Mathematical Programming 63 (1994) 235–252
North-Holland

A new pivoting algorithm for the linear complementarity problem allowing for an arbitrary starting point

Hans Kremers and Dolf Talman

Department of Econometrics, Tilburg University, Tilburg, Netherlands

Received 21 December 1990
Revised 16 March 1993

The linear complementarity problem is to find nonnegative vectors which are affinely related and complementary. In this paper we propose a new complementary pivoting algorithm for solving the linear complementarity problem as a more efficient alternative to the algorithms proposed by Lemke and by Talman and Van der Heyden. The algorithm can start at an arbitrary nonnegative vector and converges under the same conditions as Lemke's algorithm.

Key words: Linear complementarity problem, pivoting algorithm, stationary point problem.

1. Introduction

The *linear complementarity problem* (LCP(q, M)) is to find vectors $w \in \mathbb{R}^n$ and $z \in \mathbb{R}^n$ satisfying

$$w = Mz + q, \qquad w^{\mathrm{T}}z = 0, \qquad w \geqslant 0, \qquad z \geqslant 0, \tag{1.1}$$

where M is a given $n * n$-matrix and q a given n-vector. The linear complementarity problem is quite common in mathematical programming because the problem is frequently met in different areas of scientific research where optimization plays an important role. Often these optimization problems lead to Karush–Kuhn–Tucker conditions which take the form of a linear complementarity problem.

The popularity of the linear complementarity problem in mathematical programming has led to a variety of algorithms attempting to solve the problem. Among this variety of algorithms the Lemke complementary pivoting algorithm [5] is undoubtedly one of the most renowned algorithms. The Lemke algorithm is a path-following algorithm starting in $z = 0$ and generates a piecewise linear path either towards a solution to the linear comple-

Correspondence to: Dr. Hans Kremers, Department of Econometrics, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, Netherlands.
This research is part of the VF-program "Competition and Cooperation".

SSDI 0025-5610(93)E0040-L

mentarity problem or towards infinity. In Eaves [2] and Eaves [3] Lemke's algorithm has been generalized to computing stationary points on a polyhedron.

The major drawback of the Lemke complementary pivoting algorithm is that one is stuck to the fixed starting point $z=0$. This feature may cause inefficiencies when one has some idea concerning the possible location of a solution to the linear complementarity problem. Such information might for example be available when one tries to solve a nonlinear complementarity problem by a sequence of linear complementarity problems (see Mathiesen [6]). This inefficiency in processing the information makes it worthwhile to adapt Lemke's algorithm for an arbitrarily chosen starting point. Talman and Van der Heyden [8] present a whole class of algorithms generalizing the Lemke complementary pivoting algorithm to an arbitrarily chosen starting point. All the algorithms in this class however use a pivot system of at least $n+1$ equations in order to guarantee possible convergence of the algorithm. Moreover none of these algorithms seem to be very natural in solving the linear complementarity problem. To get rid of these inefficiencies we propose a new pivoting algorithm to solve the linear complementarity problem allowing for an arbitrarily chosen starting point. This algorithm has a natural interpretation as a path-following algorithm and it does not need more than n equations in the pivot system.

The algorithm leaves the starting point in one out of $n+1$ possible directions. There are n rays that connect the starting point with each of the n axes of $\mathbb{R}^n_+$ and one ray that connects the starting point with the origin. This allows the algorithm to leave the starting point z^0 in such a way that, with $w^0=Mz^0+q$, it will raise z_i from z_i^0 when w_i^0 is negative and smaller than all other components of w^0, while the algorithm will lower each z_i from z_i^0 proportionally towards zero when w_i^0 is positive or not smaller than all other components of w^0. In particular this latter feature endows the algorithm with a very natural interpretation. For example, the algorithm will stop with a solution to the linear complementarity problem if it reaches the origin. This is contrary to the algorithm in the Talman and Van der Heyden class of algorithms having also $n+1$ rays to leave the starting point. In that algorithm there are n rays that leave the starting point parallel to each of the axes and there is one ray connecting the starting point with the origin.

In our algorithm the intersection of the rays with each of the axes can arbitrarily be chosen. In Section 4 of this paper we suggest a particular choice of these intersections such that it might be possible to see in advance whether the algorithm might not solve the problem.

