Nr. 22

Schwerpunktprogramm der Deutschen Forschungsgemeinschaft

ANWENDUNGSBEZOGENE OPTIMIERUNG UND STEUERUNG

Cycling Examples for the Shadow Vertex Algorithm

by

Karl Heinz Borgwardt, Gabriele Joas

Report No. 336 1991

Institut für Mathematik

UNIVERSITÄT AUGSBURG

Universitätsstraße 8

D-8900 Augsburg

Cycling Examples for the Shadow Vertex Algorithm

Karl Heinz Borgwardt, Gabriele Joas

Abstract

We are primarily interested in the average number of pivot steps, the shadow vertex algorithm takes to solve linear programs. Therefore we have to guarantee that the algorithm works determined and cycle-free.
First the problems are divided into different classes of degeneracy according to their difficulty. We will then show that determination is lost with degenerate problems, which requires an adaption of the shadow vertex algorithm. For an obvious modification of the algorithm we are able to construct cycling examples for almost all classes of degeneracy, whereas the shadow vertex algorithm works cycle-free for the rest of them.

Introduction

We consider problems of the form

$$\begin{aligned}
&\max\ v^T x \\
&\text{s.t.} \quad a_1^T x \leq 1, \ldots, a_m^T x \leq 1 \\
&\text{where} \quad v, x, a_1, \ldots, a_m \in \mathbf{R}^n \quad \text{and} \quad m \geq n.
\end{aligned}$$

In addition $u \in \mathbf{R}^n$ —the so-called start vector— is given. What it is needed for, will be explained later.

For all successful probabilistic studies of the simplex algorithm the shadow vertex variant, which comes close to a modification of the Gass-Saaty algorithm [2], served as the basis. Borgwardt [1] showed that the average running time of this variant is polynomial, if the input vectors $a_1, \ldots, a_m, u$ and v are distributed independently, identically and symmetrically under rotations. The rotation symmetry of the input however implies, that degenerate problems (the exact definition of degeneracy is given later) only appear with probability 0.

It is now our aim to examine the running time of the shadow vertex algorithm even on degenerate problems. Therefore it must be guaranteed that the shadow vertex algorithm works determined and cycle-free on these problems. Otherwise, if the probability of cycling is positive, the average running time grows exceedingly. Then modifications as well in the algorithm as in the theoretical proof methods are necessary. The question whether the shadow vertex algorithm cycles or not is answered in [4]. The authors of [4] discuss a cycling example, which was introduced in [5]. This example however does not fit into our special form with constant right side 1. In addition, the feasible region only consists of the origin, and u is a negative multiple of v, what means that the problem is highly degenerate. The question whether cycling on our problem type with right side 1 (that implies a full dimensional feasible region) is possible, remains still open. It is answered positively in this paper. Besides we examine how "severe" degeneracy has to be to produce cycles. It will be shown that already harmless degenerate problems, compared to that cited in [4], can cycle.

In part 1 the shadow vertex algorithm is explained. Especially the geometric differences between the feasible regions of degenerate and non-degenerate problems are shown.

Afterwards a classification of degenerate problems according to their difficulties is given, where the related geometry plays an important role.

Part 3 deals with one degenerate vertex. The most important questions in this context are: Does the shadow vertex algorithm still keep its determination in a degenerate vertex? Are cycles possible? The first question has to be negotiated.

That is why in the last part we modify the shadow vertex algorithm in the following way. In each pivot step choose **any** of the possible entering resp. leaving basis-vectors. However, for this obvious modification, the second question has to be answered in the affirmative.

In the next step for almost all previously defined degeneracy classes cycling examples are constructed. For the rest it is proven that the shadow vertex algorithm works cycle-free.

1. The Shadow Vertex Algorithm

1.1 The Geometry

We consider problems of the form:

$$\begin{aligned} &\max \ v^T x \\ &\text{s.t.} \quad a_1^T x \le 1, \dots, a_m^T x \le 1 \\ &\text{where} \quad v, x, a_1, \dots, a_m \in \mathbf{R}^n \quad \text{and} \quad m \ge n. \end{aligned} \tag{1.1}$$

In addition a so-called start vector $u \in \mathbf{R}^n$ is given. For the present we are dealing with non-degenerate problems, afterwards we will show the differences to degenerate problems. We use the following characterization of non-degeneracy:

Each subset of $n+1$ elements out of $\{a_1, \dots, a_m\}$ is in general position (primal non-degeneracy), and each n-element subset of $\{a_1, \dots, a_m, u, v\}$ is linearly independent (polar non-degeneracy). (1.2)

We solve problems of the form (1.1) by the shadow vertex variant of the simplex algorithm introduced by [1]. For this purpose it is useful not to carry out the studies in the primal space, which contains the feasible region, but in the dual (or polar) space.
By $X := \{x \in \mathbf{R}^n \mid a_1^T x \le 1, \dots, a_m^T x \le 1\}$ we denote the feasible region, which is a polyhedron, and is called **primal polyhedron**. $Y := \{y \in \mathbf{R}^n \mid y^T x \le 1 \quad \forall x \in X\}$ is the corresponding **polar polyhedron**.
Y lies in the already mentioned dual space. It can easily be characterized in the following way.

Lemma 1: *$Y = CH(0, a_1, \dots, a_m)$, where CH denotes the convex hull.*

For a better understanding fig. 1 shows a primal and its related polar polyhedron.

The following lemma clarifies the connection between the primal and the dual space. Here $\Delta := \{\Delta^1, \dots, \Delta^n\} \subset \{1, \dots, m\}$ is an n-element set of indices.

Lemma 2: *Let x_Δ be a vertex of X and $\Sigma(\Delta)$ the convex hull of all a_{Δ^i}, $\Delta^i \in \{1, \dots, m\}$ with $a_{\Delta^i}^T x_\Delta = 1$. Then $\Sigma(\Delta)$ is a facet of Y.*

If the vertex x_Δ is non-degenerate, exactly n of the restrictions hold as strict equalities. As we examine degenerate vertices, it is possible that there are more than n restrictions, which hold as strict equalities. In both cases we say that $\Sigma(\Delta)$ is a **boundary polytope** of Y.

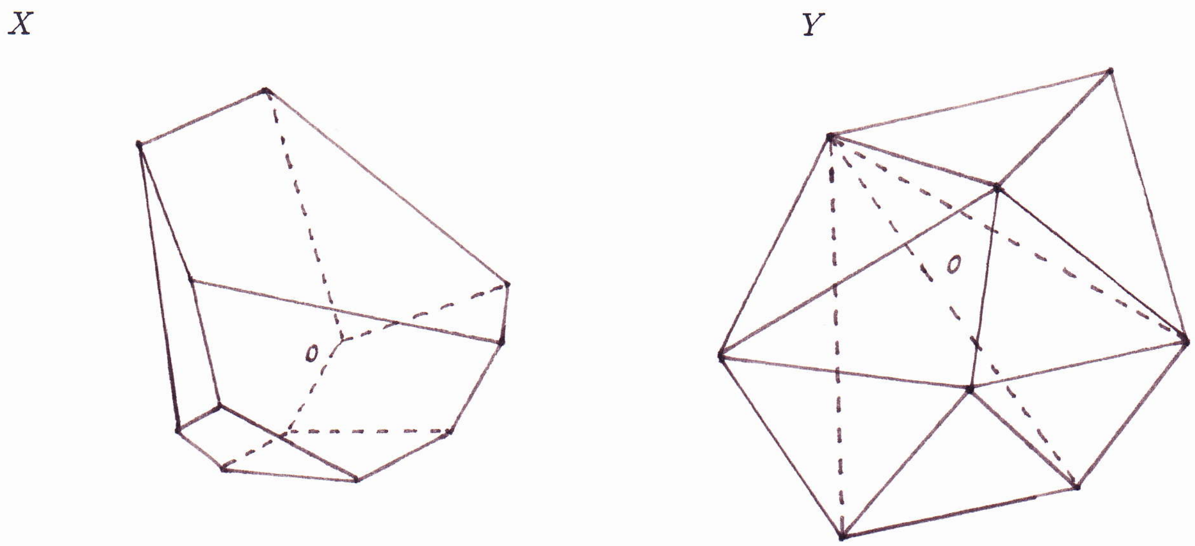


Fig. 1

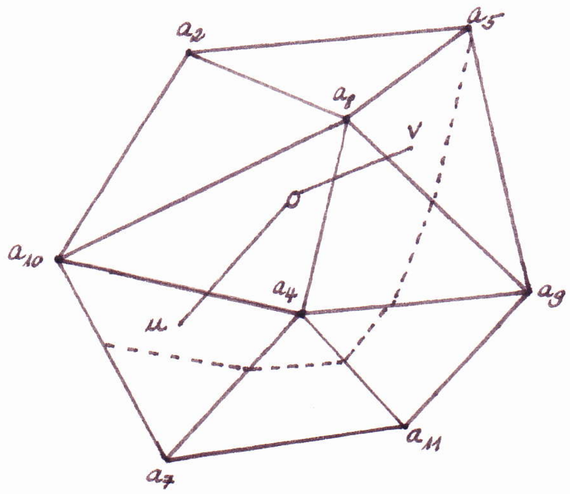


Fig. 2

Let us now consider the geometry standing behind the shadow vertex algorithm, as long as degeneracy is absent.

