Discrete Applied Mathematics 160 (2012) 2104–2115

Contents lists available at SciVerse ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

On the existence of Hamiltonian paths for history based pivot rules on acyclic unique sink orientations of hypercubes

Yoshikazu Aoshima [a,b], David Avis [c,d], Theresa Deering [d], Yoshitake Matsumoto [a,e], Sonoko Moriyama [f,*]

[a] Department of Computer Science Graduate School of Information Science and Technology, The University of Tokyo, Japan
[b] ERATO-SORST Quantum Computation and Information Project, JST, Japan
[c] School of Informatics, Kyoto University, Japan
[d] School of Computer Science, McGill University, Canada
[e] Google Japan Inc., Japan
[f] Graduate School of Information Sciences, Tohoku University, Japan

ARTICLE INFO

Article history:
Received 10 March 2011
Received in revised form 9 May 2012
Accepted 28 May 2012
Available online 20 June 2012

Keywords:
Simplex method
Polytopal digraphs
Pivot rule
Klee–Minty cube
Depth first search

ABSTRACT

An acyclic USO on a hypercube is formed by directing its edges in such a way that the digraph is acyclic and each face of the hypercube has a unique sink and a unique source. A path to the global sink of an acyclic USO can be modelled as pivoting in a unit hypercube of the same dimension with an abstract objective function, and vice versa. In such a way, Zadeh's 'least entered rule' and other history based pivot rules can be applied to the problem of finding the global sink of an acyclic USO. In this paper we present some theoretical and empirical results on the existence of acyclic USOs for which the various history based pivot rules can be made to follow a Hamiltonian path. In particular, we develop an algorithm that can enumerate all such paths up to dimension 6 using efficient pruning techniques. We show that Zadeh's original rule admits Hamiltonian paths up to dimension 9 at least, and prove that most of the other rules do not for all dimensions greater than 5.

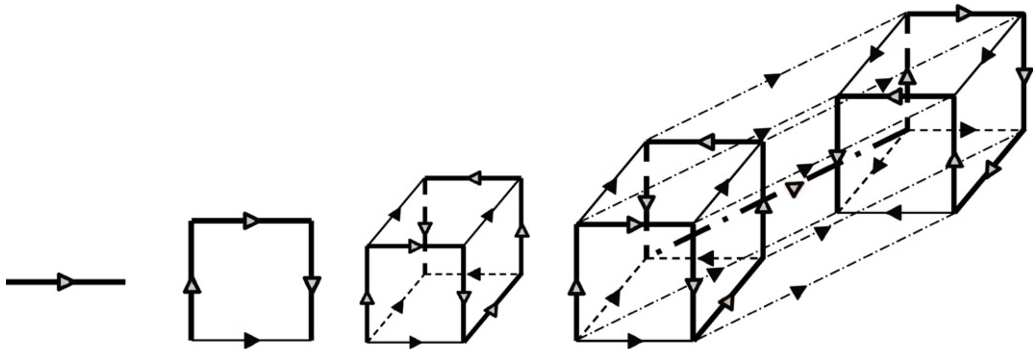
1. Introduction

It is now over 30 years since Khachian showed that linear programming problems can be solved in polynomial time [12]. His ellipsoid algorithm and subsequent interior point methods are not, however, strongly polynomial time algorithms and no such algorithms are known. Pivoting algorithms, such as Dantzig's simplex method [5] still offer the possibility of being strongly polynomial. One reason for this is that pivoting algorithms follow a path on the graph defined by the skeleton of a polyhedron, and it is widely believed that the diameter of this graph is polynomially bounded in the dimensions of the linear program.

In fact, Hirsch conjectured that the diameter of any d-dimensional polytope with n facets, where $n > d \geq 2$, is less than or equal to $n-d$. Very recently Santos has found that this conjecture is false, by exhibiting a polytope with dimension $d = 43$ and $n = 86$ facets with diameter equal to $n - d + 1$ [17]. Nevertheless, the belief that the diameter is polynomial is still strong. The subexponential bounds of Kalai [11] and Matoušek et al. [15] also give further grounds for hope. These papers use randomized pivot selection rules, and no deterministic rules that achieve these subexponential bounds are known.

* Corresponding author. Tel.: +81 358414102; fax: +81 358006933.
E-mail addresses: y-aoshima@is.s.u-tokyo.ac.jp (Y. Aoshima), avis@cs.mcgill.ca (D. Avis), theresa.deering@mail.mcgill.ca (T. Deering), ymatsu@is.s.u-tokyo.ac.jp (Y. Matsumoto), moriso@dais.is.tohoku.ac.jp (S. Moriyama).

0166-218X/$ – see front matter © 2012 Published by Elsevier B.V.
doi:10.1016/j.dam.2012.05.023



(a) Dimension 1 or 2. (b) Dimension 3. (c) Dimension 4.

Fig. 1. Klee–Minty cube.

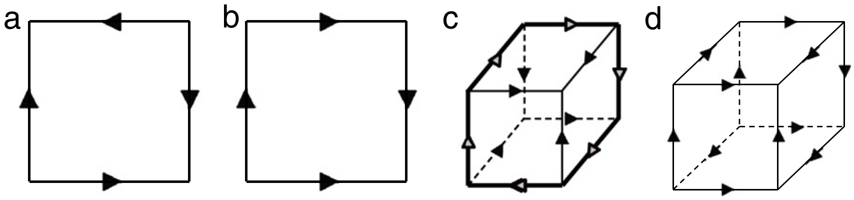


Fig. 2. (a) is a non-USO, (b) and (d) are AUSO cubes, (c) is a USO cube which has a cycle.

The simplex method is a family of algorithms, with each member of the family being determined by a pivot selection rule. In practice, Dantzig's original rule works extremely well. However, Klee and Minty [13] constructed a case where the simplex method using this rule follows an exponential length path on a family of suitably stretched hypercubes, since called *Klee–Minty cubes*. In fact it visits every vertex, that is, Hamiltonian path, of the hypercube (see Fig. 1). Subsequent research demonstrated that many other pivot rules take exponential time on variants of the Klee–Minty examples. Such pivot rules include the maximum improvement rule [10] and Bland's rule [2].

There are still some pivot rules for which the behaviour of the simplex method is unknown. A particularly interesting set of these rules are the history based pivot rules, of which the best known is the *least entered rule* proposed by Zadeh in a 1980 Stanford University Technical Report, that was recently reprinted [20]. Very recently, 30 years after it originally appeared, Friedmann showed that this rule requires at least sub-exponential time in the worst case such as $2^{\Omega(\sqrt{d})}$ [7]. A non-trivial upper bound on Zadeh's rule is still unknown.

To motivate the least entered rule, Zadeh pointed out a characteristic of Klee–Minty examples: some variables pivot very few times and other variables pivot an exponential number of times. Zadeh's pivot rule avoids this behaviour by making each variable pivot, roughly, the same number of times. In this way, it behaves similarly to the random pivot selection rules mentioned above. In Section 3 we show that if Zadeh's rule follows a Hamiltonian path on a hypercube, then indeed, each variable must pivot an exponential number of times in the dimension d of the cube.

