A Universal Construction for Unique Sink Orientations*

Michaela Borzechowski[1], Joseph Doolittle[2], and Simon Weber[3]

1 **Institut für Informatik, Freie Universität Berlin, Germany**
michaela.borzechowski@fu-berlin.de
2 **Institut für Geometrie, Technische Universität Graz, Austria**
nerdyjoe@gmail.com
3 **Department of Computer Science, ETH Zürich, Switzerland**
simon.weber@inf.ethz.ch

Abstract

Unique Sink Orientations (USOs) of cubes capture the combinatorial structure of many essential algebraic and geometric problems. It is crucial to have systematic constructions of USOs for various structural and algorithmic questions, including enumeration of USOs and algorithm analysis. While some construction methods for USOs already exist, each one of them has some significant downside. Inspired by cube tilings of space, we expand upon existing techniques to develop *generalized rewriting rules* for USOs. These rewriting rules are a new construction framework which can be applied to all USOs. Furthermore, they can generate every USO using only USOs of lower dimension.

Related Version arXiv:2211.06072

1 Introduction

A Unique Sink Orientation (USO) is an orientation of the hypercube graph, such that every non-empty face (subcube) has a unique sink. See Figure 1 for an example. USOs were first defined by Szabó and Welzl in 2001 [21]. They encode the combinatorial structure of several problems, for examples the P-matrix linear complementarity problem, linear programming, and many more [8, 11, 13, 17, 19]. USOs have also attracted attention as purely combinatorial objects, with interest in structural and algorithmic directions [2, 5, 6, 7, 9, 10, 16, 18].

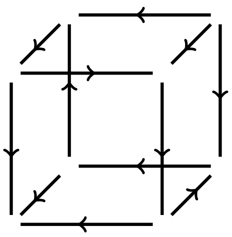
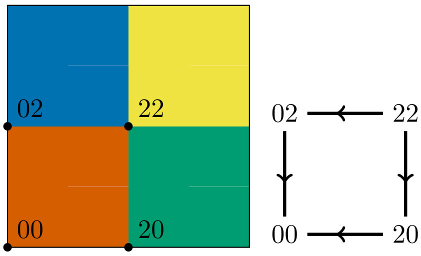


Figure 1 A Unique Sink Orientation of the 3-cube.

On the structural side, enumerating and sampling USOs are important unsolved challenges. The main issues are that USOs are hard to recognize [9] and while they are very sparse

* Michaela Borzechowski is supported by the German Research Foundation DFG within the Research Training Group GRK 2434 *Facets of Complexity*. Joseph Doolittle was supported by the Austrian Science Fund FWF, grant P 33278. Simon Weber is supported by the Swiss National Science Foundation under project no. 204320.

40th European Workshop on Computational Geometry, Ioannina, Greece, March 13–15, 2024.
This is an extended abstract of a presentation given at EuroCG'24. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.



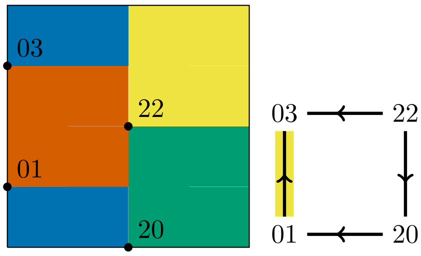


Figure 2 Two examples of $4\mathbb{Z}^2$-periodic tilings of $\mathbb{R}^2$ with their corresponding USOs.

among all cube orientations, there still exists a doubly exponential number (in terms of the cube dimension) of them [16].

A few systematic construction methods for USOs are known: the *product construction* [18], *inherited orientations*, *flipping* all edges of one dimension at once [21] and flipping equivalence classes of edges (called *phases*) that preserves the USO condition [17]. Of all these methods, only the product construction is able to increase the dimension of the USO, and only phase flips are capable of theoretically generating all USOs of a fixed dimension — however no systematic strategy for this is known and the mixing rate of the natural Markov chain based on phase flips remains unknown too. We discuss the existing construction methods more in-depth in the full version of the paper [1].

Results. Based on a remark of Schurr [17], we prove a one-to-one correspondence between so-called $4\mathbb{Z}^k$*-periodic tilings* and k-dimensional USOs. Representations of these tilings can be manipulated in the language of string rewriting, in particular this technique was used to disprove Keller's conjecture on unit cube tilings [12, 14, 15]. We generalize these construction techniques and translate them into the language of USOs. Our generalization provides a very general framework with many parameters, and every choice of parameters is a new construction which can be applied to any USO. Given both 1-dimensional USOs and a specific 2-dimensional USO (the *bow*), repeated application of constructions from our framework can be used to generate all USOs of dimension $k \geq 1$, we thus call our framework *universal*.