An important feature of the algorithm is that it performs in the same way as the Lemke complementary pivoting algorithm when generating points sufficiently far away from the origin. In particular, the algorithm coincides with Lemke's algorithm when the origin is the starting point. Hence, the algorithm converges under the same conditions as the Lemke complementary pivoting algorithm. Any convergence theorem for the Lemke complementary pivoting algorithm can therefore be applied for the algorithm.

The paper is divided as follows. First we describe the algorithm. The algorithm follows a path of points in $\mathbb{R}^n_+$. Each point on this path can be interpreted as a so-called stationary point of an affine function on a parametrized set. This interpretation is elaborated in Section 2. The steps of the algorithm are enumerated in Section 3 while Section 4 is dedicated to convergency issues. Section 5 describes some examples to clarify the algorithm.

2. The algorithm

The algorithm follows a piecewise linear path of points in $\mathbb{R}^n_+$ starting in some arbitrarily chosen point $z^0 \in \mathbb{R}^n_+$. We allow z^0 to lie on the boundary of $\mathbb{R}^n_+$. In case $z^0 = 0$ the algorithm coincides with the Lemke complementary pivoting algorithm.

To motivate the algorithm, let (w^*, z^*) be a solution to LCP(q, M) defined in (1.1). The conditions defining a solution (w^*, z^*) to LCP(q, M) can be interpreted as

$$\text{if } z^* = 0 \text{ then } \min_h w_h^* \geqslant 0,$$

$$\text{if } z^* \neq 0 \text{ then } \min_h w_h^* = 0,$$

and for all $j \in \{1, \ldots, n\}$,

$$\text{if } z_j^* > 0 \text{ then } w_j^* = \min_h w_h^*,$$

$$\text{if } z_j^* = 0 \text{ then } w_j^* \geqslant \min_h w_h^*.$$

Let $w^0 = Mz^0 + q$. Then this interpretation of the conditions to hold at a solution (w^*, z^*) of LCP(q, M) makes it very natural to leave the starting point z^0 in the following way. If $\min_h w_h^0 \geqslant 0$, then the algorithm decreases z proportionally from z^0. Otherwise, the algorithm decreases z_j from z_j^0 for all $j \in \{1, \ldots, n\}$ such that $w_j^0 > \min_h w_h^0$ and increases z_j from z_j^0 for all $j \in \{1, \ldots, n\}$ such that $w_j^0 = \min_h w_h^0 < 0$. More precisely, when $\min_h w_h^0 \geqslant 0$ then it is most natural to leave z^0 towards the origin and when $w_j^0 = \min_h w_h^0 < 0$ for some $j \in \{1, \ldots, n\}$ then it is most natural to leave z^0 towards a point $ae(j)$ for some $a > z_j^0$ on the jth axis, $e(j)$ denoting the jth unit vector in $\mathbb{R}^n$. This interpretation gives $n+1$ rays to leave the starting point, namely $r(j) = ae(j) - z^0$ for $j \in \{1, \ldots, n\}$ and $r(n+1) = -z^0$.

Let S be a set in $\mathbb{R}^n$ and let $f: S \to \mathbb{R}^n$ be a function. A point x^* in S is called a *stationary point* of f on S if $x^{*\mathrm{T}} f(x^*) \geqslant x^{\mathrm{T}} f(x^*)$ for all x in S. The *stationary point problem* (SPP) on S with respect to f is to find a stationary point of f on S. Notice that the point x^* being a stationary point of f on S is equivalent to x^* solving the optimization problem given by $\max\{x^{\mathrm{T}} f(x^*) \mid x \in S\}$. Furthermore, in case S is a polyhedron this optimization problem reduces to a linear programming problem as it maximizes the linear function $x^{\mathrm{T}} f(x^*)$ subject to the linear constraints defining the polyhedron S.

LCP(q, M) is equivalent to the SPP on $\mathbb{R}^n_+$ with respect to the affine function g defined by $g(z) = -Mz - q$ on $\mathbb{R}^n_+$, as can easily be shown. Combining this interpretation of an LCP with the idea above yielding the $n+1$ rays to leave the starting point z^0, we propose a path following algorithm which is such that each point z on the path is a stationary point of the function g on the set $\mathscr{H}(t) \cap \mathbb{R}^n_+$ for some $t \geqslant 0$, where

$$\mathscr{H}(t) = \left\{ z^0 + \sum_{j=1}^{n+1} \lambda_j r(j) \,\middle|\, \lambda_j \geqslant 0 \text{ for } j \in \{1, \ldots, n+1\}, \text{ and } \sum_{j=1}^{n+1} \lambda_j \leqslant t \right\}.$$

The number a should be such that $a > e^{\mathrm{T}}z^0$, assuring that $z^0 \in \mathscr{H}(t)$ for any $t > 0$.[1] The vector e denotes the n-vector with all components equal to one.