Zadeh's rule differs from the former pivot rules in that it uses information from the entire pivot history up to that point. Such pivot rules are called *history based pivot rules*. Besides Zadeh's rule, these include the least-recently basic rule [4], the least-recently considered rule [4], the least-recently entered rule [6], and the least iterations in the basis rule [3]. We remark that for each of these pivot rules, there exists no known exponential lower bound.

In this paper, we study the behaviour of history based pivot rules on an abstraction of linear programming known as acyclic unique sink orientations (AUSOs) of hypercubes, that were introduced by Szabó and Welzl [19]. These are orientations of the hypercube so that the resulting directed graph is acyclic, and each face of each dimension has a unique sink and a unique source, see Fig. 2. The goal is to find the unique sink of the hypercube.

As noted in [19], various optimization problems can be solved using this model. The direction on an edge of the hypercube corresponds to an increase in the value of an abstract objective function defined on the vertices of the hypercube. The concept of abstract objective functions was first introduced by Adler and Saigal [1]. The related concept of completely unimodal numberings was introduced by Williamson Hoke [8]. Although AUSOs need not correspond to actual polytopes and objective functions, the notions of linear programming, such as bases and pivots, are readily available. Therefore we obtain an abstract model on which to observe the behaviour of various history based pivot rules.

AUSOs on d-dimensional hypercubes have a structure that makes for convenient notation and terminology. Each vertex is labelled $0, \ldots, (2^d - 1)$ such that the binary representation of adjacent vertices' labels differ by exactly one bit. Each edge has a *direction* and an *orientation*. The direction is given by a number $1, \ldots, d$ indicating which bit is different between the two endpoints (counted right-to-left). The orientation is given by a positive sign (+) if the differing bit is 0 at the edge's tail

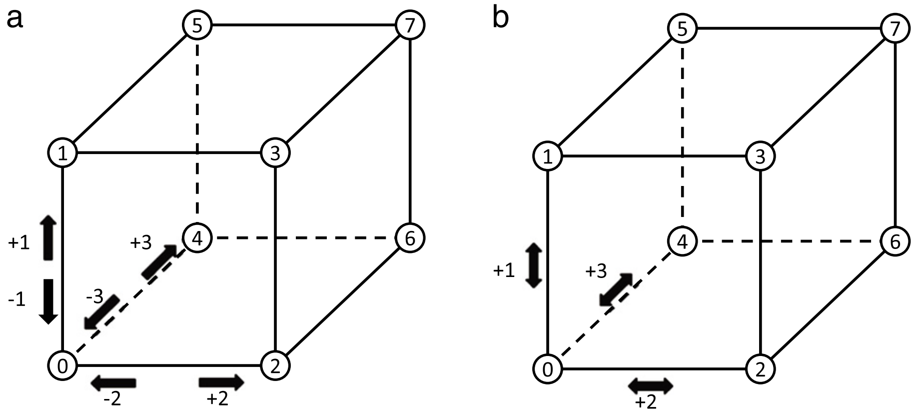


Fig. 3. Signed and unsigned direction on a 3-dimensional cube.

and 1 at its head, and it is given by a negative sign ($-$) otherwise. We will use the terms *direction* to denote which bit is to be changed and *signed direction* which also specifies the orientation. For emphasis, to specify the direction without sign we use the term *unsigned direction* (see Fig. 3). For some pivot rules only the unsigned direction is important whereas for others both the orientation and direction are important.

Although AUSOs do not necessarily correspond to LP digraphs, the above vertex labelling can be used to model moving along a path on an AUSO as pivoting in the dictionary $x_{d+i} = 1 - x_i$ for $i = 1, \ldots, d$. A pivot $+i$ corresponds to a pivot where x_i enters the basis and x_{d+i} leaves, and a pivot $-i$ corresponds to a pivot where x_{d+i} enters the basis and x_i leaves. This allows the AUSO to inherit various pivoting strategies that are defined in terms of LPs.

The Klee–Minty examples mentioned above can be modelled as AUSOs. In fact, they show that Dantzig's original pivot rule for the simplex method leads to a Hamiltonian path on an associated AUSO for each dimension d. As mentioned, similar results have been found for other pivot rules. In this paper, we investigate whether history based pivot rules can lead to Hamiltonian paths on AUSOs.

Our focus on Hamiltonian paths has the following motivations. First, if such paths exist for a given pivot rule they are obviously the worst case examples. Secondly, the number of AUSOs is extremely large. Stickney showed there are 19 in 3 dimensions [18], Moriyama's program showed there are 12 640 in 4 dimensions [16], and Matoušek [14] has shown that there are at least 2^{2^d} AUSOs in d-dimensions. So just listing their degree sequences when $d = 6$ requires at least 2^{70} steps. Except for extremely low dimensions, it is therefore not possible to construct all acyclic USOs. However searching for all acyclic USOs which contain a Hamiltonian path greatly reduces the search space. This is due to a remarkable indegree characterization due to Williamson-Hoke discussed in Section 3. We are able to exploit this property 'on the fly' to eliminate early prefixes of Hamiltonian paths that cannot be completed to an acyclic USO. This is because the final indegree of each vertex is known as soon as it enters the path. The enumeration enabled us to see that in fact most rules do not follow Hamiltonian paths, a fact we were then able to prove. Of course proving that a pivot rule cannot follow a Hamiltonian path does not say anything about the existence or not of other exponential length paths. However searching for these is likely to be significantly more difficult.

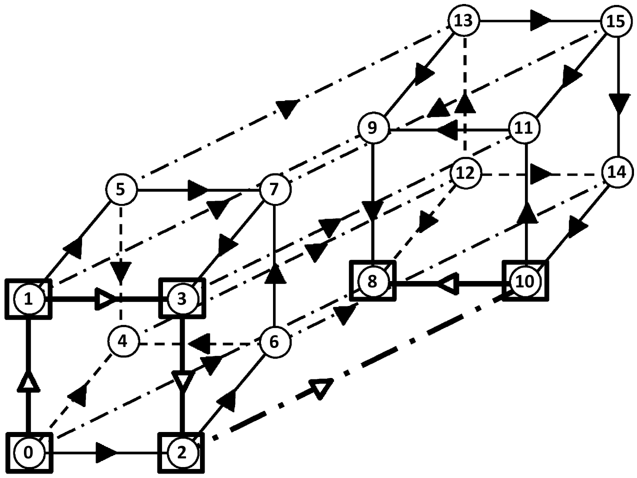
The paper is structured as follows. In the next section we define various history based pivot rules that have appeared in the literature: Zadeh's original rule, least-used direction rule, least-recently considered rule, least-recently basic rule, least-recently entered rule, and least iterations in the basis rule. We also give an example that shows they are all different. In Section 3 we develop an algorithm that generates all Hamiltonian paths, if any, followed by these history based pivot rules. We also provide computational results that show that most of these rules do not in fact produce Hamiltonian paths for dimensions up to 7, except in very low dimensions. In Section 4 we prove this fact holds for all higher dimensions for four of the history based rules we have presented.