In the full version [1], we additionally show that we can realize all existing constructions that are applicable to all USOs as special cases of our framework. We also point out another special case of our construction as a new dimension-preserving modification, the *partial swap*.

2 Unit Cube Tilings and USOs

In a $4\mathbb{Z}^k$*-periodic tiling* [20] we tile the k-cube C of side length 4 by 2^k integer-grid aligned k-cubes (*tiles*) of side length 2, such that (i) every point of K is contained in at least one tile, and (ii) if a point is contained in multiple tiles, it lies on the boundary of all such tiles. These tiles may wrap around the boundary of C, exiting on one side and entering again on the opposite side (see Figure 2). This then defines a periodic tiling of $\mathbb{R}^k$, as infinitely repeating the tiling of C fills $\mathbb{R}^k$.

It was shown by Szabó [20] that Keller's conjecture [12] — a conjecture claiming that all cube tilings of $\mathbb{R}^k$ contain two tiles that share a facet (so-called *twins*) — can be decided by only considering these $4\mathbb{Z}^k$-periodic tilings. Note that Keller's conjecture has since been resolved and is known to hold up to dimension 7 [3] and fail for dimensions 8 and above [15].

A $4\mathbb{Z}^k$-periodic tiling can be described by a set of 2^k strings in $\{0, 1, 2, 3\}^k$, each string describing the coordinates of the bottom left corner of one tile. A set of strings describes a valid $4\mathbb{Z}^k$-periodic tiling if and only if for every pair of strings, there is at least one coordinate in which the integer entries differ by exactly 2 [4, 14].

Schurr [17] briefly mentioned a bijection between $4\mathbb{Z}^k$-periodic tilings and USOs of the k-cube. We want to make this more explicit.

A set of strings describing a tiling also describes an orientation as follows. Each string $s \in \{0, 1, 2, 3\}^k$ describes one vertex of the k-cube and the orientation of its incident edges:

- If $s_i = 0$ or $s_i = 1$, then s is in the lower i-facet of the cube.
- If $s_i = 2$ or $s_i = 3$, then s is in the upper i-facet of the cube.
- If $s_i = 0$ or $s_i = 2$, then the edge from s in dimension i is *downwards* oriented.
- If $s_i = 1$ or $s_i = 3$, then the edge from s in dimension i is *upwards* oriented.

In other words, the two bits of the binary encoding of s_i encode the location and the orientation of the vertex in dimension i, respectively. Equivalently, we can also retrieve a $4\mathbb{Z}^k$-periodic tiling from a k-cube and its orientation. See Figure 2 for an example and note that we always mark upwards edges by a yellow background.

It remains to show that the set of strings describes an USO if and only if the tiling it describes is valid. To see this, we use the characterization of USOs by the Szabó-Welzl condition [21]: An orientation is USO if and only if for each pair of distinct vertices v, w, there exists a dimension i in the subcube they span such that they both have the same up-map in that dimension, i.e., both have an upwards i-edge or both have a downwards i-edge. This corresponds directly to the condition that any pair of strings s, t differs by exactly 2 in the i'th coordinate:

- The difference of two strings s_i and t_i is greater than 1 if and only if the vertices s and t lie in different i-facets, and thus i is a dimension of the subcube these vertices span.
- The difference of two strings s_i and t_i is even if and only if the vertices s and t agree on the orientation of their incident edge in dimension i.

Combining these two conditions yields that s_i and t_i differ by exactly 2, as desired. Thus, the Szabó-Welzl condition is equivalent to the condition for tiling validity.

3 Rewriting Rules

The first disproof of Keller's conjecture by Lagarias and Shor [14] used string rewriting to create higher dimensional tilings from lower dimensional tilings. In this section, we generalize their technique to operations that can be applied to all USOs, so-called *generalized rewriting rules*. We first define *simple rewriting rules*, which are used to rewrite a single digit in each string of an USO. To define such a rule we need four lists, which specify what to replace each possible digit with. From Lagarias and Shor's approach we extract the conditions necessary to hold for these four lists, such that the result is again an USO.