Fig. 2 shows the foreground of a polar polyhedron. Let u be the start vector mentioned earlier, and v the gradient of the objective. Now we explain what this start vector is needed for. Therefore we consider the primal space. Suppose that phase I of the simplex algorithm has already been done, and has provided us with a vertex x_0 of X. Then this vertex should optimize the given objective $u^T x$ on X. Now back to fig. 2. In the polar space we start at this u and move along the plane $span(u, v)$ to v. During this movement we intersect facets of the polyhedron Y, the so-called **boundary simplices**. In the primal interpretation this means moving on a simplex path from one vertex to the next. Each facet is spanned by exactly n independent vectors, which build a basis of $\mathbf{R}^n$. Passing over from one facet to the next, we have to perform a pivot step, what means that one vector leaves the basis, while another vector enters it. These vectors are called **leaving** resp. **entering** vector. Arriving at v we get the optimal boundary polytope, and therefore the optimal basis, and the algorithm stops. These results are summarized in the following lemma.

Lemma 3: *The boundary polytopes which intersect* $CC(u,v) = \{y \mid y = \lambda u + \rho v, \lambda \geq 0, \rho \geq 0\} \subset span(u,v)$ *can be arranged uniquely in a sequence* $\Sigma(\Delta_0), \ldots, \Sigma(\Delta_s)$, *such that* $\Delta_i \neq \Delta_j$ *for* $i \neq j$, Δ_i *and* Δ_{i+1} *differ only in one element, and* $arc(z_i, v) \geq arc(z_{i+1}, v)$ *for every pair* (z_i, z_{i+1}) *with* $z_i \in \Sigma(\Delta_i) \cap span(u,v)$, $z_{i+1} \in \Sigma(\Delta_{i+1}) \cap span(u,v)$.

Remark: If the problem has an optimal solution, $\Sigma(\Delta_s)$ is the boundary polytope belonging to the optimal vertex in X. If the problem is unbounded, $\Sigma(\Delta_s)$ corresponds to the vertex x_s, where it becomes obvious that the problem has no solution.

The proofs to all three lemmas as well as primal and dual descriptions of the algorithm can be found in [1].

Now let us deal with degenerate problems, which result from the violation of the general position condition. First we want to examine the situation when one degenerate boundary polytope is at hand like in figure 3.

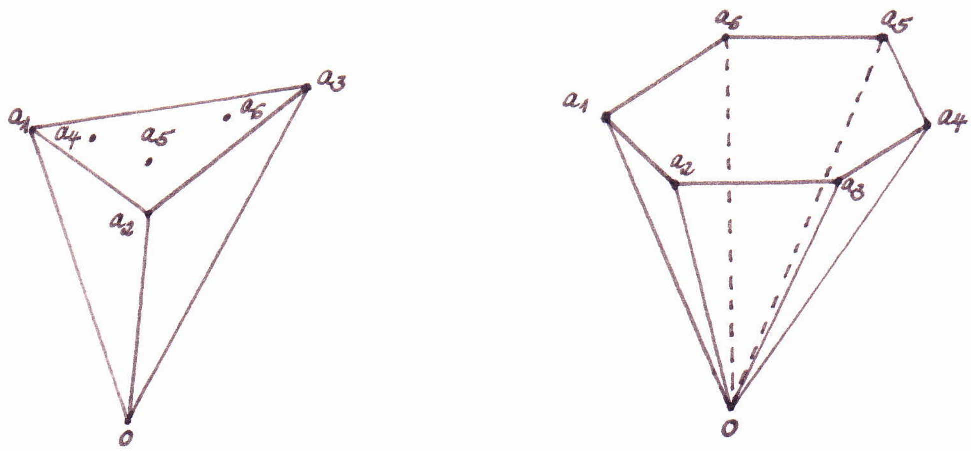


Fig. 3

Here both boundary polytopes are spanned by more than n (in our case $n = 3$) vectors, namely $a_1, \ldots, a_6$, which implies degeneracy. Each restriction induced by the 6 vectors holds with equality in a vertex of the primal polyhedron X.

In principle the shadow vertex algorithm works under degeneracy like in non-degenerate cases, which we already explained. As single facets can be spanned by more than n vectors, we eventually have to perform pivot steps during crossing one facet.
In figure 4 the number of pivot steps (comp. fig. 2) can grow from 4 to 7, because the boundary polytopes $CH(a_4, a_7, a_{10})$ and $CH(a_4, a_8, a_9)$ possibly are not run through in one single step. A possible sequence of basis for the simplex path is e.g.:

$$(a_7, a_{10}, a_{12}) \longrightarrow (a_4, a_{10}, a_{12}) \longrightarrow (a_4, a_7, a_{12}) \longrightarrow (a_4, a_7, a_{11}) \longrightarrow$$
$$(a_4, a_9, a_{11}) \longrightarrow (a_4, a_9, a_{13}) \longrightarrow (a_8, a_9, a_{13}) \longrightarrow (a_5, a_8, a_9)$$

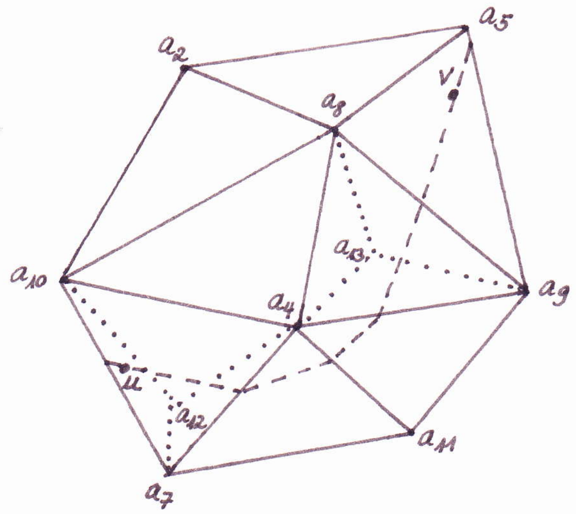


Fig. 4

Degeneracy can also have its origin in the fact, that there exist i, $2 \leq i \leq n$, vectors out of $\{a_1, \ldots, a_m, u, v\}$, which are linearly dependent. The harmless case is, if n of the vectors $\{a_1, \ldots, a_m\}$ are linearly dependent. Then these vectors do not form a basis, and hence no vertex in the primal space. So this n-element vector set cannot contribute additional pivot steps, and it does not need to be considered in our theoretical analysis. But if the vectors u and v are involved, then the situation is more complicated and the algorithm may be stalled, and "ambiguities" may occur. Different facets of Y can have more than one common point belonging to $CC(u, v)$, or one point of $CC(u, v)$ is contained in more than two facets of Y. (These cases are excluded under non-degeneracy!) Out of this one can already see, that degenerate problems can have different complexities. That is why we want to classify degeneracy formally. But before doing that, let us explain the numerics behind the shadow vertex algorithm.

1.2 Numerical Realization

In this part we want to introduce the tableau representation, which we will refer from now on.

In each vertex x_Δ of X at least n restrictions $a_i^T x = 1$ hold with equality, and there exist exactly n restriction vectors out of them, which build a basis of $\mathbf{R}^n$. Assume $a_{\Delta^1}, \dots, a_{\Delta^n}$ is such a basis with $\Delta = \{\Delta^1, \dots, \Delta^n\} \subset \{1, \dots, m\}$. We are now able to represent each of the vectors $a_1, \dots, a_m, u, v$ as a linear combination of these basis vectors. Then the entries of the simplex tableau reflect the coefficients of the linear combinations.

$$
\begin{array}{c|ccccccc|c|c}
 & a_1 & \cdots & a_n & a_{n+1} & \cdots & a_m & & v & u \\
\hline
a_{\Delta^1} & 1 & & 0 & & & & & \alpha_1 & \beta_1 \\
\vdots & & \ddots & & & \gamma_{kl} & & & \vdots & \vdots \\
a_{\Delta^n} & 0 & & 1 & & & & & \alpha_n & \beta_n \\
\hline
 & 0 & \cdots & 0 & \Psi_{n+1} & \cdots & \Psi_m & & Q_v & Q_u \\
\hline
 & \| & & \| & & & & & & \\
 & \Psi_1 & & \Psi_n & & & & & &
\end{array}
$$

W.l.o.g. $\Delta = \{1, \dots, n\}$. The values $\Psi_1, \dots, \Psi_m$ are the slacks $1 - a_i^T x_\Delta$, $i = 1, \dots, m$, Q_v and Q_u are the objective values of $v^T x$ resp. $u^T x$.
A vertex x_Δ of X is optimal, if $\alpha_k \geq 0$ $\forall k = 1, \dots, n$, because then v lies in the polar cone of the corresponding restrictions, which are active in x_Δ. If there is an $\alpha_k < 0$ $(k \in \{1, \dots, n\})$, then the actual basis is not yet optimal, and we have to perform a pivot step. That means a basis vector a_l $\ (l \in \Delta,\ l = \Delta^i)$ with $\alpha_i < 0$ is replaced by a_j with $j \notin \Delta$.