2. History based pivot rules

In this section we review a number of history based pivot rules that have appeared in the literature starting with Zadeh's original rule. We also present an example to show that the rules all behave differently. The difference of these rules can be seen from the difference of the history array h. This array is indexed by the $2d$ directions (or sometimes, all d unsigned directions), and represents current historical information required for the given rule.

Zadeh noticed that the Klee–Minty construction (see Fig. 1) greatly favours some directions over others, and designed a new pivot rule to defeat this.

> *Zadeh's rule* (a.k.a. the *least entered rule*) [20]: For the entering variable, select the improving variable that has entered the basis least often thus far. (Fig. 4) The history array h is defined on all $2d$ directions, and $h(t)$ is the number of times the direction t is used.



Vertex (binary)	direction								Outgoing direction (bold for chosen)
	$+1$	-1	$+2$	-2	$+3$	-3	$+4$	-4	
0(0000)	0	0	0	0	0	0	0	0	$\mathbf{+1}, +2, +3, +4$
1(0001)	1	0	0	0	0	0	0	0	$\mathbf{+2}, +3, +4$
3(0011)	1	0	1	0	0	0	0	0	$\mathbf{-1}, +4$
2(0010)	1	1	1	0	0	0	0	0	$+3, \mathbf{+4}$
10(1010)	1	1	1	0	0	0	1	0	$+1, \mathbf{-2}$
8(1000)	1	1	1	1	0	0	1	0	

Fig. 4. Zadeh's rule.

In Zadeh's rule, as in others that we will study, there may be ties in selecting the entering variable. We will assume that ties may be broken arbitrarily in this paper. Note that Zadeh's rule chooses between all $2d$ variables (d decision variables and d slack variables) whereas the next history-based rule chooses between the d pairs of decision and slack variables, (x_i, x_{d+i}), each of which defines a *direction*. Directions are not a very useful concept in arbitrary polytopes, as no two edges may be parallel, but they are a natural feature of hypercubes and are inherited by zonotopes, which are projections of hypercubes. They directly inherit the d directions of the hypercube, some of which may no longer appear.

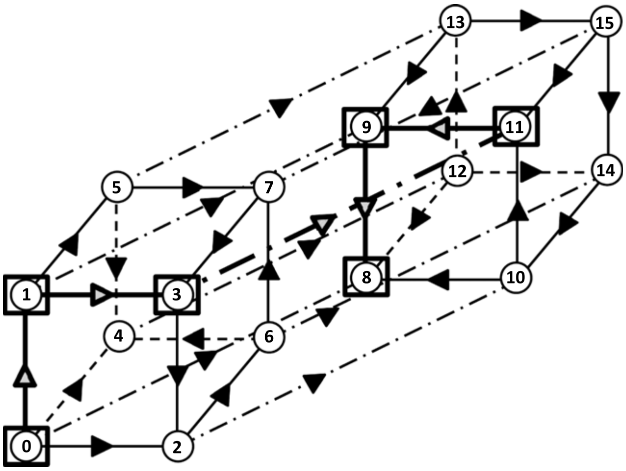
> *Least-used direction rule* (LUD) [3]: For the entering variable, select the improving variable whose unsigned direction has been used least often thus far. (Fig. 5) The history array h is defined on all d unsigned directions, and $h(t)$ is the number of times the direction t is used.

We now give some other history-based rules that have appeared in the literature. We show the paths generated by these rules on the previous example in the Appendix.

- *Least-recently considered rule* [4]: Fix an ordering of the variables $v_1, v_2, \ldots, v_{2d}$ and let the previous entering variable be v_i. For the entering variable, select the improving variable that first appears in the sequence $v_{i+1}, v_{i+2}, \ldots, v_{2d}, v_1, \ldots, v_{i-1}$ (or $v_1, \ldots, v_{2d}$ if this is the first pivot). The history array h is defined on all $2d$ directions and is initialized by setting $h(t)$ to be the rank of t in the given fixed ordering. If direction s is chosen the array is updated as $h(t) \leftarrow (h(t) - h(s) - 1 \mod 2d) + 1$. The Appendix shows the example of the case when initial sequence is $\{+2, -4, +1, -3, -2, +3, -1, +4\}$ (Fig. 11).
- *Least-recently basic rule* [Johnson in [4]]: For the entering variable, select the improving variable that left the basis least-recently. The history array h is defined on all $2d$ directions: $h(t)$ is the step number the $|t|$-th bit of the vertex was last 1 if t is positive or was last 0 if t is negative (Fig. 12).
- *Least-recently entered rule* (a.k.a. *least-recently used*) [6]: For the entering variable, select the improving variable that entered the basis least-recently thus far. The history array h is defined on all $2d$ directions: $h(t)$ is the step number when the $|t|$-th bit of the vertex last changes from 0 to 1 if t is positive or from 1 to 0 if t is negative (Fig. 13).
- *Least iterations in the basis rule* [3]: For the entering variable, select the improving variable that has been in the basis for the least number of iterations (Fig. 14). The history array h is defined on all $2d$ directions, and is the number of times the $|t|$-th bit of the vertex is 1 if t is positive or 0 if t is negative.

Note that all the examples illustrate distinct paths on the same AUSO cube.

In the following section we will describe an algorithm to determine if there are any AUSOs that admit Hamiltonian paths for the history based methods described in this section.



Vertex (binary)	direction 1	2	3	4	Outgoing directions (bold for chosen)
0(0000)	0	0	0	0	**1**, 2, 3, 4
1(0001)	1	0	0	0	**2**, 3, 4
3(0011)	1	1	0	0	1, **4**
11(1011)	1	1	0	1	**2**
9(1001)	1	2	0	1	**1**
8(1000)	2	2	0	1	

Fig. 5. Least used direction rule.

3. Searching for Hamiltonian paths on AUSOs that follow history based pivot rules

We developed an algorithm for determining if the various history based rules can be made to follow a Hamiltonian path on an AUSO. As noted in the introduction, it is known that the number of AUSOs is a doubly exponential, so a direct search quickly becomes infeasible. We use the fact that we are looking for AUSOs with Hamiltonian paths, which greatly reduces the search space.