▶ **Definition 3.1.** Let $S^{(0)}$, $S^{(1)}$, $S^{(2)}$, $S^{(3)} \subseteq \{0, 1, 2, 3\}^d$ with the properties that

(i) $\left(S^{(0)} \cup S^{(2)}\right)$ defines a d-dimensional USO (a $4\mathbb{Z}^d$-periodic tiling) and $S^{(0)} \cap S^{(2)} = \emptyset$, and
(ii) $\left(S^{(1)} \cup S^{(3)}\right)$ defines a d-dimensional USO (a $4\mathbb{Z}^d$-periodic tiling) and $S^{(1)} \cap S^{(3)} = \emptyset$.

The sets $\left(S^{(0)}, S^{(1)}, S^{(2)}, S^{(3)}\right)$ define a *simple rewriting rule*. We define the function S_h to *apply* this simple rewriting rule to a k-dimensional input USO K on dimension $h \in [k]$. It maps subsets of $\{0, 1, 2, 3\}^k$ to subsets of $\{0, 1, 2, 3\}^{k+d-1}$. Applying the simple rewriting rule to a single vertex of the input USO is defined as follows:

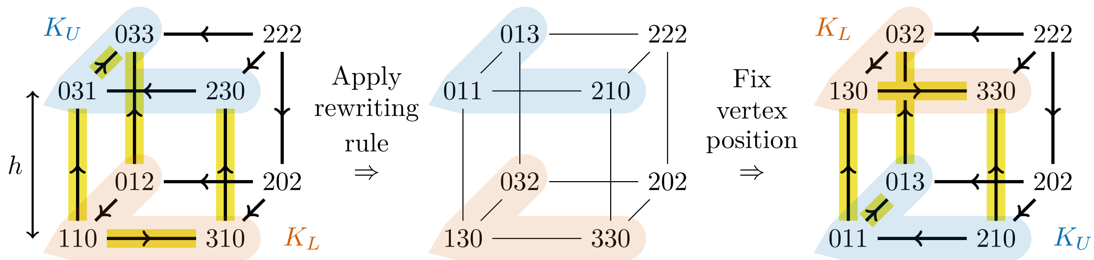
$$S_h(v) := \left\{ v_1, \ldots, v_{h-1}, s_1, \ldots, s_d, v_{h+1}, \ldots, v_k \mid s \in S^{(v_h)} \right\}.$$

We write $S_h(K)$ (for a set $K \subseteq \{0, 1, 2, 3\}^k$) for the union of the outputs of S_h when applied to all elements of K, i.e., $S_h(K) := \bigcup_{v \in K} S_h(v)$.

For each string $v \in K$, $S_h(v)$ produces a set of strings which depends on the value of the entry v_h. For each element s of $S^{(v_h)}$, a string is generated by replacing the entry v_h with s. The single vertex v is thus mapped to $|S^{(v_h)}|$ vertices. Note that some of the sets $S^{(\cdot)}$ may be empty. In this case, when $|S^{(v_h)}| = 0$, no strings are produced from v.

A particularly interesting operation on USOs is the following 1-dimensional rewriting rule, which we call the *partial swap*: $(S^{(0)} = \{0\}, S^{(1)} = \{3\}, S^{(2)} = \{2\}, S^{(3)} = \{1\})$. We analyze this rewriting rule in more detail in the full version of this paper [1].

▶ **Example 3.2.** To the USO $K = \{110, 310, 012, 202, 031, 230, 033, 222\}$ we apply the partial swap in dimension $h = 2$, i.e., we rewrite the second coordinate of each vertex. In the resulting USO, the subgraphs K_L and K_U swapped places.



In the full version [1] we show the following lemma, i.e., that simple rewriting rules are correct USO constructions.

▶ **Lemma 3.3.** *Applying any rewriting rule S_h to an USO K of strings in $\{0,1,2,3\}^k$ results in a valid USO $S_h(K)$ of strings in $\{0,1,2,3\}^{k+d-1}$.*

3.1 Generalized Rewriting Rules

To arrive at their counterexamples to Keller's conjecture, Lagarias and Shor used a more general rewriting technique [14]. They do not only use the four digits $0, 1, 2, 3$ in their input tiling, but also "alternative digits" $0'$ and $1'$, which only differ from their normal counterparts for the purposes of the rewriting, but specify the same coordinate for the tiling. We generalize our construction based on this idea, by letting the input USO specify one of i labels at each vertex.