The number t can be considered as a homotopy parameter running from zero to infinity. For $t=0$ the set $\mathscr{H}(0) \cap \mathbb{R}^n_+$ only consists of the starting point z^0. Hence z^0 is a stationary point of g on $\mathscr{H}(0) \cap \mathbb{R}^n_+$. For $t=1$ the set $\mathscr{H}(1) \cap \mathbb{R}^n_+$ is the convex hull of the origin and the points $ae(j), j \in \{1, \ldots, n\}$, on the axes of $\mathbb{R}^n_+$. If the algorithm generates a stationary point z of g on $\mathscr{H}(1) \cap \mathrm{bd}\ \mathbb{R}^n_+$ such that $e^{\mathrm{T}}z < a$ then z is also a stationary point of g on $\mathbb{R}^n_+$ and consequently a solution to the linear complementarity problem. For $t \geqslant 1$ the set $\mathscr{H}(t) \cap \mathbb{R}^n_+$ is equal to the convex hull of the origin and the points $[(1-t)e^{\mathrm{T}}z^0 + ta]e(j)$, $j \in \{1, \ldots, n\}$, on the axes of $\mathbb{R}^n_+$.[2] Clearly, for $t \geqslant 0$, the set $\mathscr{H}(t)$ is equal to

$$\mathscr{H}(t) = \{z \in \mathbb{R}^n \mid z \geqslant (1-t)z^0,\ e^{\mathrm{T}}z \leqslant (1-t)e^{\mathrm{T}}z^0 + ta\}$$

and hence, $\mathscr{H}(t) \cap \mathbb{R}^n_+$ is equal to the set

$$\{z \in \mathbb{R}^n \mid z \geqslant \max\{1-t, 0\}z^0,\ e^{\mathrm{T}}z \leqslant (1-t)e^{\mathrm{T}}z^0 + ta\}.$$

For an illustration of the set $\mathscr{H}(t) \cap \mathbb{R}^n_+$ for different values of t we refer to Figure 1. Now the algorithm follows a path of stationary points of g on $\mathscr{H}(t) \cap \mathbb{R}^n_+$ for varying parameter $t \geqslant 0$ starting for $t=0$ in the arbitrarily chosen point $z^0 \in \mathbb{R}^n_+$. Barring degeneracy, this path terminates either in a ray or at a solution.

If $\bar{z}$ in $\mathbb{R}^n_+$ is a stationary point of g on $\mathscr{H}(\bar{t}) \cap \mathbb{R}^n_+$ for some $\bar{t} \geqslant 0$ then $\bar{z}$ maximizes $z^{\mathrm{T}}g(\bar{z})$ over $\mathscr{H}(\bar{t}) \cap \mathbb{R}^n_+$. By definition of $\mathscr{H}(t)$ this implies that $\bar{z}$ solves the maximization problem, denoted as the primal, given by

$$\begin{array}{ll} \max & z^{\mathrm{T}}g(\bar{z}) \\ \text{s.t.} & z \geqslant \max\{1-\bar{t}, 0\}z^0, \\ & e^{\mathrm{T}}z \leqslant (1-\bar{t})e^{\mathrm{T}}z^0 + \bar{t}a. \end{array}$$

This maximization problem is a linear programming problem. According to the Duality Theorem of Linear Programming this maximization problem is equivalent to the minimization problem, denoted as the dual, given by

$$\begin{array}{ll} \min & \theta((1-\bar{t})e^{\mathrm{T}}z^0 + \bar{t}a) - \max\{1-\bar{t}, 0\}\mu^{\mathrm{T}}z^0 \\ \text{s.t.} & g(\bar{z}) = -\mu + \theta e, \\ & \theta \geqslant 0, \quad \mu \geqslant 0, \end{array}$$

[1] In Section 4 of this paper we will make use of this freedom by letting the choice of a depend on the data of the matrix M and the vector q.

[2] Notice that a point on the kth axis of $\mathbb{R}^n_+$ is given by $\gamma e(k)$ for some $\gamma \geqslant 0$. Hence the point of bd $\mathscr{H}(t)$, $t \geqslant 1$, on the kth axis of $\mathbb{R}^n_+$ follows from finding the values of γ, $\lambda_1, \ldots, \lambda_n$ such that $\sum_{j=1}^n \lambda_j = t$ and $\gamma e(k) = z^0 + \sum_{j=1}^{n+1} \lambda_j r(j)$. Adding up the latter equation over all components gives $\gamma = (1-t)e^{\mathrm{T}}z^0 + ta$.