3.1. Preliminaries

Our basic approach is to generate Hamiltonian paths starting with an unoriented hypercube, rather than first orienting the cube and checking if it is Hamiltonian. Suppose that a cube has a Hamiltonian path labelled with vertices $v_1, \ldots, v_N$. Then acyclicity implies immediately that each edge of the hypercube $v_i v_j$ with $i < j$ must be directed from v_i to v_j. Therefore, given a Hamiltonian path on the cube, we can easily construct the unique acyclic orientations for all edges of the cube. It still remains to test whether this orientation is an AUSO. Fortunately there is an efficient way to do this based on Williamson Hoke's theorem [8]:

Theorem 3.1. *If an orientation on a d-dimensional cube is acyclic, the following conditions are equivalent.*

- *The orientation is a unique sink orientation.*
- *For $k = 0, \ldots, d$ there are exactly $\binom{d}{k}$ vertices with indegree k $\left(\text{and hence } \binom{d}{k} \text{ vertices with outdegree } k\right)$.*

This makes it very easy to check if a given Hamiltonian path appears in an AUSO cube: we need only test the degree sequence. Furthermore, we can even use this test as the Hamiltonian path is being constructed. Note that when a vertex is added to the path its indegree and out-degree are known. Also partial degree information is known for unexplored vertices. Therefore if Williamson Hoke's condition is violated, we need not complete the construction of the given path. This leads to an efficient pruning technique. We also have the following interesting corollary.

Corollary 3.1. *In a Hamiltonian path on an AUSO d-cube starting from vertex* 0*, the indegree of a vertex is* 1 *if and only it is reached by a positive direction (or, unsigned direction) that is being used for the first time.*

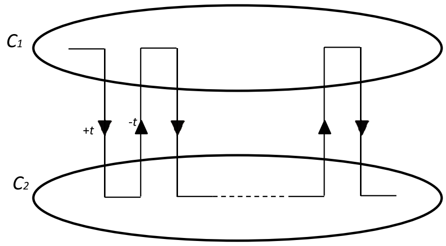


Fig. 6. The two $(d-1)$-dimensional cubes C_1 and C_2 separated by the direction t.

Proof. Suppose (v, v') is an edge on the Hamiltonian path that uses the direction $+t$ for the first time. Then all previous vertices in the path must have zero on the t-th bit. However all neighbours of v' on the hypercube except v have one on this bit, so they cannot have been visited yet. Therefore the indegree of v' is one. Since there are d directions, this yields d vertices on the path with indegree one. By Williamson Hoke's theorem this is the entire set of such vertices. □

As mentioned in the introduction, Zadeh's rule encourages each variable to be used as a pivot variable roughly the same number of times. We make this precise in the following result.

Theorem 3.2. *Assume that there is a Hamiltonian path P that follows Zadeh's rule on an AUSO of an n-cube. The least-used signed direction is used at least $\frac{2^{n-2}}{n} - \frac{3}{2}$ times.*

Proof. We may assume that P starts at vertex zero. Let $-t$ to be the least-used direction, and k be the number of times signed direction $-t$ is used. Partition the d-dimensional AUSO into two $(d-1)$-dimensional hypercubes C_1 and C_2 where the direction t separates the two (see Fig. 6) and P starts in C_1.

Let m_i, $i = 1, \dots, n$ be the number of times that signed direction $+i$ is used in P and m_{n+i}, $i = 1, \dots, n$ be the number of times that signed direction $-i$ is used in P. Since all 2^n vertices are visited, we have

$$\sum_{i=1}^{2n} m_i = 2^n - 1.$$

We know that the minimum value $m_{n+t} = k$ and $m_t = k + 1$. We can estimate the sum in another way by computing m_i as P is followed. Suppose we are at vertex v in C_1 and follow a signed direction $+i$ with $m_i \geq k + 2$. The signed direction $+t$ would have been a preferred choice since $m_t \leq k+1$. If signed direction $+t$ was not taken, then it must be that its neighbour in C_2 was already visited. We call this a blocked pair. A similar analysis holds if v is in C_2 and a signed direction $-i$ is chosen with $m_{n+i} \geq k + 1$. There can be at most 2^{n-1} blocked pairs. So in computing the sum of the m_i along P we have at most a contribution of $n(k + 2) + n(k + 1) - 1$ for the unblocked pivots and a contribution of at most 2^{n-1} for the blocked pivots. Therefore

$$\sum_{i=1}^{2n} m_i \leq n(2k + 3) - 1 + 2^{n-1}.$$

Combining the two expressions for the sum, the theorem follows. □

Unfortunately Theorem 3.2 only holds when the path is Hamiltonian. It is possible for a non-Hamiltonian exponential length path to use a signed direction as few as zero times! An example is shown in Fig. 7. Here we assume that C_1 and C_2 are copies of an AUSO cube with a long path. The resulting cube C is easily seen to be an AUSO. Note that since the path is non-Hamiltonian in C, vertices unvisited by the path in C_1 may be directed into C_2.

3.2. The algorithm and its validity

In this subsection we describe an algorithm that can generate, up to equivalence, all Hamiltonian paths on AUSOs using any of the history based pivot rules described in Section 2. In this paper, when we say two paths are equivalent, it means they are equivalent up to permutation. In other words, when two paths P and Q are equivalent, there is a permutation of coordinates $f : \{1, 2, \dots, d\} \to \{1, 2, \dots, d\}$ such that $f(P) = Q$. Algorithm 1 gives the pseudocode of our algorithm. We assume that the Hamiltonian path starts from the vertex labelled 0. We denote the indegree of the vertex x by $\text{indeg}(x)$. For $t = \pm 1, \dots, \pm d$ the function $\text{move}(x, t)$ returns the neighbour of x using the signed direction t, that is, the vertex $x + \text{sign}(t)2^{|t|-1}$. Note that we focus on this function only when t is a feasible move. The array h denotes the history information of the path and depends on the pivoting rule. For example, in the case of Zadeh's rule, $h(t)$ is the number of times the signed direction t is taken. We claim that the algorithm outputs, up to equivalence, all required Hamiltonian paths and that there are no duplications. First of all we show that each of the required Hamiltonian paths are equivalent to one of the paths output by the program. Below, by 'history based pivot rule' we refer to any of the rules described in Section 2.

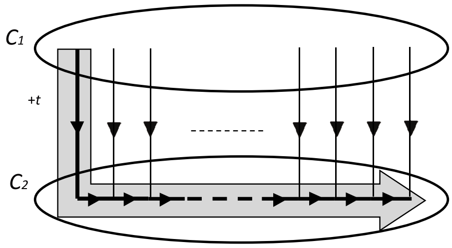


Fig. 7. An example of a non-Hamiltonian exponential path where one signed direction is never used.

Algorithm 1 Enumerate HP on AUSO-cube with history based pivot rule

1: $path \leftarrow \{0\}$.
2: **if** current path is Hamiltonian path **then**
3:   **if** $path$ is USO **then**
4:     output the result
5:   **end if**
6: **else**
7:   $m \leftarrow \min_{t\in\{\pm 1,\ldots,\pm d\},t \text{ is feasible}}\{h(t) \mid move(path.end, t) \text{ is not visited }\}$
8:   **for all** t such that $h(t) = m$ **do**
9:     **if** $h(t) = 0$ and $\exists t' < t$ $s.t.$ $h(t') = 0$ **then**
10:       continue
11:     **else**
12:       $v \leftarrow move(path.end, t)$
13:       **if** $h(t) \neq 0$ and $indeg(v) = 0$ **then**
14:         continue
15:       **end if**
16:       $path \leftarrow path + v$
17:       renew h
18:       continue searching (from line 2)
19:       recover h
20:       delete $path.end$
21:     **end if**
22:   **end for**
23: **end if**

Lemma 3.1. *For every Hamiltonian path P on a d-cube which can be followed by a history based pivot rule, there is a labelling of the cube such that P begins with vertex* 0 *and the order of positive directions first used in P is* $\{1, 2, \ldots, d-1, d\}$.