In a pivot step the following conditions must be true:

$$
\begin{gathered}
\alpha_i < 0, \quad \gamma_{ij} < 0 \quad \text{and} \\
\frac{\Psi_j}{\gamma_{ij}} = \max_{\substack{l \notin \Delta \\ \gamma_{il} < 0}} \frac{\Psi_l}{\gamma_{il}}
\end{gathered}
$$

$\{a_{\Delta^1}, \dots, a_{\Delta^{i-1}}, a_j, a_{\Delta^{i+1}}, \dots, a_{\Delta^n}\}$ is then the new basis and we can calculate a new tableau. If $\gamma_{ij} \geq 0$ for i with $\alpha_i < 0$ and $j = 1, \dots, m$, then $v^T x$ is unbounded over X.

Until now it is not clear which row to choose, if more than one α_k is less than zero $(k \in \{1, \dots, n\})$. The leaving vector in the shadow vertex variant has to fullfil the following condition:

$$
\frac{-\beta_i}{\alpha_i} = \min_{\substack{1 \leq k \leq n \\ \alpha_k < 0}} \frac{-\beta_k}{\alpha_k}
$$

2. Classification of Degeneracy

For all later considerations we assume:

$$\begin{array}{l}\text{Each } n\text{-element subset } \{v, \bar{a}_1, \ldots, \bar{a}_{n-1}\} \text{ and} \\ \text{each } n\text{-element subset } \{u, \bar{a}_1, \ldots, \bar{a}_{n-1}\} \text{ with} \\ \{\bar{a}_1, \ldots, \bar{a}_{n-1}\} \subset \{a_1, \ldots, a_m\} \text{ is linearly independent.}\end{array} \tag{2.1}$$

The number of spanning vectors of a boundary polytope as well as the degree of linear dependence of the input vectors are the central quantities for a classification.

Definition 1: *A boundary polytope $\Sigma(\Delta) = CH(a_{\Delta^1}, \ldots, a_{\Delta^p})$, $p \geq n$ of Y, which is intersected by $CC(u, v)$ belongs to the degeneracy class $D(p, \alpha, \beta)$ with:*

$\alpha := i$, *if every n-element subset out of $\{a_{\Delta^1}, \ldots, a_{\Delta^p}\}$ is linearly independent.*

$\alpha := d$, *if there exist n vectors out of $\{a_{\Delta^1}, \ldots, a_{\Delta^p}\}$, which are linearly dependent.*

$\beta := i$, *$i \in \mathbf{N}$, $1 \leq i \leq n-1$, if there is a $w \in CC(u, v) \cap \Sigma(\Delta)$ which has a representation $w = \lambda_1 \bar{a}_1 + \ldots + \lambda_i \bar{a}_i$ where $\lambda_1, \ldots, \lambda_i > 0$, $\sum_{k=1}^{i} \lambda_k = 1$ and $\{\bar{a}_1, \ldots, \bar{a}_i\} \subset \{a_{\Delta^1}, \ldots, a_{\Delta^p}\}$, but there is no $w \in CC(u, v) \cap \Sigma(\Delta)$, which can be represented by less than i vectors out of $\{a_{\Delta^1}, \ldots, a_{\Delta^p}\}$ in the way mentioned above.*

Definition 2: *A problem of the form (1.1) belongs to the degeneracy class*

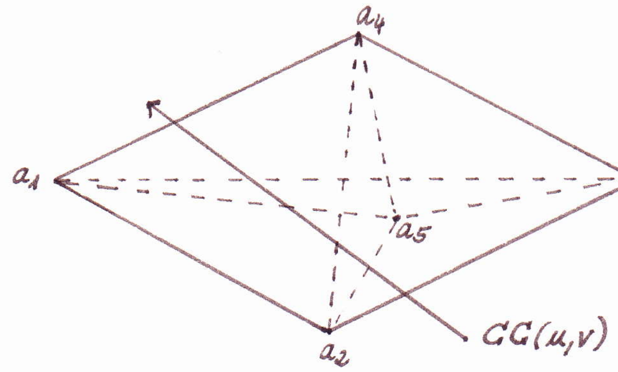
- *$D(p, i, \beta_1)$, if all of the boundary polytopes of Y belong to the degeneracy class $D(k, i, \beta)$, $n \leq k \leq p$, $\beta \geq \beta_1$, if (at least) one boundary polytope belongs to the class $D(k, i, \beta_1)$, and (at least) one boundary polytope is spanned by p vectors.*
- *$D(p, d, \beta_2)$, if all boundary polytopes of Y belong to the class $D(k, \cdot, \beta)$, $n \leq k \leq p$, $\beta \geq \beta_2$, and if (at least) one boundary polytope belongs to the class $D(k, d, \beta)$, $\beta \geq \beta_2$, and one to the class $D(k, \cdot, \beta_2)$, and if (at least) one boundary polytope is spanned by p vectors.*

What does this classification mean graphically? Fig. 5 answers this question. Boundary polytopes $\Sigma(\Delta) = CH(a_1, \ldots, a_5) \subset \mathbf{R}^3$ seen from above are shown in the following.

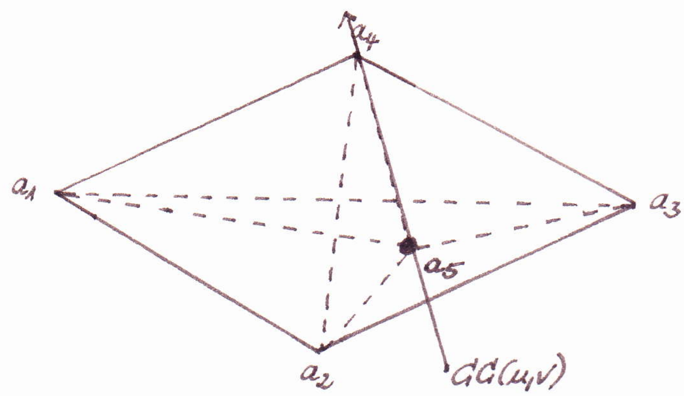
(ii), (iii), (v), and (vi) of the following figure are primally as well as polarly degenerate. The critical points are responsible for (ii), (iii), (v), and (vi) belonging to an "unpleasant" degeneracy class, for there exist (linear) dependencies between u, v, and the a-vectors. Loosely speaking: The smaller the value of β, the worse is the degree of degeneracy.

The next step is to examine the performance of the simplex algorithm within a single degenerate boundary polytope.