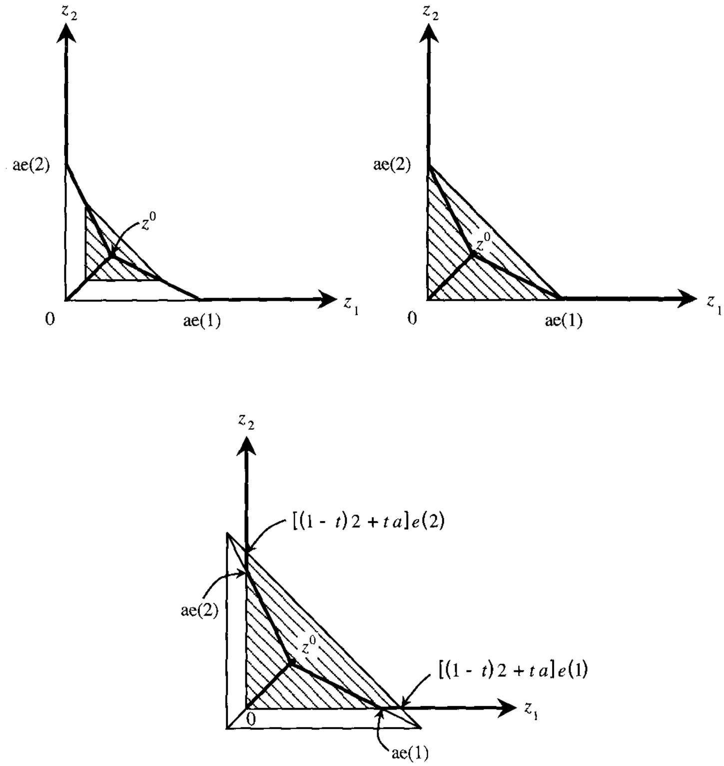

Fig. 1. The subset $\mathscr{H}(t)\cap\mathbb{R}^2_+$ for $t=\frac{1}{2}$, 1, resp. $\frac{3}{2}$, given $z^0=(1,1)^{\mathrm{T}}$ and $a=3$.

where θ is the dual variable to the constraint $e^{\mathrm{T}}z\leqslant(1-\bar{t})e^{\mathrm{T}}z^0+\bar{t}a$ and μ the n-vector with μ_j the dual variable to the constraint $z_j\geqslant\max\{1-\bar{t},0\}z_j^0$ for $j\in\{1,\dots,n\}$.

Given $g(\bar{z})$ the dual has a unique solution $\bar{\theta}=\max\{0,\max_h g_h(\bar{z})\}$, $\bar{\mu}_j=\bar{\theta}-g_j(\bar{z})$ for $j\in\{1,\dots,n\}$, Let $\bar{\mathscr{T}}$ be a subset of $\{1,\dots,n+1\}$ such that $\bar{\mu}_j=0$ for all $j\in\bar{\mathscr{T}}\setminus\{n+1\}$, $\bar{\mu}_j>0$ for all $j\notin\bar{\mathscr{T}}\cup\{n+1\}$, $\bar{\theta}=0$ if $n+1\in\bar{\mathscr{T}}$, and $\bar{\theta}>0$ if $n+1\notin\bar{\mathscr{T}}$. Then $g_j(\bar{z})=\bar{\theta}$ for all $j\in\bar{\mathscr{T}}\setminus\{n+1\}$, $g_j(\bar{z})<\bar{\theta}$ for all $j\notin\bar{\mathscr{T}}\cup\{n+1\}$, $\max_h g_h(\bar{z})=0$ if $n+1\in\bar{\mathscr{T}}$, and $\max_h g_h(\bar{z})>0$ if $n+1\notin\bar{\mathscr{T}}$.

Definition 2.1. For a subset $\mathscr{T}$ of $\{1,\dots,n+1\}$ a point $z\in\mathbb{R}^n_+$ is $\mathscr{T}$-complete if $j\in\mathscr{T}$ implies $g_j(z)=\theta$ and $n+1\in\mathscr{T}$ implies $\theta=0$, where $\theta=\max\{\max_h g_h(z),0\}$.