Proof of Lemma 3.1. We show how to embed P on a d-cube so that it has the required properties. We label the first vertex in P as 0 and the initial edge of P as direction $+1$. Continuing, for $i = 2, \ldots, d$ we consider the first edge of P that leaves a face of the cube of dimension $i - 1$. We define the direction used by this edge as $+i$. This induces a labelling of the cube with the desired properties. □

We remark that all paths produced by Algorithm 1 satisfy the conditions of Lemma 3.1 due to lines 8–10. Next we will prove that Algorithm 1 does not produce duplicate paths.

Lemma 3.2. *Let P and Q be two Hamiltonian paths produced by Algorithm* 1*. If there is a bijection (permutation of the coordinates) $f : \{1, 2, \ldots, d\} \to \{1, 2, \ldots, d\}$ such that $f(P) = Q$ then it is the identity mapping, i.e. $P = Q$.*

Proof of Lemma 3.2. As remarked, both P and Q satisfy the conditions of Lemma 3.1. Since $f(P) = Q$, both paths must use the k-th positive direction for the first time at the same time. By the lemma this must be direction $+k$, hence f is the identity mapping. □

As a consequence of these two lemmas we have the following result.

Theorem 3.3. *Algorithm* 1 *provides a complete duplicate free list of Hamiltonian paths on AUSO-cubes that follow a given history based pivot rule.*

Table 1
The number of Hamiltonian paths produced by history based pivot rules.

Dimension	2	3	4	5	6	7
Least entered rule	1	2	17	1072	3,262,342	$>10^{10}$
Least-used direction	1	1	1	2	0	0
Least recently entered	1	1	1	0	0	0
Least-recently considered rule	1	3	13	0	0	0
Least-recently basic rule	1	0	0	0	0	0
Least iterations in basis rule	1	0	0	0	0	0

Table 2
The number of Hamiltonian paths produced by history based pivot rules which satisfy Holt-Klee condition.

Dimension	2	3	4	5	6	7
Least entered rule	1	2	12	79	360	?
Least-used direction	1	1	1	0	0	0
Least recently entered	1	1	1	0	0	0
Least-recently considered rule	1	3	12	0	0	0
Least-recently basic rule	1	0	0	0	0	0
Least iterations in basis rule	1	0	0	0	0	0

3.3. Computational results

We implemented the algorithm and ran it on an Opteron computer with 2.2 GHz CPU, $4 \times 4 = 16$ processors and 132 GB of memory. We were able to do a complete enumeration up to dimension 6 and the results are shown in Table 1. The other rules refer to the least-recently considered, least-recently basic and least iterations in the basis rules. We see that the number of Hamiltonian path increases exponentially with Zadeh's least entered rule, whereas it becomes zero with the other pivot rules. On the basis of these results we conjecture that, except for the least entered rule, such Hamiltonian paths do not exist in any dimension greater than 6. We present proofs of these conjectures in the next section for all rules except for the least-used direction rule.

We also conducted an experiment to check whether these paths satisfy the Holt-Klee condition [9], a necessary condition for realizability of an LP-cube which states that every d-dimensional faces have at least d disjoint paths from a unique source to a unique sink (see Table 2). For dimension 7 with least entered rule, we have not found any such paths, but the computation was not completed due to the long running time.

4. Non-existence of Hamiltonian paths

In this section we prove that, except for the least entered rule and least-used direction rule, there are no Hamiltonian paths for the history based pivot rules considered except for those shown in Table 1.

Theorem 4.1. *The least iterations in basis rule and the least-recently basic rule do not have any Hamiltonian paths on a d-cube for $d \geq 3$.*

Proof. Suppose there is such a Hamiltonian path P for some d-cube. From Lemma 3.1 we can verify that the first d edges of P must take the directions $1, 2, \ldots, d$. Therefore P begins with $d+1$ vertices $0, 1, 3, \ldots, 2^d - 1$. The origin has indegree 0 and the other d vertices have indegree 1 in the AUSO induced by P. At this point, for each of the pivot rules, the direction -1 has the minimal value of h among the all the outgoing directions, so the first $d+1$ steps have to be $0, 1, \ldots, 2^d - 1, 2^d - 2$. The vertex $2^d - 2$ also has indegree 1, since it is not adjacent to any of the other vertices already on P. There are $d+1$ vertices which have indegree 1 in total: $1, 3, 7, \ldots, 2^d - 1$, and $2^d - 2$. This violates Williamson-Hoke's condition given in Theorem 3.1 which allows only d vertices to have indegree 1. □

We remark that this proof also can be used to show that the least-recently considered rule cannot have a Hamiltonian path for any $d \geq 3$ if the ordering begins with $+1, +2, \ldots, +d, -1$.

Theorem 4.2. *The least-recently entered rule does not have any Hamiltonian paths on a d-cube for $d \geq 5$.*

Proof. Suppose P is a Hamiltonian path produced by Algorithm 1 for the least-recently entered rule when $d \geq 5$. We will show that P must begin with the sequence of vertices $Q = \{Q_1, Q_2, Q_3, Q_4\}$ where $Q_1 = \{0, 1, 3, \ldots, 2^d - 1\}$, $Q_2 = \{2^d - 1 - 2^{d-2}, 2^d - 1 - 2^{d-2} - 2^{d-3}, \ldots, 2^{d-1}\}$, $Q_3 = \{2^{d-1} + 2, 2, 6, 14, \ldots, 2^d - 2\}$ and $Q_4 = \{2^d - 2 - 2^{d-2}, 2^d - 2 - 2^{d-2} - 2^{d-3}, \ldots, 2^{d-1} + 2 + 4 + 8, 2^{d-1} + 2 + 4\}$.

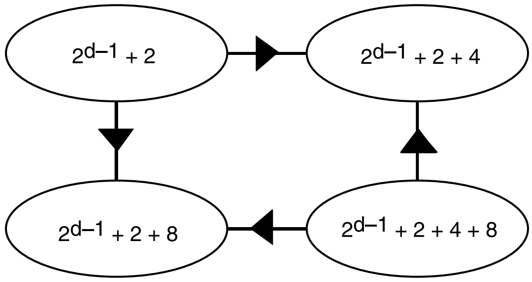


Fig. 8. 2-dimensional face with two sources and two sinks.

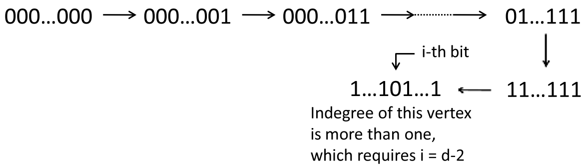


Fig. 9. Binary representation for the basic step of Q_2.