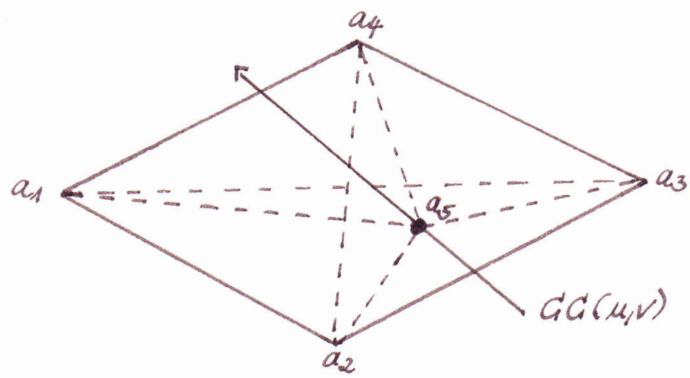
(i) $D(5, i, n-1) = D(5, i, 2)$



(ii) $D(5, i, n-2) = D(5, i, 1)$

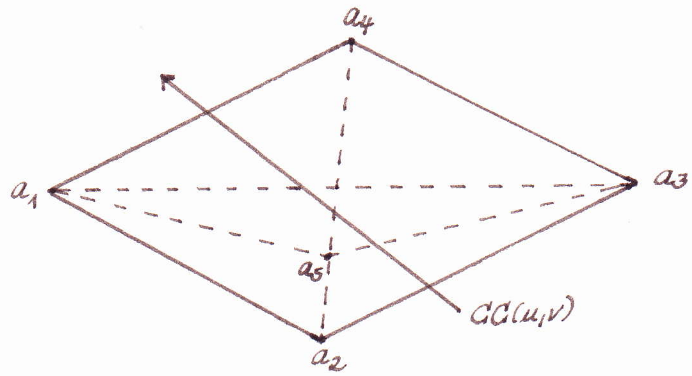


(iii) $D(5, i, n-2) = D(5, i, 1)$

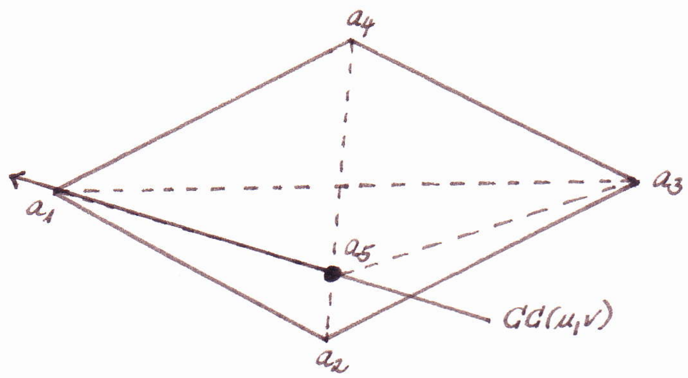


• critical points

(iv) $D(5, d, n-1) = D(5, d, 2)$



(v) $D(5, d, n-2) = D(5, d, 1)$



(vi) $D(5, d, n-2) = D(5, d, 1)$

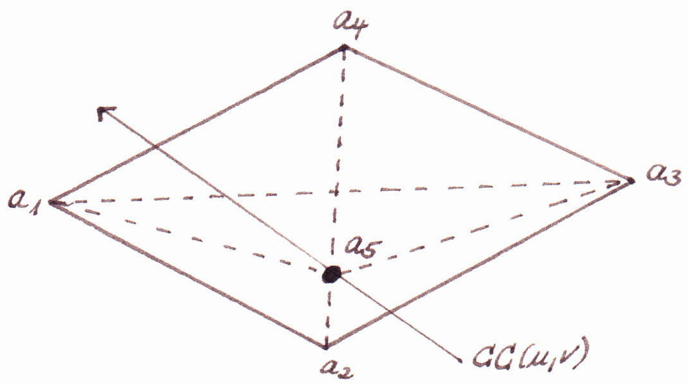


Fig. 5

3. Examination of the Shadow Vertex Algorithm Within a Degenerate Boundary Polytope

We now know the geometrical background of degeneracy. Let us examine, whether the shadow vertex algorithm keeps its distinctness under degeneracy, and whether cycling is possible. For the shadow vertex algorithm guarantees that the objective function grows from vertex to vertex, cycling can (if at all) arise only within a vertex (primal) or a boundary polytope (polar). That is why we concentrate on the examination of one single boundary polytope, which is spanned by $p > n$ restriction vectors.

We want to remark that the shadow vertex algorithm does not necessarily work definite on degenerate problems, as it is the case with non-degenerate ones. Ambiguities can arise with the choice of the leaving vector as well as with the entering vector. Let us discuss this phenomenon by looking at (iii) of fig. 5.

Assume the present basis is (a_2, a_3, a_5). The next basis can be (a_3, a_4, a_5) or (a_1, a_2, a_5), which makes clear that the leaving vector is not unique. After the choice of the leaving vector, say a_2, as well (a_1, a_3, a_5) as (a_3, a_4, a_5) is a suitable basis (ambiguity of the entering vector!). The same argumentation is valid for (ii), (v), and (vi).

For the degeneracy class $D(\cdot, \cdot, n-1)$ at least the leaving vector is determined uniquely (compare (i) and (iv)). This insight leads to the following lemmata, which can be proved easily.

Lemma 4: *The shadow vertex algorithm works cycle free on problems of the class $D(p, \cdot, n-1)$.*

Lemma 5: *For all $D(p, \cdot, n-i)$-problems, $i \geq 1$, ambiguities concerning the entering vector may occur.*

Remark: The ambiguity of the entering vector has the following geometrical interpretation. The leaving vector $\bar{a}_i$ lies in one of the two halfspaces divided by the hyperplane through the other basis vectors $\{\bar{a}_1, \ldots, \bar{a}_n\} \setminus \{\bar{a}_i\}$ and the origin. Now there are several vectors out of $\{a_{\Delta^1}, \ldots, a_{\Delta^p}\} \supset \{\bar{a}_1, \ldots, \bar{a}_n\}$ situated in the other halfspace. These are all possible candidates for entering the basis.

We have all seen that the distinctness of the shadow vertex algorithm is lost with degenerate problems. We therefore have to give an instruction how to handle ambiguities.

An obvious modification of the shadow vertex algorithm with degenerate problems is the following:

(i) We apply the shadow vertex rule as long as distinction is guaranteed.

(ii) If there are any ambiguities, then we choose the leaving or entering vector arbitrarily among the possible vectors.

This selection rule is only sensible, if the so modified shadow vertex algorithm works cycle-free *. We want to examine that in the following section.

4. Construction of Cycling Examples

At the present we conjecture, that for all degeneracy classes $D(p,\cdot,n-i)$, $p>n$, $i\geq 2$ there exist cycling examples. But it is impossible for us to create one for the class $D(p,\cdot,n-2)$ in $\mathbf{R}^3$. That is why we pass over to $\mathbf{R}^4$. We exert ourselves for choosing the input data of the cycling examples as simple as possible to make the calculations easy. Since the generating vectors of the boundary polytope ($n-1$-dimensional, that means 3-dimensional in $\mathbf{R}^4$!) lie all in one hyperplane of $\mathbf{R}^4$, we can fix their fourth component to 1. We design a $D(\cdot,\cdot,1)$-cycling example, that means there exists a point say a_{Δ^4}, the so-called critical point, which is a conical combination of u and v.

$$\begin{aligned} a_{\Delta^4} &:= (0,0,\epsilon,1)^T,\ \epsilon>0, \quad u:=(0,0,\bar{u},1)^T, \quad \bar{u}<\epsilon \quad \text{and} \\ v &:= (0,0,\bar{v},1)^T,\ \bar{v}>\epsilon \quad \text{yields} \quad a_{\Delta^4}:=\lambda_1 u+\lambda_2 v \quad \text{with} \\ \lambda_1 &= 1-\lambda_2, \quad \lambda_2=\frac{\epsilon-\bar{u}}{\bar{v}-\bar{u}}, \quad 0<\lambda_1,\lambda_2<1, \end{aligned} \tag{4.1}$$

and we get what we wished.

For simplicity we choose the third component of the remaining vectors of the boundary polytope $\Sigma(\Delta)$ equal to zero, their first two entries can be chosen arbitrarily.

Geometrically speaking we search a circle of basis, such that the following holds:

In every pivot step a set of $n-1$ basis vectors $\{\tilde{a}_1,\ldots,\tilde{a}_{n-1}\}\subset\{a_{\Delta^1},\ldots,a_{\Delta^n}\}$ stays in the basis, whose linear hull separates the vector $\tilde{a}_n:=\{a_{\Delta^1},\ldots,a_{\Delta^n}\}\setminus\{\tilde{a}_1,\ldots,\tilde{a}_{n-1}\}$ and the objective vector v. $\tilde{a}_n$ is then replaced by a vector $\hat{a}\in\{a_1,\ldots,a_m\}\setminus\{a_{\Delta^1},\ldots,a_{\Delta^n}\}$, which lies in the same halfspace as v relative to $span(\tilde{a}_1,\ldots,\tilde{a}_{n-1})$. (4.2)

Now we search for a sequence of basis with the above property and the additional condition, that after a finite number of steps we get a basis already obtained. For this purpose the critical point a_{Δ^4} must not leave the basis, for that would mean a step forward on the $span(u,v)$-plane into direction v, and hence no cycle would be possible.

Cycling examples shall now be designed on a piece of paper. Therefore we have to translate our previous ideas into $\mathbf{R}^2$, especially prescription (4.2).

* For simplicity we will omit the word "modified" in the following. In connection with degenerate problems, however, it would have to be added.

From up to now by $\bar{a}_{\Delta^i} \in \mathbf{R}^2$ we denote the 4-dimensional basis vectors a_{Δ^i} truncated by their last two components. If we choose the input vectors as described in (4.1), we observe the following phenomenon:

Theorem 1: *Let $\Delta = \{a_{\Delta^1}, \ldots, a_{\Delta^4}\}$ be a basis of $\mathbf{R}^4$ and a_{Δ^4} a critical point. a_{Δ^1} is a possible leaving vector, iff $\bar{a}_{\Delta^1}$ lies in the same halfplane as $\bar{a}_{\Delta^4}$ relative to the straight line containing $\bar{a}_{\Delta^2}$ and $\bar{a}_{\Delta^3}$.*

Proof. Let r be the normal vector on the hyperplane $H(0, a_{\Delta^2}, a_{\Delta^3}, a_{\Delta^4})$

$$\rightarrow \quad r = (\epsilon(a^2_{\Delta^3} - a^2_{\Delta^2}), -\epsilon(a^1_{\Delta^3} - a^1_{\Delta^2}), a^2_{\Delta^3}a^1_{\Delta^2} - a^2_{\Delta^2}a^1_{\Delta^3}, \epsilon(a^2_{\Delta^2}a^1_{\Delta^3} - a^2_{\Delta^3}a^1_{\Delta^2}))^T$$

with ϵ as in (4.1). The normal vector on the straight line containing $\bar{a}_{\Delta^2}$ and $\bar{a}_{\Delta^3}$ is then

$$\bar{r} = (\epsilon(a^2_{\Delta^3} - a^2_{\Delta^2}), -\epsilon(a^1_{\Delta^3} - a^1_{\Delta^2}))^T$$

$\Longrightarrow$

a_{Δ^1} is a possible leaving vector $\rightarrow sgn(r^T a_{\Delta^1}) = -sgn(r^T v)$

$$\leftrightarrow \quad sgn[(a^2_{\Delta^2}a^1_{\Delta^3} - a^1_{\Delta^2}a^2_{\Delta^3}) + (a^2_{\Delta^3} - a^2_{\Delta^2})a^1_{\Delta^1} - (a^1_{\Delta^3} - a^1_{\Delta^2})a^2_{\Delta^1}] = -sgn[a^1_{\Delta^2}a^2_{\Delta^3} - a^2_{\Delta^2}a^1_{\Delta^3}] \tag{4.3}$$

$$\bar{r}^T \bar{a}_{\Delta^2} = \epsilon(a^2_{\Delta^3}a^1_{\Delta^2} - a^1_{\Delta^3}a^2_{\Delta^2}) =: c \neq 0 \tag{4.4}$$

It remains to show:

$$\bar{r}^T \bar{a}_{\Delta^4} < c \quad \text{and} \quad \bar{r}^T \bar{a}_{\Delta^1} < c \tag{4.5}$$

or

$$\bar{r}^T \bar{a}_{\Delta^4} > c \quad \text{and} \quad \bar{r}^T \bar{a}_{\Delta^1} > c \tag{4.6}$$

We know:

$$\bar{r}^T \bar{a}_{\Delta^4} = 0$$
$$\bar{r}^T \bar{a}_{\Delta^1} = \epsilon(a^2_{\Delta^3} - a^2_{\Delta^2})a^1_{\Delta^1} + \epsilon(a^1_{\Delta^2} - a^1_{\Delta^3})a^2_{\Delta^1}$$

We have to distinguish between $c > 0$ and $c < 0$.