For the full generality of our rewriting framework, the sets $S^{(m)}$ are replaced by a list of i sets $S^{(m)}_{1,\dots,i}$ each, where the indices correspond to the possible labels attached to the vertices of the input USO. All the compatibility requirements are appropriately expanded.

▶ **Definition 3.4.** Let $d, i \in \mathbb{N}$, $S^{(0)}_{1,\dots,i}, S^{(1)}_{1,\dots,i}, S^{(2)}_{1,\dots,i}, S^{(3)}_{1,\dots i} \subseteq \{0,1,2,3\}^d$ where

(i) $\left(S^{(0)}_j \cup S^{(2)}_{j'}\right)$ defines a d-dimensional USO and $S^{(0)}_j \cap S^{(2)}_{j'} = \emptyset$ for all pairs $j, j' \in [i]$, and

(ii) $\left(S^{(1)}_j \cup S^{(3)}_{j'}\right)$ defines a d-dimensional USO and $S^{(1)}_j \cap S^{(3)}_{j'} = \emptyset$ for all pairs $j, j' \in [i]$.

The sets $\left(S^{(0)}_{1,\dots,i}, S^{(1)}_{1,\dots,i}, S^{(2)}_{1,\dots,i}, S^{(3)}_{1,\dots i}\right)$ define a *generalized rewriting rule.* We define the function T_h to *apply* this generalized rewriting rule to a k-dimensional input USO K on dimension $h \in [k]$. It maps subsets of $\{0,1,2,3\}^k \times [i]$ to subsets of $\{0,1,2,3\}^{k+d-1}$. Applying the generalized rewriting rule to a single vertex v labeled j of the input USO is defined as:

$$T_h(v, j) \coloneqq \left\{v_1, \dots, v_{h-1}, t_1, \dots, t_d, v_{h+1}, \dots, v_k \mid t \in S_j^{(v_h)}\right\}.$$

We extend the function T_h from single inputs to sets similarly to Definition 3.1.

Note that we can use duplicate sets $S_j^{(m)} = S_{j'}^{(m)}$ in case we want to have fewer than i sets for some $m \in \{0,1,2,3\}$. Lemma 3.3 holds also for generalized rewriting rules, with the proof applying *mutatis mutandis* since Definition 3.4 provides the necessary disjointness and coherence conditions:

▶ **Lemma 3.5.** *Let K be an USO of strings in $\{0,1,2,3\}^k$, and $L : K \to [i]$ an additional labelling function. Then $T_h(K,L)$ is an USO of strings in $\{0,1,2,3\}^{k+d-1}$.*

Intuitively, the effect of a generalized rewriting rule can be described as follows. Given an input USO K and a rewriting rule, we replace edges of dimension h. For simplicity, we focus on a single 2-face containing two edges of this dimension h: $\{v_1, v_2\}$ and $\{w_1, w_2\}$. The rewriting rule replaces those two h-edges by the d-dimensional USOs $T_h(\{v_1, v_2\}, L)$ and $T_h(\{w_1, w_2\}, L)$. Instead of the edges $\{v_1, w_1\}$ and $\{v_2, w_2\}$, there are now 2^d new edges between $T_h(\{v_1, v_2\}, L)$ and $T_h(\{w_1, w_2\}, L)$ as can be seen in Figure 3.

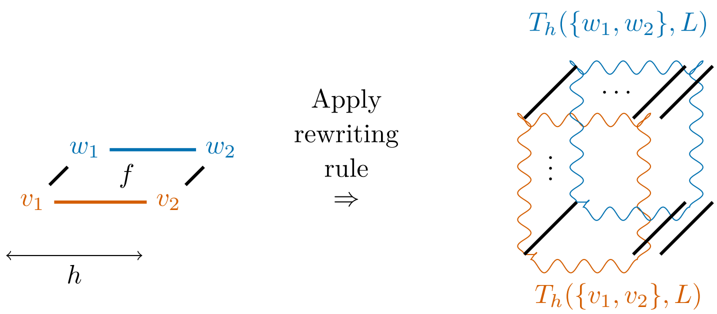


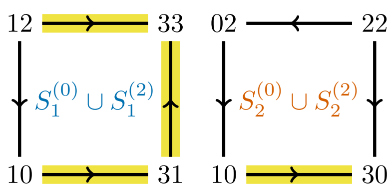
Figure 3 Sketch of the effect of the generalized rewriting rule on the 2-face f.

