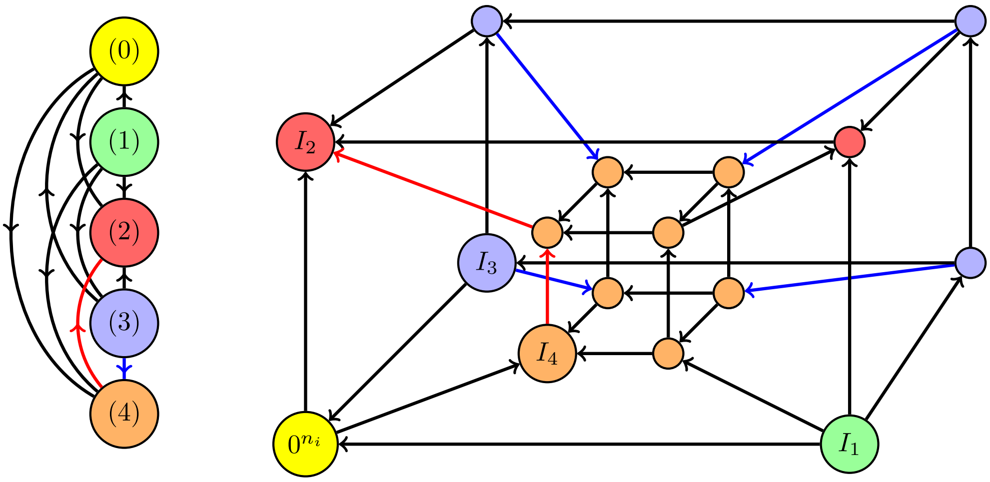

Figure 5 Example of placing grid-vertices (large) in the cube, coloring the non-grid-vertices (small) and orienting the edges. Each edge of the grid becomes a path of length two between two grid-vertices, see the red edges. Edges between vertices of different colors are oriented the same as in the grid, see the blue edges. Each subcube of the same color is oriented in a uniform way.

To formalize this construction for the complete grid, we need to define a secondary color j_i^* for each vertex $J \in V(C)$ and each grid dimension $i \in [d]$, which for all vertices with color $j_i \neq 0$ corresponds to the color of the neighboring vertex of J in dimension (i, j_i):

$$j_i^* := max(\{0\} \cup \{h \mid h \in [n_i] \setminus \{j_i\}, J_{i,h} = 1\}). \tag{14}$$

The orientation O of the cube C is now defined as follows:

$$O(J)_{i,h} := \begin{cases} \sigma((j_1, \ldots, j_i, \ldots, j_d), (j_1, \ldots, h, \ldots j_d)) \oplus J_{i,h} & \text{for } h < j_i, \\ \sigma((j_1, \ldots, j_i, \ldots, j_d), (j_1, \ldots, h, \ldots j_d)) & \text{for } h > j_i, \\ \sigma((j_1, \ldots, j_i, \ldots, j_d), (j_1, \ldots, j_i^*, \ldots j_d)) & \text{for } h = j_i. \end{cases} \tag{15}$$

▶ **Theorem 34.** *If σ is a unique sink orientation, then the orientation constructed by Equation (15) is a unique sink orientation. A circuit computing O can be computed in polynomial time. Given the sink of O, the sink of σ can be found in polynomial time.*

The proof can be found in the full version of the paper.

6 Open Questions

Total Search Problem Versions. All reductions we provided in this paper are between the promise problem versions of the involved problems. It may be interesting to also find reductions between the total search problem versions.

Missing Reductions. We were unable to show that finding an α-cut for arbitrary α is not more difficult than finding it for $\alpha_i \in \{1, |P_i|\}$. There might thus be a difference in the computational complexity of α-Ham-Sandwich and SWS-Colorful-Tangent. Similarly, in Grid-USO, we also do not know whether it is not more difficult to find a vertex with a specific refined index (which must also be unique) rather than searching for a sink. Note that in the case of α-Ham-Sandwich, it is at least known that the problem is contained in UEOPL. This is not known for α-Grid-USO.

Semantics of the Grid-USO to Cube-USO Reduction. On the levels of USOs, we do not know the exact operations that the reductions from P-GLCP to P-LCP and SWS-Colorful-Tangent to SWS-2P-Colorful-Tangent perform. It would be very interesting to analyze whether these reductions actually perform the same operation as the Grid-USO to Cube-USO reduction (Theorem 34), i.e., whether these reductions commute. It would also be interesting to study whether the Grid-USO to Cube-USO preserves realizability, i.e., whether if there exists a P-GLCP instance inducing a certain grid USO, there also exists a P-LCP instance inducing the resulting cube USO.

References

1 Michaela Borzechowski. The complexity class unique end of potential line. Master's thesis, Freie Universität Berlin, 2021.

2 Michaela Borzechowski and Wolfgang Mulzer. Unique sink orientations of grids is in unique end of potential line, 2022. doi:10.48550/arXiv.2209.02101.

3 Man-Kwun Chiu, Aruni Choudhary, and Wolfgang Mulzer. Computational Complexity of the α-Ham-Sandwich Problem. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020)*, volume 168 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 31:1–31:18, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ICALP.2020.31.