1. $c > 0$

Let us show (4.5). From (4.3) and (4.4) follows:

$$(a^2_{\Delta^2}a^1_{\Delta^3} - a^1_{\Delta^2}a^2_{\Delta^3}) + (a^2_{\Delta^3} - a^2_{\Delta^2})a^1_{\Delta^1} - (a^1_{\Delta^3} - a^1_{\Delta^2})a^2_{\Delta^1} < 0$$

$$\leftrightarrow \underbrace{\epsilon(a^2_{\Delta^3}a^1_{\Delta^2} - a^2_{\Delta^2}a^1_{\Delta^3})}_{c} > \underbrace{\epsilon(a^2_{\Delta^3} - a^2_{\Delta^2})a^1_{\Delta^1} - \epsilon(a^1_{\Delta^3} - a^1_{\Delta^2})a^2_{\Delta^1}}_{\bar{r}^T \bar{a}_{\Delta^1}}$$

2. $c < 0$

Let us show (4.6). From (4.3) and (4.4) follows:

$$(a^2_{\Delta^2} a^1_{\Delta^3} - a^1_{\Delta^2} a^2_{\Delta^3}) + (a^2_{\Delta^3} - a^2_{\Delta^2}) a^1_{\Delta^1} - (a^1_{\Delta^3} - a^1_{\Delta^2}) a^2_{\Delta^1} > 0$$
$$\leftrightarrow \underbrace{\epsilon(a^2_{\Delta^3} a^1_{\Delta^2} - a^2_{\Delta^2} a^1_{\Delta^3})}_{c} < \underbrace{\epsilon(a^2_{\Delta^3} - a^2_{\Delta^2}) a^1_{\Delta^1} - \epsilon(a^1_{\Delta^3} - a^1_{\Delta^2}) a^2_{\Delta^1}}_{\bar{r}^T \bar{a}_{\Delta^1}}$$

$\Longleftarrow$

$$sgn(\bar{r}^T \bar{a}_{\Delta^1}) = sgn(\bar{r}^T \bar{a}_{\Delta^4})$$

$$\leftrightarrow sgn(a^1_{\Delta^1} \epsilon(a^2_{\Delta^3} - a^2_{\Delta^2}) - a^2_{\Delta^1} \epsilon(a^1_{\Delta^3} - a^1_{\Delta^2})) = sgn(0) = 0$$
$$\leftrightarrow a^1_{\Delta^1}(a^2_{\Delta^3} - a^2_{\Delta^2}) - a^2_{\Delta^1}(a^1_{\Delta^3} - a^1_{\Delta^2}) = 0 \qquad (4.7)$$

$$\begin{aligned} sgn(r^T a_{\Delta^1}) &= sgn[(a^2_{\Delta^2} a^1_{\Delta^3} - a^1_{\Delta^2} a^2_{\Delta^3}) + (a^2_{\Delta^3} - a^2_{\Delta^2}) a^1_{\Delta^1} - (a^1_{\Delta^3} - a^1_{\Delta^2}) a^2_{\Delta^1}] \overset{(4.7)}{=} \\ &= sgn(a^2_{\Delta^2} a^1_{\Delta^3} - a^1_{\Delta^2} a^2_{\Delta^3}) = \\ &= -sgn(a^1_{\Delta^2} a^2_{\Delta^3} - a^2_{\Delta^2} a^1_{\Delta^3}) = \\ &= -sgn(r^T v) \end{aligned}$$

$\square$

Using all previous results, we now can specify cycling examples of the class $D(p, \cdot, n-3)$ $(= D(p, \cdot, 1))$ in the space $\mathbf{R}^4$. We consider only a small selection, which should clarify the geometry.

Fig. 6–8 show boundary polytopes of this kind with regular shape. Since we operate in $\mathbf{R}^4$, the boundary polytopes are 3-dimensional (facets!). We consider such a boundary polytope $\Sigma(\Delta)$ where $\Sigma(\Delta) = Y \cap \{x \mid x^4 = 1\}$. This means, that the boundary polytope has distance 1 from the origin in x^4-direction. Furthermore all points of the boundary polytope but the critical one should span one of the polytope's facets. These points are marked by a black dot in fig. 6–8, and have a zero-entry in their third component. This special position in direction x^3 is not absolutly necessary for constructing cycling examples, but it simplifies the graphical as well as the computational representation. The critical point, which has to stay in the basis during the cycle is marked by a cross. It lies on $span(u, v)$ and it's third component x^3 is equal to ϵ with $\epsilon > 0$.

Hence what we see in our figures is the two-dimensional projection (onto the first two components x^1 and x^2) of the three-dimensional boundary polytope in the 4-dimensional space.

The principle behind the construction of all three figures is the following: Let us ignore the critical point for a while, since it does not leave the basis during the whole cycle, and let us concentrate on the remaining three basis vectors. Because of our choice, they lie in the two-dimensional plane and form triangles (linear independence!). According to theorem 1 we have to remove that vertex of the triangle from the basis, which lies in the same halfplane as the critical point relative to the straight line containing the other two vertices of the triangle. This point is then replaced by a point of the opposite halfplane. After a finite number of exchanges of this kind we have to return to the start triangle to get a cycle.

$$\Sigma(\Delta) = CH(a_1, \dots, a_7)$$

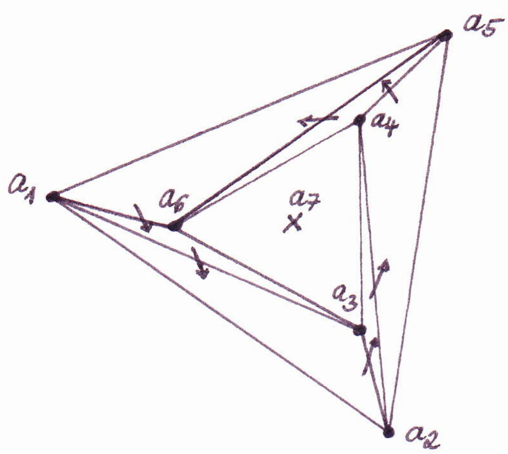


Fig. 6

$$\Sigma(\Delta) = CH(a_1, \dots, a_9)$$

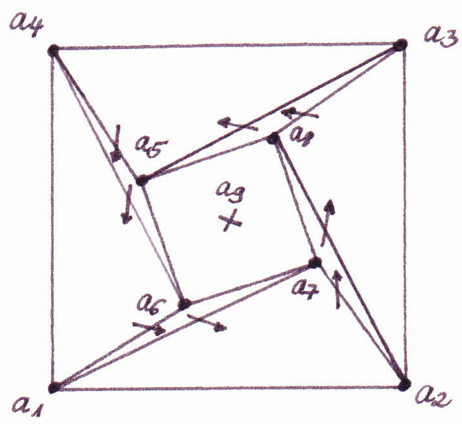


Fig. 7

$$\Sigma(\Delta) = CH(a_1, \ldots, a_{13})$$

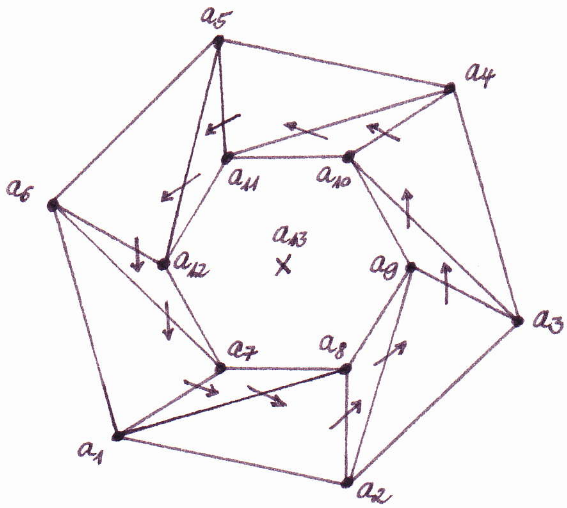


Fig. 8

From appendix I–III one can draw the calculations to the fig. 6–8.