It holds that if $\{v_1, v_2\}$ is a downwards edge, $T_h(\{v_1, v_2\}, L) = S^{(0)}_{L(v_1)} \cup S^{(2)}_{L(v_2)}$. If $\{v_1, v_2\}$ is an upwards edge, $T_h(\{v_1, v_2\}, L) = S^{(1)}_{L(v_1)} \cup S^{(3)}_{L(v_2)}$. Analogously, the edge $\{w_1, w_2\}$ is replaced by the respective union of sets. In either case, this is guaranteed to be an USO by the conditions (i) and (ii) in Definition 3.4.

The edges between the USOs $T_h(\{v_1, v_2\}, L)$ and $T_h(\{w_1, w_2\}, L)$ copy their orientation either from the edge $\{v_1, w_1\}$ or from the edge $\{v_2, w_2\}$. Which of these edges is copied depends on whether the resulting edge is incident to a vertex in $T_h(\{v_1\}, L)$ or $T_h(\{v_2\}, L)$. This depends on how the sets $S_j^{(0)} \cup S_{j'}^{(2)}$ (and $S_j^{(1)} \cup S_{j'}^{(3)}$ respectively) are split into their parts, i.e., which vertices of the d-USOs they describe lie in which set. Note that these unions are split the same way, no matter j and j'.

▶ **Example 3.6.** The following is a generalized rewriting rule for $d = 2$.

$S_1^{(0)} = S_1^{(1)} = \{10\}$
$S_1^{(2)} = S_1^{(3)} = \{12, 33, 31\}$
$S_2^{(0)} = S_2^{(1)} = \{10\}$
$S_2^{(2)} = S_2^{(3)} = \{02, 22, 30\}$



We apply this rewriting rule to dimension $h = 1$ of the *bow* $K = \{01, 20, 03, 22\}$ with the labeling function $L(01) = 2, L(03) = 1, L(20) = 2$ and $L(22) = 1$. This means, we replace the first coordinate of each vertex. The result is $T_1(K, L) = \{101, 300, 020, 220, 103, 122, 312, 332\}$.

4 Universality of the Construction

Our construction is universal, meaning it is sufficiently general to generate all USOs, using only the 1-dimensional USOs, and the 2-dimensional “bow” as base cases.

▶ **Theorem 4.1** (Universality). *Starting with the set of both* 1*-dimensional USOs* $\{0, 2\}$ *and* $\{1, 3\}$*, one can generate every USO of dimension* $n \geq 1$ *by repeated application of generalized rewriting rules to the bow* $\{01, 20, 03, 22\}$*, where every set* $S_j^{(m)}$ *used in a rewriting rule is a subset of some set of strings describing an USO already obtained before.*

To prove this theorem, we show the following lemma in the full version [1], which states that for any n-dimensional USO there exists a generalized rewriting rule which creates this USO by only using $(n-1)$-dimensional USOs and the bow. From Lemma 4.2, Theorem 4.1 follows as a direct consequence.

▶ **Lemma 4.2.** *Let* K *be an* n*-dimensional USO. Then there exists a generalized rewriting rule* $\left(S_{1,2}^{(0)}, S_{1,2}^{(1)}, S_{1,2}^{(2)}, S_{1,2}^{(3)}\right)$*, where each* $S_j^{(m)}$ *is a (partial)* $(n-1)$*-dimensional USO, and*

$$K = T_1(bow = \{01, 20, 03, 22\}, L), \text{ for } L(01) = 2, L(03) = 1, L(20) = 2, L(22) = 1.$$

Proof (sketch). With the input labelling L, each set $S_{1,2}^{(0)}$ and $S_{1,2}^{(2)}$ is used to rewrite exactly one string of the bow. Furthermore, each of these strings has a unique digit in the second coordinate. We ignore the unused sets $S_{1,2}^{(1)}$ and $S_{1,2}^{(3)}$, and define $S_{1,2}^{(0)}$ and $S_{1,2}^{(2)}$ by simply splitting the strings of our target USO K depending on their last digit (and discarding that last digit), i.e., depending on the two n-facets of K and the edges between them:

- $S_1^{(0)}$: rewrites 03, contains vertices of the *upper* n-facet of K with an *upwards* n-edge.
- $S_2^{(0)}$: rewrites 01, contains vertices of the *lower* n-facet of K with an *upwards* n-edge.
- $S_1^{(2)}$: rewrites 22, contains vertices of the *upper* n-facet of K with an *downwards* n-edge.
- $S_2^{(2)}$: rewrites 20, contains vertices of the *lower* n-facet of K with an *downwards* n-edge.