4 Richard W. Cottle and George B. Dantzig. A generalization of the linear complementarity problem. *Journal of Combinatorial Theory*, 8(1):79–90, 1970.

5 Richard W. Cottle, Jong-Shi Pang, and Richard E. Stone. *The Linear Complementarity Problem*. Society for Industrial and Applied Mathematics (SIAM), 2009.

6 Aniekan A. Ebiefung. Existence theory and Q-matrix characterization for the generalized linear complementarity problem. *Linear Algebra and its Applications*, 223-224:155–169, 1995. Honoring Miroslav Fiedler and Vlastimil Ptak. doi:10.1016/0024-3795(95)00091-5.

7 Aniekan A. Ebiefung. Existence theory and Q-matrix characterization for the generalized linear complementarity problem: Revisited. *Journal of Mathematical Analysis and Applications*, 329(2):1421–1429, 2007. doi:10.1016/j.jmaa.2006.07.036.

8 Herbert Edelsbrunner. Algorithms in combinatorial geometry. In A. Salomaa W. Brauer, G. Rozenberg, editor, *EATCS Monographs on Theoretical Computer Science*, volume 10. Springer Berlin Heidelberg, 1987. doi:10.1007/978-3-642-61568-9.

9 John Fearnley, Spencer Gordon, Ruta Mehta, and Rahul Savani. Unique end of potential line. *Journal of Computer and System Sciences*, 114:1–35, 2020. doi:10.1016/j.jcss.2020.05.007.

10 Bernd Gärtner, Walter D. Morris jr., and Leo Rüst. Unique sink orientations of grids. *Algorithmica*, 51(2):200–235, 2008. doi:10.1007/s00453-007-9090-x.

11 Bernd Gärtner and Leo Rüst. Simple stochastic games and P-matrix generalized linear complementarity problems. In Maciej Liśkiewicz and Rüdiger Reischuk, editors, *Fundamentals of Computation Theory*, pages 209–220. Springer Berlin Heidelberg, 2005. doi:10.1007/11537311_19.

12 Bernd Gärtner and Ingo Schurr. Linear programming and unique sink orientations. In *Proc. 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 749–757, 2006. doi:10.5555/1109557.1109639.

13 Bernd Gärtner and Antonis Thomas. The complexity of recognizing unique sink orientations. In Ernst W. Mayr and Nicolas Ollinger, editors, *32nd International Symposium on Theoretical Aspects of Computer Science (STACS 2015)*, volume 30 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 341–353, Dagstuhl, Germany, 2015. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.STACS.2015.341.

14 Mika Göös, Alexandros Hollender, Siddhartha Jain, Gilbert Maystre, William Pires, Robert Robere, and Ran Tao. Separations in proof complexity and tfnp. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1150–1161, 2022. doi:10.1109/FOCS54457.2022.00111.

15 George J. Habetler and Bohdan P. Szanc. Existence and uniqueness of solutions for the generalized linear complementarity problem. *Journal of Optimization Theory and Applications*, 84(1):103–116, January 1995. doi:10.1007/BF02191738.

16 Thomas Dueholm Hansen and Rasmus Ibsen-Jensen. The complexity of interior point methods for solving discounted turn-based stochastic games. In *Proc. of CiE*, volume 7921, pages 252–262. Springer, 2013.

17 Marcin Jurdzinski and Rahul Savani. A simple P-matrix linear complementarity problem for discounted games. In *Proc. of Conference on Computability in Europe (CiE)*, pages 283–293, 2008. doi:10.1007/978-3-540-69407-6_32.

18 Nimrod Megiddo. A note on the complexity of P-matrix LCP and computing an equilibrium. Technical report, IBM Research, 1988.

19 Katta G. Murty. On the number of solutions to the complementarity problem and spanning properties of complementary cones. *Linear Algebra and its Applications*, 5(1):65–108, 1972. doi:10.1016/0024-3795(72)90019-5.

20 Katta G. Murty and Feng-Tien Yu. *Linear complementarity, linear and nonlinear programming.* Heldermann, 1988.

21 William Steiger and Jihui Zhao. Generalized ham-sandwich cuts. *Discrete & Computational Geometry*, 44(3):535–545, 2010. doi:10.1007/s00454-009-9225-8.

22 Alan Stickney and Layne Watson. Digraph models of Bard-type algorithms for the linear complementarity problem. *Mathematics of Operations Research*, 3(4):322–333, 1978. URL: https://www.jstor.org/stable/3689630.

23 Ola Svensson and Sergei G. Vorobyov. Linear complementarity and P-matrices for stochastic games. In Irina B. Virbitskaite and Andrei Voronkov, editors, *Perspectives of Systems Informatics, 6th International Andrei Ershov Memorial Conference, PSI 2006, Novosibirsk, Russia, June 27-30, 2006. Revised Papers*, volume 4378, pages 409–423, 2006.

24 Tibor Szabó and Emo Welzl. Unique sink orientations of cubes. In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, pages 547–555, 2001. doi:10.1109/SFCS.2001.959931.