We now discuss a cycling example related to fig. 6 at full length. Therefore we assign concrete numbers to the vectors, such that we qualitatively get fig. 6. Afterwards we explain by this figure the construction of a cycle. Last but not least we show the cycling by explicit calculation of the tableaus.

Let

$$\begin{aligned} a_1 &:= (-4.5, -2.25, 0, 1)^T, \quad a_2 := (6, -4.75, 0, 1)^T, \\ a_3 &:= (2, -1.25, 0, 1)^T, \quad a_4 := (0, 2.75, 0, 1)^T, \\ a_5 &:= (0.5, 6.25, 0, 1)^T, \quad a_6 := (-2, -1.25, 0, 1)^T, \\ a_7 &:= (0, 0, 1, 1)^T, \\ v &:= (0, 0, -1, 1)^T, \quad u := (0, 0, 2, 1)^T. \end{aligned}$$

This assignment yields a somehow distorted image of fig. 6 which however owns the same geometric properties and is shown in fig. 9. It has the advantage, that during the calculation (exact, since we use integer arithmetic!) only numbers with moderate size appear. We start with the basis $\Delta = \{3, 4, 2, 7\}$ and get the following tableau restricted to the boundary polytope:

$$\Sigma(\Delta) = CH(a_1, \ldots, a_7)$$

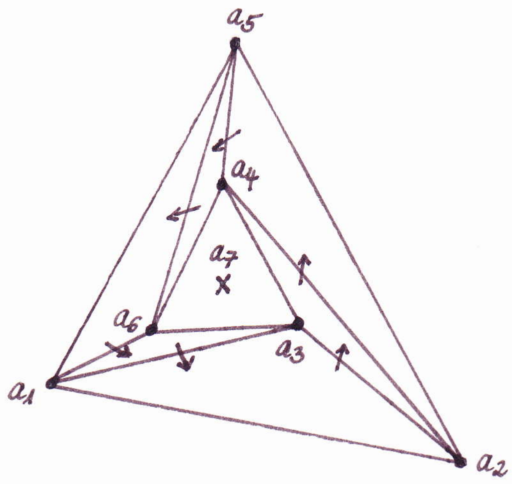


Fig. 9

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	v	u
a_3	85/12	0	1	0	−11/4	13/3	0	−11/6	11/3
a_4	−107/36	0	0	1	11/4	−14/9	0	2/9	−4/9
a_2	−28/9	1	0	0	1	−16/9	0	11/18	−11/9
a_7	0	0	0	0	0	0	1	2	−1
	0	0	0	0	0	0	0	−1	−1

Only a_3 can leave the basis. $span(a_4, a_2, a_7)$ separates v and a_3. According to theorem 1 that is equivalent to the fact that in the two-dimensional projection the straight line containing $\bar{a}_2$ and $\bar{a}_4$ does **not** separate $\bar{a}_3$ and $\bar{a}_7$. During the cycle a_7 never leaves the basis.

From up to now for the leaving vector all those vectors come into question, which lie in the same halfplane as $\bar{a}_7$ relative to the straight line containing the remaining basis vectors. Let a_5 be the entering vector. Then the tableau looks as follows:

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	v	u
a_5	−85/33	0	−4/11	0	1	−52/33	0	2/3	4/3
a_4	37/9	0	1	1	0	25/9	0	−29/18	29/9
a_2	−53/99	1	4/11	0	0	−20/99	0	−1/18	1/9
a_7	0	0	0	0	0	0	1	2	−1
	0	0	0	0	0	0	0	−1	−1

a_4 or a_2 may leave the basis. $\bar{a}_4$ lies namely in the same halfplane as $\bar{a}_7$ relative to $G_1 := \{x \mid x = \bar{a}_2 + \lambda(\bar{a}_5 - \bar{a}_2),\ \lambda \in \mathbf{R}\}$, and $\bar{a}_2$ lies in the same halfplane as $\bar{a}_7$ relative to $G_2 := \{x \mid x = \bar{a}_4 + \lambda(\bar{a}_5 - \bar{a}_4),\ \lambda \in \mathbf{R}\}$. We choose a_2 as leaving and a_6 as entering vector.

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	v	u
a_5	8/5	−39/5	−16/5	0	1	0	0	11/10	−11/5
a_4	−13/4	55/4	6	1	0	0	0	−19/8	19/4
a_6	53/20	−99/20	−9/5	0	0	1	0	11/40	−11/20
a_7	0	0	0	0	0	0	1	2	−1
	0	0	0	0	0	0	0	−1	−1

Let a_4 be the leaving vector, for $\bar{a}_7$ and $\bar{a}_4$ lie in the same halfplane relative to $G_3 := \{x \mid x = \bar{a}_5 + \lambda(\bar{a}_6 - \bar{a}_5),\ \lambda \in \mathbf{R}\}$. Let a_1 be the entering vector.

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	v	u
a_5	0	−67/65	−16/65	32/65	1	0	0	−9/130	9/65
a_1	1	−55/13	−24/13	−4/13	0	0	0	19/26	−19/13
a_6	0	407/65	201/65	53/65	0	1	0	−108/65	216/65
a_7	0	0	0	0	0	0	1	2	−1
	0	0	0	0	0	0	0	−1	−1

1 and 3 are possible pivot rows, since $\bar{a}_5$ and $\bar{a}_7$ lie in the same halfplane relative to $G_4 = \{x \mid x = \bar{a}_1 + \lambda(\bar{a}_6 - \bar{a}_1),\ \lambda \in \mathbf{R}\}$. The same is true for $\bar{a}_6$ and $\bar{a}_7$ and the straight line defined by $\bar{a}_1$ and $\bar{a}_5$. Let us choose row 1 and column 3 to perform a pivot step. We get:

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	v	u
a_3	0	67/16	1	−2	−65/16	0	0	9/32	−9/16
a_1	1	7/2	0	−4	−15/2	0	0	5/4	−5/2
a_6	0	−107/16	0	7	201/16	1	0	−81/32	81/16
a_7	0	0	0	0	0	0	1	2	−1
	0	0	0	0	0	0	0	−1	−1

According to the scheme repeatedly described we now choose a_6 and a_2 and get:

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	v	u
a_3	0	0	1	255/107	407/107	67/107	0	−279/214	279/107
a_1	1	0	0	−36/107	−99/107	56/107	0	−8/107	16/107
a_2	0	1	0	−112/107	−201/107	−16/107	0	81/214	−81/107
a_7	0	0	0	0	0	0	1	2	−1
	0	0	0	0	0	0	0	−1	−1

Assume we choose a_1 as the leaving vector, and a_4 as the entering vector, then we get the basis $\Delta = \{3, 4, 2, 7\}$, i.e. the start basis. That means we have constructed a cycle (also compare the calculations in the appendix).

Now it is obvious how to construct $D(p, \cdot, 1)$-cycling examples in higher dimensions $n \geq 5$.

(4.8)

Step 1: Design a $D(p, \cdot, 1)$-cycling example of $\mathbf{R}^4$ as already explained.

Step 2: To get a cycling example in the n-dimensional space, extend the 4-dimensional vectors by $n-4$ entries with value zero. In addition a linearly independent completion of the $n-4$ vectors is needed. For that purpose we choose (w.l.o.g. $x^4 = 1$ for all points of the boundary polytope as in our examples)

$$a_1^* := (0, 0, 0, 1, e_1^{n-4})^T,$$
$$a_2^* := (0, 0, 0, 1, e_2^{n-4})^T,$$
$$\vdots$$
$$a_{n-4}^* := (0, 0, 0, 1, e_{n-4}^{n-4})^T,$$

where e_i^{n-4} is the i^{th} unit vector of dimension $n-4$.

Step 3: For the start tableau choose $\beta_1, \ldots, \beta_4$ and $\alpha_1, \ldots, \alpha_4$ like in the 4-dimensional start tableau and $\alpha_5 = \cdots = \alpha_n := \alpha_1$, $\beta_5 = \cdots = \beta_n := \beta_1$. Again a_{Δ^4} is the vector not leaving the basis during the whole cycle.

Remark: By pivoting on the vectors a_{Δ^1}, a_{Δ^2}, and a_{Δ^3} we can produce a cycle analogous to the 4-dimensional case (compare proof of lemma 6).

In the next step we generalize the degree of degeneracy, i.e. we design $D(p, \cdot, i)$-cycling examples, where $1 \leq i \leq n-3$, and the dimension n of the space is arbitrarily chosen.

(4.9)

This construction is the same as in (4.8) besides Step 3. Instead of we perform Step 3'.

<u>Step 3'</u>: For the start tableau choose $\beta_1, \ldots, \beta_4$ and $\alpha_1, \ldots, \alpha_4$ like in the 4-dimensional start tableau. Besides $\alpha_j := \alpha_1$ and $\beta_j := \beta_1$ for all $j = 5, \ldots, n-i+1$, $\alpha_j := \alpha_4$, $\beta_j := \beta_4$ otherwise.

Remark:

1. By pivoting on the vectors a_{Δ^1}, a_{Δ^2} and a_{Δ^3} we can again produce a cycle analogous to the 4-dimensional case (compare proof of lemma 6).