Q includes the vertices $\{2^{d-1}+2, 2^{d-1}+2+4+8, 2^{d-1}+2+4\}$ as a subsequence and does not contain the vertex $2^{d-1}+2+8$. These four vertices lie on a 2-face which has two sources, $2^{d-1}+2$ and $2^{d-1}+2+4+8$, a contradiction (see Fig. 8). It remains to show that P begins as specified.

- $Q_1 = 0, 1, 3, \ldots, 2^d-1$. This follows from Lemma 3.1.
- $Q_2 = 2^d-1, 2^d-2^{d-2}-1, 2^d-2^{d-2}-2^{d-3}-1, \ldots, 2^{d-1}$.
  We prove this by mathematical induction. For the basic step, we will show only $2^d-2^{d-2}-1$ can come right after 2^d-1. When we visited the vertex 2^d-1, all of the bits are 1. It means the next vertex can be represented as $2^d-2^k-1 = \sum_{i=0}^{d-1} 2^i - 2^k$ $(d-1 > k \geq 0)$. By Corollary 3.1, vertex $\sum_{i=0}^{d-1} 2^i - 2^k$ should have two visited neighbours, one of which is obviously the vertex 2^d-1. In other words, there exists $j \neq k$ such that $\sum_{i=0}^{d-1} 2^i - 2^k - 2^j \in \{0, 1, 3, \ldots, 2^d-1\} = \left\{v \middle| \exists l \text{ s.t. } v = \sum_{i=0}^{l} 2^i\right\} \cup \{0\}$. Since $d \geq 3$ forces $\sum_{i=0}^{d-1} 2^i - 2^k - 2^j$ not to be equal to 0, $\sum_{i=0}^{d-1} 2^i - 2^k - 2^j$ should be represented as $\sum_{i=0}^{l} 2^i = \sum_{i=0}^{d-1} 2^i - \sum_{i=l+1}^{d-1} 2^i$ for certain l. Therefore, the (k, j) equal $(d-1, d-2)$ or $(d-2, d-1)$, and $d-1 > k$ requires $k = d-2$. (See Fig. 9 for the binary representation.)
  We can prove the inductive step similarly. If the path is continued by $2^d-1, 2^d-1-2^{d-2}, \ldots, 2^d-1-\left\{\sum_{i=d-2-k}^{d-2} 2^i\right\}$, the next vertex should be equal to $\sum_{i=0}^{d-1} 2^i - \sum_{i=d-2-k}^{d-2} 2^i + 2^j$ $(d-2-k \leq j \leq d-2)$ or $\sum_{i=0}^{d-1} 2^i - \sum_{i=d-2-k}^{d-2} 2^i - 2^j$ $(j = d-1$ or $j < d-2-k)$. By Corollary 3.1, two neighbours of it are in $\{0, 1, 3, \ldots, 2^d-1, 2^d-1-2^{d-2}, \ldots, 2^d-1-\sum_{i=d-2-k}^{d-2} 2^i\}$. Using binary numbers, $2^d - \left\{\sum_{k=0}^{i} 2^{d-(2+k)}\right\} - 1$ can be denoted $100\ldots0011\ldots11$, where we have $k+1$ 0 s. (See Fig. 10 for the binary representation.)
- $Q_3 = \{2^{d-1}+2, 2, 6, 14, \ldots, 2^d-2\}$
  At the vertex 2^{d-1}, the history array becomes

$$h(x) = \begin{cases} d+x & (\text{if } x > 0) \\ 1 & (\text{if } x = -d) \\ 2d+1-x & (\text{if } d-1 \leq x \leq 0). \end{cases}$$

  (See Table 3.) Although its minimum value is 1, when $x = -d$, and the second smallest value is 2, when $x = +1$, we cannot use either the direction $-d$ or $+1$, since they lead to visited vertices. That leads us to use the direction $+2$, whose value is third smallest. The vertex $2^{d-1}+2$ enables us to use the direction $-d$ at last. Afterward, to avoid visiting an already visited vertex, we have to follow the sequence $\{2^{d-1}+2, 2, 6, 14, \ldots, 2^d-2\}$
- $Q_4 = \{2^d-2-2^{d-2}, 2^d-2-2^{d-2}-2^{d-3}, \ldots, 2^{d-1}+2+4+8, 2^{d-1}+2+4\}$
  This follows the same reasoning as Q_3, that is, using the smallest direction which reaches unvisited vertex fixes Q_4. Note that direction $+1$ cannot be used because the destination has already been visited in Q_2. □

For the least entered rule, we can prove the following feature concerning the beginning of any Hamiltonian path.

Theorem 4.3. *For every d dimensional Hamiltonian path using the least entered rule, different signed directions are used for the first $2d-1$ steps.*

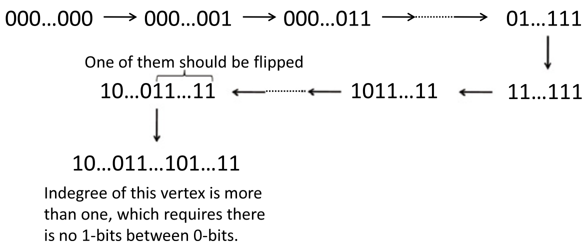


Fig. 10. Binary representation for the inductive step of Q_2.

Table 3
The history information of least-recently entered rule.

Vertex	Direction											Comment
(Binary)	$+1$	-1	$+2$	-2	$+3$	-3	$\cdots$	$+d-1$	$-(d-1)$	$+d$	$-d$	
$0(000\ldots000)$	0	1	0	1	0	1	$\cdots$	0	1	0	1	Initial state
$1(000\ldots001)$	2	1	0	1	0	1	$\cdots$	0	1	0	1	
$3(000\ldots011)$	2	1	3	1	0	1	$\cdots$	0	1	0	1	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
$2^{d-1}-1(011\ldots111)$	2	1	3	1	4	1	$\cdots$	d	1	0	1	
$2^d-1(111\ldots111)$	2	1	3	1	4	1	$\cdots$	d	1	$d+1$	1	End of Q_1
$2^d-1-2^{d-2}(101\ldots111)$	2	1	3	1	4	1	$\cdots$	d	$d+2$	$d+1$	1	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
$2^{d-1}+1(100\ldots001)$	2	1	3	$2d-1$	4	$2d-2$	$\cdots$	d	$d+2$	$d+1$	1	
$2^{d-1}(100\ldots000)$	2	$2d$	3	$2d-1$	4	$2d-2$	$\cdots$	d	$d+2$	$d+1$	1	End of Q_2
$2^{d-1}+2(100\ldots010)$	2	$2d$	$2d+1$	$2d-1$	4	$2d-2$	$\cdots$	d	$d+2$	$d+1$	1	
$2(000\ldots010)$	2	$2d$	$2d+1$	$2d-1$	4	$2d-2$	$\cdots$	d	$d+2$	$d+1$	$2d+2$	
$6(000\ldots110)$	2	$2d$	$2d+1$	$2d-1$	$2d+3$	$2d-2$	$\cdots$	d	$d+2$	$d+1$	$2d+2$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
$2^{d-1}-2(011\ldots110)$	2	$2d$	$2d+1$	$2d-1$	$2d+3$	$2d-2$	$\cdots$	$3d-1$	$d+2$	$d+1$	$2d+2$	
$2^d-2(111\ldots110)$	2	$2d$	$2d+1$	$2d-1$	$2d+3$	$2d-2$	$\cdots$	$3d-1$	$d+2$	$3d$	$2d+2$	End of Q_3

Proof. Let v be a vertex visited during the first $2d-1$ steps. It is enough to show there is a direction t and an unvisited vertex move(v, t) for which $h(t) = 0$.