Thus, when applied to the bow, we end up with exactly our target USO. It remains to prove that these sets form a valid generalized rewriting rule. For this, we can show that $S_1^{(0)} \cup S_1^{(2)}$ and $S_2^{(0)} \cup S_2^{(2)}$ are the two n-facets of K, while the other set combinations are the n-facets of K after applying a *partial swap*. For details we refer to the full version [1]. ◀

5 Future Work

Unfortunately, our rewriting rules exhibit a similar weakness to the phase flips of Schurr. While they are universal and each step in the universality proof is very systematic, our construction does not yet provide a suitable way to enumerate all USOs. This is in part because checking conditions (i) and (ii) of Definition 3.4 is computationally expensive. As future work, we suggest searching for more interesting special cases of (generalized) rewriting rules, or for other more systematic ways to enumerate USOs. Our framework could also be further generalized to rewrite multiple dimensions at once, similar to the approach taken by Mackey [15] to find the 8-dimensional counterexample to Keller's conjecture.

References

1 Michaela Borzechowski, Joseph Doolittle, and Simon Weber. A universal construction for unique sink orientations, 2022. doi:10.48550/ARXIV.2211.06072.

2 Vitor Bosshard and Bernd Gärtner. Pseudo unique sink orientations, 2017. doi:10.48550/ARXIV.1704.08481.

3 Joshua Brakensiek, Marijn Heule, John Mackey, and David Narváez. The resolution of Keller's conjecture. *Journal of Automated Reasoning*, 66(3):277–300, Aug 2022. doi:10.1007/s10817-022-09623-5.

4 Keresztély Corrádi and Sándor Szabó. A combinatorial approach for Keller's conjecture. *Periodica Mathematica Hungarica*, 21(2):95–100, Jun 1990. doi:10.1007/BF01946848.

5 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.

6 Bernd Gärtner. The random-facet simplex algorithm on combinatorial cubes. *Random Structures & Algorithms*, 20(3):353–381, 2002. doi:10.1002/rsa.10034.

7 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.

8 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.

9 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.

10 Bernd Gärtner and Antonis Thomas. The niceness of unique sink orientations. In Klaus Jansen, Claire Mathieu, José D. P. Rolim, and Chris Umans, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX-/RANDOM 2016)*, volume 60 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 30:1–30:14, Dagstuhl, Germany, 2016. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.APPROX-RANDOM.2016.30.

11 Bernd Gärtner and Emo Welzl. Explicit and implicit enforcing - randomized optimization. In *Computational Discrete Mathematics: Advanced Lectures*, pages 25–46. Springer Berlin Heidelberg, 2001. doi:10.1007/3-540-45506-X_3.

12 Ott-Heinrich Keller. Über die lückenlose erfüllung des raumes mit würfeln. *Journal für die reine und angewandte Mathematik*, 1930(163):231–248, 1930. doi:10.1515/crll.1930.163.231.

13 Lorenz Klaus. *A fresh look at the complexity of pivoting in linear complementarity*. PhD thesis, ETH Zürich, 2012. doi:10.3929/ethz-a-007604201.

14 Jeffrey C. Lagarias and Peter W. Shor. Keller's cube-tiling conjecture is false in high dimensions. *Bulletin of the American Mathematical Society*, 27(2):279–283, 1992. doi:10.1090/S0273-0979-1992-00318-X.

15 John Mackey. A cube tiling of dimension eight with no facesharing. *Discrete & Computational Geometry*, 28(2):275–279, Aug 2002. doi:10.1007/s00454-002-2801-9.

16 Jiří Matoušek. The number of unique-sink orientations of the hypercube. *Combinatorica*, 26(1):91–99, 2006. doi:10.1007/s00493-006-0007-0.

17 Ingo Schurr. *Unique Sink Orientations of Cubes*. PhD thesis, ETH Zürich, 2004. doi:10.3929/ethz-a-004844278.

18 Ingo Schurr and Tibor Szabó. Finding the sink takes some time: An almost quadratic lower bound for finding the sink of unique sink oriented cubes. *Discrete & Computational Geometry*, 31(4):627–642, 2004. doi:10.1007/s00454-003-0813-8.

19 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.

20 Sándor Szabó. Cube tilings as contributions of algebra to geometry. *Contributions to Algebra and Geometry*, 34(1):63–75, 1993.

21 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.