2. In the start tableau we have $\alpha_j > 0$ at most $n-3$ times (compare Step 3' and the construction of the 4-dimensional cycling example). The corresponding basis vectors have to stay in the basis. The most harmless degree of degeneracy, we can produce by this construction, is therefore $D(p, \cdot, n-3)$.

Lemma 6: *(4.8) and (4.9) produce cycling examples of the desired degeneracy classes.*

Proof. We try to construct a $D(p, \cdot, i)$-cycling example in the space of dimension n. The start tableau looks according to (4.8) and (4.9) as follows:

$$
\begin{array}{c|cccccc|c|c}
 & a_1 & \cdots & a_m & a_1^* & \cdots & a_{n-4}^* & v & u \\
\hline
a_{\Delta^1} & & & & & & & \alpha_1 & \beta_1 \\
a_{\Delta^2} & & & & & & & \alpha_2 & \beta_2 \\
a_{\Delta^3} & & & & & 0 & & \alpha_3 & \beta_3 \\
a_{\Delta^4} & & (\gamma_{ij})_{\substack{i=1,\ldots,n \\ j=1,\ldots,m}} & & & & & \alpha_4 & \beta_4 \\
\cline{8-9}
a_1^* & & & & 1 & & & \alpha_1 & \beta_1 \\
\vdots & & & & & & & \vdots & \vdots \\
a_{n-i-3}^* & & & & & \ddots & & \alpha_1 & \beta_1 \\
a_{n-i-2}^* & & & & & & & \alpha_4 & \beta_4 \\
\vdots & & & & & & & \vdots & \vdots \\
a_{n-4}^* & & & & & & 1 & \alpha_4 & \beta_4 \\
\hline
 & 0 & & & \cdots & & 0 & Q_v & Q_u
\end{array}
$$

The second part of the v-column contains $i-1$ entries of value α_4 and $n-i-3$ entries of value α_1.

Let $a_1, \ldots, a_m$ be the vectors we get from the 4-dimensional cycling example extended by $n-4$ entries of value zero. a_{Δ^4} is the vector, which does not leave the basis during the cycle. Let $a_1^*, \ldots, a_{n-4}^*$ be the linearly independent completion of the $\mathbf{R}^n$. They will never leave the basis since they are absolutely necessary for it (no negative γ-entry in this part of the tableau!).

Note that $\mathbf{R}^n$ is the direct sum of $span(a_1, \ldots, a_m)$ and $span(a_1^*, \ldots, a_{n-4}^*)$. In the first four lines the entries of $a_1, \ldots, a_m$ are therefore the same as in the 4-dimensional case, because this upper part of the tableau is only generated by $a_1, \ldots, a_m$. The same is true for the corresponding part of the columns u and v. This implies that the cycles produced by $a_1, \ldots, a_m$ in the 4-dimensional case also lead to cycles in $\mathbf{R}^n$.

Let $\lambda = 1$ and $\mu = -\frac{\beta_1}{\alpha_1} = -\frac{\beta_2}{\alpha_2} = -\frac{\beta_3}{\alpha_3} > 0$.
Then

$$\lambda u + \mu v = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_1 \\ \vdots \\ \beta_1 \\ \beta_4 \\ \vdots \\ \beta_4 \end{pmatrix} + \begin{pmatrix} -\beta_1 \\ -\beta_2 \\ -\beta_3 \\ -\frac{\alpha_4\beta_1}{\alpha_1} \\ -\beta_1 \\ \vdots \\ -\beta_1 \\ -\frac{\alpha_4\beta_1}{\alpha_1} \\ \vdots \\ -\frac{\alpha_4\beta_1}{\alpha_1} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \beta_4 - \frac{\alpha_4\beta_1}{\alpha_1} \\ 0 \\ \vdots \\ 0 \\ \beta_4 - \frac{\alpha_4\beta_1}{\alpha_1} \\ \vdots \\ \beta_4 - \frac{\alpha_4\beta_1}{\alpha_1} \end{pmatrix}$$

Therefore the vector $\lambda u + \mu v$ can be represented by i basis-vectors, and this means that the degeneracy class $D(p, \cdot, i)$ is at hand. $\square$

Now we know, that the shadow vertex algorithm works cycle-free on $D(p, \cdot, n-1)$-problems, but there exist cycling examples for the classes $D(p, \cdot, n-i)$, $i \geq 3$. It remains the question whether cycling for the class $D(p, \cdot, n-2)$ is possible. In the following we will prove the finiteness of the shadow vertex algorithm for this kind of problems. For that purpose we need some more lemmata.

Lemma 7: *Assume a problem of the class $D(p, \cdot, n-2)$, $p > n$ is at hand. Then there exist $i, j \in \{1, \ldots, n\}$, $i \neq j$ with*

$$\alpha_i < 0,\ \alpha_j < 0 \quad -\frac{\beta_i}{\alpha_i} = -\frac{\beta_j}{\alpha_j} \quad \text{and}$$

$$-\frac{\beta_i}{\alpha_i} < -\frac{\beta_k}{\alpha_k} \quad \forall k \in \{1, \ldots, n\} \setminus \{i, j\} \quad \text{where } \alpha_k < 0.$$

Proof. According to the definition of $D(p, \cdot, n-2)$, $span(u, v)$ intersects the present cone $CC(a_{\Delta^1}, \ldots, a_{\Delta^n})$ in a $(n-2)$-dimensional face.

$$\rightarrow\ \exists i, j, \in \{1, \ldots, n\},\ i \neq j \quad \text{and} \quad \exists\ \underline{\lambda},\ \bar{\lambda} \in \mathbf{R}^+ \quad \text{with}$$

$$\beta_k + \lambda\alpha_k > 0 \quad \forall\ \lambda \in (\underline{\lambda},\ \bar{\lambda}] \quad \forall\ k \in \{1, \ldots, n\} \setminus \{i, j\}$$

$$\begin{aligned} \beta_i + \bar{\lambda}\alpha_i &= 0 \\ \beta_j + \bar{\lambda}\alpha_j &= 0 \end{aligned} \tag{4.10}$$

$$\rightarrow \bar{\lambda} = -\frac{\beta_i}{\alpha_i} = -\frac{\beta_j}{\alpha_j}$$

$$(\alpha_i,\ \alpha_j \neq 0 \quad \text{comp. (2.1)!})$$

$$\text{and} \quad \bar{\lambda} < -\frac{\beta_k}{\alpha_k} \quad \forall\, k \in (\{1,\ldots,n\} \setminus \{i,j\}) \cap \{k \mid \alpha_k < 0\}$$

We now show $\alpha_i < 0,\ \alpha_j < 0$.

$$\begin{aligned} \beta_i + \lambda\alpha_i \geq 0 \quad \forall\, \lambda \in (\underline{\lambda},\, \bar{\lambda}) \\ \beta_j + \lambda\alpha_j \geq 0 \quad \forall\, \lambda \in (\underline{\lambda},\, \bar{\lambda}) \end{aligned}$$

Assume $\alpha_i > 0 \ \rightarrow \ \forall \epsilon > 0$ the following holds:

$$\beta_i + (\lambda + \epsilon)\alpha_i = \underbrace{\beta_i + \lambda\alpha_i}_{\geq 0} + \underbrace{\epsilon\alpha_i}_{>0} > 0$$

This is a contradiction to (4.10). For α_j we argue analogously. □

Lemma 8: *We assume (2.1). For 2−dimensional problems there can only be an ambiguity in the first pivot step choosing the leaving vector. Afterwards each leaving vector is definitely determined.*

Proof. In the following we assume that neither unboundedness nor optimality is achieved, since otherwise we could stop. $\rightarrow \exists\ \alpha_i < 0,\ i \in \{1,2\}$.
The only critical case is, if $\alpha_1 < 0$ and $\alpha_2 < 0$ besides

$$-\frac{\beta_1}{\alpha_1} = -\frac{\beta_2}{\alpha_2} =: \lambda, \qquad \lambda > 0.$$

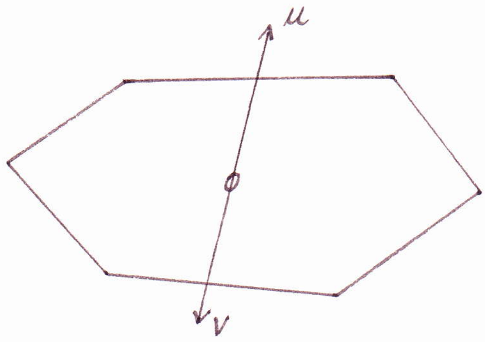
Otherwise the leaving vector is at hand.
Let

$$-\frac{\beta_1}{\alpha_1} = -\frac{\beta_2}{\alpha_2} =: \lambda, \qquad \lambda > 0.$$

$$\begin{aligned} \rightarrow \beta_1 &= (-\lambda)\alpha_1 \\ \beta_2 &= (-\lambda)\alpha_2 \qquad \rightarrow \beta_1 > 0,\ \beta_2 > 0 \end{aligned}$$

$\rightarrow$ u is a negative multiple of v, i.e. u and v are linearly dependent. The situation is as follows:

$Y \subset \mathbf{R}^2$



Here we are in the start simplex, i.e. in the first pivot step. If we now choose the pivot row arbitrarily (w.l.o.g. the first row) the up-date achieves that $\alpha_1 > 0$. Then either optimality or one of the harmless cases is at hand. $\square$