If $t > 0$ and $h(t) = 0$ then vertex move(v, t) cannot have been visited yet, so t is a candidate direction. Otherwise, if for each $t > 0$ we have $h(t) = 1$ and there exists at least two negative directions $-t_1, -t_2$ such that $h(-t_1) = h(-t_2) = 0$. Assume that the direction $+t_1$ was used earlier than $+t_2$. So for all vertices visited so far, it is impossible to have both the t_2-th bit at 1 and the t_1-th bit at 0. This means that the vertex move$(v, -t_1)$ has not been visited and $-t_1$ is a candidate direction. □

5. Discussion

From our computational experiments, Zadeh's least entered rule seems very likely to have Hamiltonian paths on AUSO cubes. Using our program, we could verify such paths exist up to dimension 9, but did not yet find any for dimension 10. Furthermore, we could not find any general construction, so this is an open problem. Even if such Hamiltonian paths exist, it is not clear whether or not they could be obtained on AUSOs that are realizable as polytopes.

Although we showed that a number of history based pivot rules do not admit Hamiltonian paths in general, they may still admit exponential length paths. Since our program makes heavy use of the fact that we are searching for Hamiltonian paths, we were not able to use it to check this for low dimensions.

Our computer result for Zadeh's rule allow ties to be broken arbitrarily, as does the theoretical lower bound obtained in [7]. It would be interesting to see the effects of various deterministic tie breaking rules on these results.

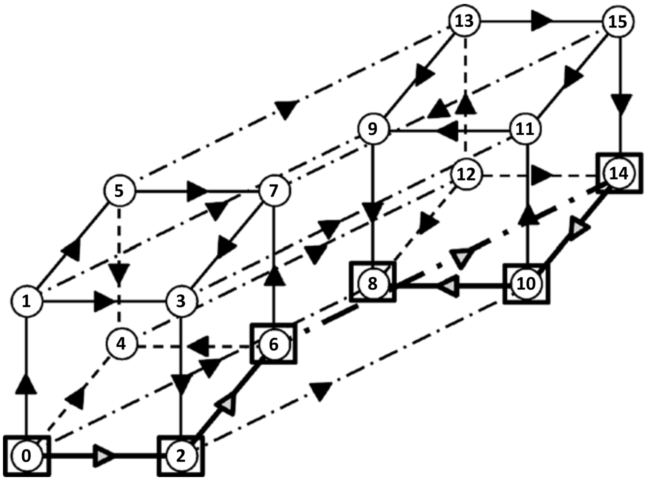
Acknowledgements

We are grateful to ERATO-SORST Quantum Computation and Information Project, Japan Science and Technology Agency. All of our computational experiments are conducted on the cluster computer in ERATO-SORST. Work on this project was also supported by an INTRIQ-ERATO/SORST collaboration grant funded by MDEIE (Québec), a discovery grant from NSERC (Canada), and KAKENHI (Japan).

We thank the anonymous reviewers for their useful comments which helped us to improve this paper.

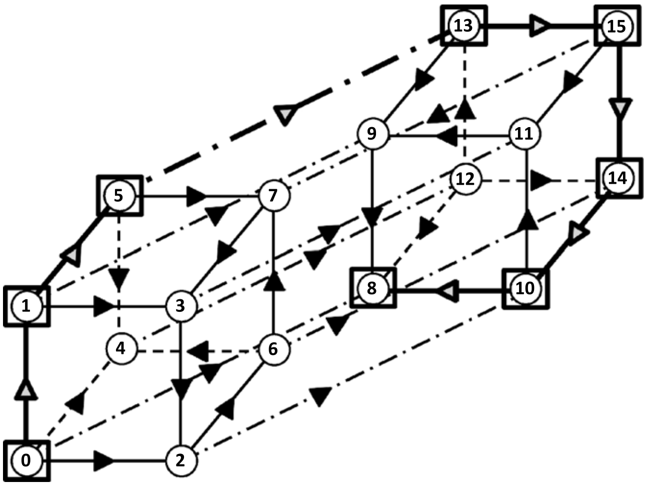
Appendix

Figs. 11–14.



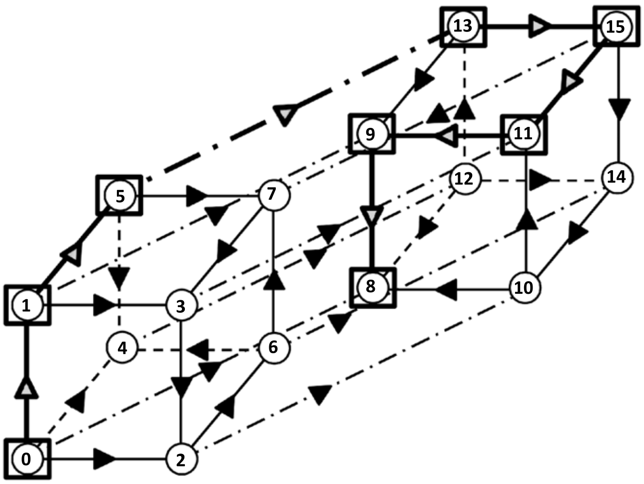
Vertex (binary)	direction								Outgoing directions (bold for chosen)
	+1	−1	+2	−2	+3	−3	+4	−4	
0(0000)	3	7	1	5	6	4	8	2	+1, **+2**, +3, +4
2(0010)	2	6	8	4	5	3	7	1	**+3**, +4
6(0110)	5	1	3	7	8	6	2	4	+1, −2, **+4**
14(1110)	3	7	1	5	6	4	8	2	**−3**
10(1010)	7	3	5	1	2	8	4	6	+1, **−2**
8(1000)	6	2	4	8	1	7	3	5	

Fig. 11. Least recently considered rule.