Lemma 9: *The orthogonal projection Π of the vectors $a_1, \ldots, a_m, u, v$ onto the orthogonal complement of span$(a_{\Delta^{k+1}}, \ldots, a_{\Delta^n})$ for a $k \in \{1, \ldots, n\}$ does not change the entries of the first k tableau lines.*

Proof.

$$\Pi(a_{\Delta^{k+1}}) = \cdots = \Pi(a_{\Delta^n}) = 0$$

$$a_i = \sum_{j=1}^{n} \gamma_{ji} a_{\Delta^j}$$

$$\rightarrow \Pi(a_i) = \Pi\left(\sum_{j=1}^{n} \gamma_{ji} a_{\Delta^j}\right) = \sum_{j=1}^{n} \gamma_{ji} \Pi(a_{\Delta^j}) = \sum_{j=1}^{k} \gamma_{ji} \Pi(a_{\Delta^j})$$

For the columns u and v we argue analogously. $\square$

Lemma 10: *By pivoting only in two fixed rows no cycle can arise.*

Proof. W.l.o.g. we consider only the first and the second row of our tableaus as well as the slack part. This shortened tableau is then interpreted as the tableau of a 2−dimensional linear program. Besides we restrict ourselves on the columns, which have a zero entry in the slack part, i.e. on the p, $p > n$ generating vectors of the boundary polytope, for only these (if at all) can produce a cycle. We then get a start tableau of the form:

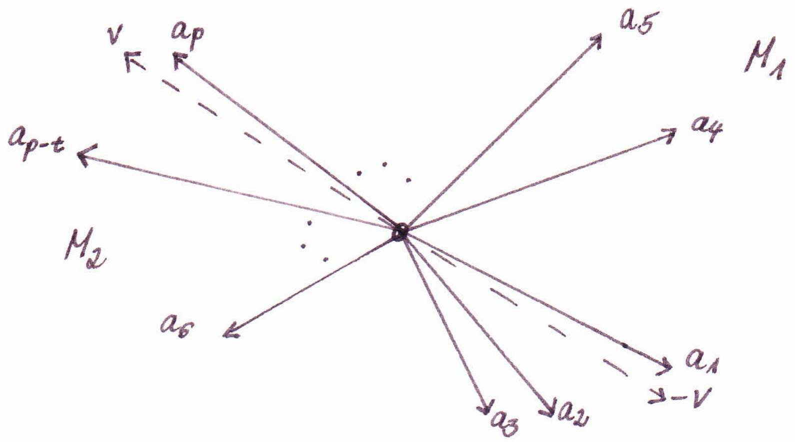
	a_1	$\cdots$	a_k	$\cdots$	a_l	$\cdots$	a_p	v	u
a_k								α_1	β_1
				γ_{ij}					
a_l								α_2	β_2
	0			$\cdots$			0	Q_v	Q_u

W.l.o.g. we choose a_k as the leaving vector. In the next pivot step it will be replaced by $a_{k'}$. According to lemma 9 we can interpret the columns of the tableau as vectors in $\mathbf{R}^2$ represented by the canonical basis. Assume (w.l.o.g.) that the vectors are numbered such that

$$0^\circ \leq arc(-v, a_i) < arc(-v, a_{i+1}) \leq 180^\circ \quad \forall i \in \{1, \dots, p\}.$$

$G := \{\lambda v,\ \lambda \in \mathbf{R}\}$ divides the $\mathbf{R}^2$ into two halfplanes denoted by M_1 and M_2.

$t \in \{1, \dots, p-7\}$



We have to distinguish two cases.

Case 1: $a_{k'} \in M_1 \wedge a_l \in M_2 \quad \vee \quad a_{k'} \in M_2 \wedge a_l \in M_1$
$G' := \{\mu a_l,\ \mu \in \mathbf{R}\}$ divides the $\mathbf{R}^2$ into two halfspaces, one containing $-v$ and the other containing v. Then $a_{k'}$ lies in that halfspace, which contains v. According to our assumption of case 1, $v \in CC(a_{k'}, a_l)$, and we can stop.

Case 2: $a_{k'}, a_l \in M_1 \quad \vee \quad a_{k'}, a_l \in M_2$
$\rightarrow \quad arc(-v, a_{k'}) > arc(-v, a_l) \quad \rightarrow \quad k' > l$

As long as case 2 is at hand, we must repeat the procedure increasing the indices of the entering vector. Therefore case 2 can occur at most $p-1$ times. Hence after at most p pivot steps case 1 must apply. This means that cycling is excluded. $\square$

Lemma 11: *Consider a $D(p, \cdot, n-2)$-problem where (w.l.o.g.) $u + \bar{\mu}v = \lambda_3 a_{\Delta^3} + \dots + \lambda_n a_{\Delta^n} =: P_1$, $\lambda_3, \dots, \lambda_n > 0$. Then none of the vectors $a_{\Delta^3}, \dots, a_{\Delta^n}$ can be removed from the basis without leaving P_1, which results in an increase of μ.*

Proof.

$$P_1 = u + \bar{\mu}v = \lambda_3 a_{\Delta^3} + \dots + \lambda_n a_{\Delta^n}$$
$$P_2 := u + \tilde{\mu}v = \eta_1 a_{\Delta^1} + \eta_2 a_{\Delta^2} + \eta_4 a_{\Delta^4} + \dots + \eta_n a_{\Delta^n}$$

W.l.o.g. we remove a_{Δ^3} from the basis.

Assume $\bar{\mu} = \tilde{\mu} \quad (\leftrightarrow \quad u + \bar{\mu}v = u + \tilde{\mu}v \quad \leftrightarrow \quad P_1 = P_2)$

$$\rightarrow \lambda_3 a_{\Delta^3} + \cdots + \lambda_n a_{\Delta^n} = \eta_1 a_{\Delta^1} + \eta_2 a_{\Delta^2} + \eta_4 a_{\Delta^4} + \cdots + \eta_n a_{\Delta^n}$$

$$\leftrightarrow \eta_1 a_{\Delta^1} + \eta_2 a_{\Delta^2} - \lambda_3 a_{\Delta^3} + (\eta_4 - \lambda_4) a_{\Delta^4} + \cdots + (\eta_n - \lambda_n) a_{\Delta^n} = 0$$

$$\rightarrow \eta_1 = \eta_2 = \lambda_3 = 0, \quad \eta_4 = \lambda_4, \ldots, \eta_n = \lambda_n,$$

because $\{a_{\Delta^1}, \ldots, a_{\Delta^n}\}$ is a basis of $\mathbf{R}^n$. Since $\lambda_3 = 0$ we do not have a $D(p, \cdot, n-2)$-problem but a $D(p, \cdot, n-3)$-problem. $\rightarrow \ \bar{\mu} \neq \tilde{\mu} \ \rightarrow \ \tilde{\mu} > \bar{\mu}$. $\square$

Theorem 2: *The shadow vertex algorithm works cycle-free on $D(p, \cdot, n-2)$-problems.*

Proof. If at all cycles can arise, then arriving at P_1 (comp. lemma 11). $P_1 = u + \bar{\mu}v = \lambda_3 a_{\Delta^3} + \cdots + \lambda_n a_{\Delta^n}$ where $\lambda_1, \ldots, \lambda_n > 0$.

Assume there is a cycle around P_1. According to lemma 11 none of the vectors $a_{\Delta^3}, \ldots, a_{\Delta^n}$ is allowed to leave the basis during the cycle, since otherwise P_1 would be left. Therefore the cycle has to be produced by pivoting in the first two lines of the tableau. But this is a contradiction to lemma 10. $\square$

Corollary: *In $\mathbf{R}^3$ the shadow vertex algorithm works cycle-free.*

This fact explains, why our efforts to construct cycling examples in $\mathbf{R}^3$ have failed, and why we have been successful in $\mathbf{R}^4$.

Conclusion:

We have seen that the modified shadow vertex algorithm loses its distinctness on degenerate problems. While on $D(p, \cdot, n-1)$ and $D(p, \cdot, n-2)$-problems it works cycle-free, for all other degeneracy classes cycles can arise. For our examinations, the probabilistic analysis of the shadow vertex algorithm, cycling has to be prevented. For this purpose we either must disturb the degenerate problems such that non-degenerate ones result from that, or we have to restrict our considerations only on $D(p, \cdot, n-1)$ and $D(p, \cdot, n-2)$-problems. The detailed discussion of appropriate perturbation methods would be too lengthy and should not be done within this paper.

References:

[1] Borgwardt, K. H.: The Simplex Method, A Probabilistic Analysis, Springer-Verlag, Heidelberg, 1987

[2] Gass, S., Saaty, Th.: The Computational Algorithm for the Parametric Objective Function, Naval Research Logistics Quarterly 2, 1955, 39–45

[3] Klee, V., Kleinschmidt, P.: The d-Step Conjecture and its Relatives, University of Washington and University of Bochum, 1985

[4] Klee, V., Kleinschmidt, P.: Geometry of the Gass-Saaty Parametric Cost LP Algorithm, Discrete & Computational Geometry, New York, 1990

[5] Murty, K.: Linear Programming, Wiley, New York, 1983