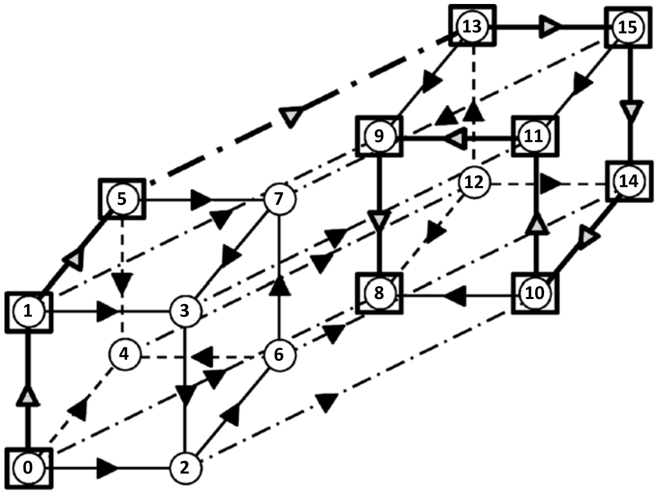
Step number	Vertex (binary)	direction								Outgoing directions (bold for chosen)
		+1	−1	+2	−2	+3	−3	+4	−4	
1	0(0000)	0	1	0	1	0	1	0	1	**+1**, +2, +3, +4
2	1(0001)	2	1	0	2	0	2	0	2	+2, **+3**, +4
3	5(0101)	3	1	0	3	3	2	0	3	−1, +2, **+4**
4	13(1101)	4	1	0	4	4	2	4	3	**+2**, −3
5	15(1111)	5	1	5	4	5	2	5	3	**−1**, −3, −4
6	14(1110)	5	6	6	4	6	2	6	3	**−3**
7	10(1010)	5	7	7	4	6	7	7	3	+1, **−2**
8	8(1000)	5	8	7	8	6	8	8	3	

Fig. 12. Least recently basic rule.



Step number	Vertex (binary)	direction								Outgoing directions (bold for chosen)
		+1	−1	+2	−2	+3	−3	+4	−4	
1	0(0000)	0	1	0	1	0	1	0	1	**+1**, +2, +3, +4
2	1(0001)	2	1	0	1	0	1	0	1	+2, **+3**, +4
3	5(0101)	2	1	0	1	3	1	0	1	−1, +2, **+4**
4	13(1101)	2	1	0	1	3	1	4	1	**+2**, −3
5	15(1111)	2	1	5	1	3	1	4	1	−1, **−3**, −4
6	11(1011)	2	1	5	1	3	6	4	1	**−2**
7	9(1001)	2	1	5	7	3	6	4	1	**−1**
8	8(1000)	2	8	5	7	3	6	4	1	

Fig. 13. Least recently entered rule.



Vertex (binary)	direction								Outgoing directions (bold for chosen)
	$+1$	-1	$+2$	-2	$+3$	-3	$+4$	-4	
0(0000)	0	1	0	1	0	1	0	1	$\mathbf{+1}, +2, +3, +4$
1(0001)	1	1	0	2	0	2	0	2	$+2, \mathbf{+3}, +4$
5(0101)	2	1	0	3	1	2	0	3	$-1, +2, \mathbf{+4}$
13(1101)	3	1	0	4	2	2	1	3	$\mathbf{+2}, -3$
15(1111)	4	1	1	4	3	2	2	3	$\mathbf{-1}, -3, -4$
14(1110)	4	2	2	4	4	2	3	3	$\mathbf{-3}$
10(1010)	4	3	3	4	4	3	4	3	$\mathbf{+1}, -2$
11(1011)	5	3	4	4	4	4	5	3	$\mathbf{-2}$
9(1001)	5	3	4	5	4	5	6	3	$\mathbf{-1}$
8(1000)	5	4	4	6	4	6	7	3	

Fig. 14. Least iterations in basis rule.

References

[1] I. Adler, R. Saigal, Long monotone paths in abstract polytopes, Mathematics of Operations Research 1 (1) (1976) 89–95.
[2] D. Avis, V. Chvátal, Notes on Bland's pivoting rule, in: Polyhedral Combinatorics, 1978, pp. 24–34.
[3] D. Avis, S. Moriyama, Y. Matsumoto, History based pivot rules and unique sink orientations, in: Japan–Canada Workshop, July 2009.
[4] W.H. Cunningham, Theoretical properties of the network simplex method, Mathematics of Operations Research 4 (2) (1979) 196–208.
[5] G.B. Dantzig, M.N. Thapa, Linear Programming: Theory and Extensions, Springer Verlag, 2003.
[6] Y. Fathi, C. Tovey, Affirmative action algorithms, Mathematical Programming 34 (3) (1986) 292–301.
[7] O. Friedmann, A subexponential lower bound for Zadeh's pivot rule for solving linear programs and games, in: Proceedings of the 15th Integer Programming and Combinatorial Optimization, 2011, pp. 192–206.
[8] K.W. Hoke, Completely unimodal numberings of a simple polytope, Discrete Applied Mathematics 20 (1) (1988) 81.
[9] F. Holt, V. Klee, A proof of the strict monotone 4-step conjecture, in: Advances in Discrete and Computational Geometry: Proceedings of the 1996 AMS-IMS-SIAM Joint Summer Research Conference, Discrete and Computational Geometry—Ten Years Later, July 14–18, 1996, Mount Holyoke College, vol. 233, Amer Mathematical Society, 1998, p. 201.
[10] R.G. Jeroslow, The simplex algorithm with the pivot rule of maximizing criterion improvement, Discrete Mathematics 4 (4) (1973) 367–377.
[11] G. Kalai, A subexponential randomized simplex algorithm (extended abstract), in: Proceedings of the Twenty-Fourth Annual ACM Symposium on Theory of Computing, ACM, New York, NY, USA, 1992, pp. 475–482.
[12] L.G. Khachian, Polynomial algorithms in linear programming, Zhurnal Vychislitel'noĭ Matematiki i Matematicheskoĭ Fiziki 20 (1980) 51–68.
[13] V. Klee, G.J. Minty, How good is the simplex algorithm, in: Inequalities III, Academic Press, New York, 1972, pp. 159–175.
[14] J. Matoušek, The number of unique-sink orientations of the hypercube, Combinatorica 26 (2006) 91–99.
[15] J. Matoušek, M. Sharir, E. Welzl, A subexponential bound for linear programming, in: Proc. 8th Annual Symposium on Computational Geometry, ACM Press, 1992, pp. 1–8.
[16] S. Moriyama, Enumeration of shellable *h*-assignments using rsshell, 2006. http://www-imai.is.s.u-tokyo.ac.jp/~moriso/enumRsshell/.
[17] F. Santos, A counterexample to the Hirsch conjecture, 2010, Arxiv Preprint arXiv:1006.2814.
[18] A. Stickney, L. Watson, Digraph models of Bard-type algorithms for the linear complementarity problem, Mathematics of Operations Research 3 (4) (1978) 322–333.
[19] T. Szabó, E. Welzl, Unique sink orientations of cubes, in: Annual Symposium on Foundations of Computer Science, vol. 42, 2001, pp. 547–557.
[20] N. Zadeh, What is the worst case behavior of the simplex algorithm? in: Polyhedral Computation, in: CRM Proceedings and Lecture Notes, vol. 48, American Mathematical Society, 2009, pp. 131–143.