Furthermore, the solution $(\bar{\mu},\bar{\theta})$ to the dual is complementary to the constraints in the primal in $\bar{z}$. Hence, if $j\in\bar{\mathscr{T}}\setminus\{n+1\}$ then $\bar{\mu}_j=0$, implying $\bar{z}_j\geqslant\max\{1-\bar{t},0\}z_j^0$. If $j\notin\bar{\mathscr{T}}\cup\{n+1\}$ then $\bar{\mu}_j>0$, implying $\bar{z}_j=\max\{1-\bar{t},0\}z_j^0$. If $n+1\in\bar{\mathscr{T}}$ then $\bar{\theta}=0$, implying $e^{\mathrm{T}}\bar{z}\leqslant(1-\bar{t})e^{\mathrm{T}}z^0+\bar{t}a$. If $n+1\notin\bar{\mathscr{T}}$ then $\bar{\theta}>0$, implying $e^{\mathrm{T}}\bar{z}=(1-\bar{t})e^{\mathrm{T}}z^0+\bar{t}a$. Therefore, the subset $\bar{\mathscr{T}}$ of $\{1,\dots,n+1\}$ defines a subset of $\mathbb{R}^n_+$. For any subset $\mathscr{T}$ of $\{1,\dots,n+1\}$ we can define subsets $A(\mathscr{T})$ and $A^0(\mathscr{T})$ of $\mathbb{R}^n_+$ as follows.

Definition 2.2. For $\mathscr{T} \subset \{1, \ldots, n+1\}$,

$$A(\mathscr{T}) = \emptyset \quad \text{if } n+1 \in \mathscr{T} \quad \text{and} \quad z_h^0 = 0 \quad \text{for all } h \notin \mathscr{T}$$

and otherwise

$$A(\mathscr{T}) = (\{z^0\} + \text{cone}(\{r(j) \mid j \in \mathscr{T}\})) \cap \mathbb{R}_+^n.$$

Definition 2.3. For $\mathscr{T} \subset \{1, \ldots, n+1\}$,

$$A^0(\mathscr{T}) = \emptyset \quad \text{if } n+1 \in \mathscr{T} \quad \text{or} \quad z_h^0 = 0 \quad \text{for all } h \notin \mathscr{T}$$

and otherwise

$$A^0(\mathscr{T}) = \left\{ \sum_{j \in \mathscr{T}} \lambda_j ae(j) \,\middle|\, \lambda_j \geqslant 0 \text{ for } j \in \mathscr{T} \text{ and } \sum_{j \in \mathscr{T}} \lambda_j \geqslant 1 \right\}.$$

Figure 2 gives a subdivision of $\mathbb{R}_+^n$ into subsets $A(\mathscr{T})$ and $A^0(\mathscr{T})$ for subsets $\mathscr{T}$ of $\{1, \ldots, n+1\}$ when $n=2$. Theorem 2.1 proves that z being a stationary point of g on $\mathscr{H}(t) \cap \mathbb{R}_+^n$ for some $t \geqslant 0$ is equivalent to z being a $\mathscr{T}$-complete point in $A(\mathscr{T})$ or $A^0(\mathscr{T})$ for some subset $\mathscr{T}$ of $\{1, \ldots, n+1\}$.

Theorem 2.1. *The point $z \in \mathbb{R}_+^n$ is a $\mathscr{T}$-complete point in $A(\mathscr{T})$ or $A^0(\mathscr{T})$ for some $\mathscr{T} \subset \{1, \ldots, n+1\}$ if and only if z is a stationary point of g on $\mathscr{H}(t) \cap \mathbb{R}_+^n$ for some $t \geqslant 0$.*

Proof. Let $\bar{z}$ be a $\mathscr{T}$-complete point in $A(\mathscr{T})$ for some $\mathscr{T} \subset \{1, \ldots, n+1\}$. Let $\bar{t} = e^{\mathrm{T}}\bar{z}$. Then $\bar{z}$ solves the primal for $t = \bar{t}$. Hence for all $z \in \mathscr{H}(\bar{t})$ it holds that $z^{\mathrm{T}}g(\bar{z}) \leqslant \bar{z}^{\mathrm{T}}g(\bar{z})$. A similar reasoning holds for $\bar{z} \in A^0(\mathscr{T})$. The converse follows from the discussion above. □

To generate stationary points of g on $\mathscr{H}(t) \cap \mathbb{R}_+^n$ for varying $t \geqslant 0$ the algorithm should therefore generate a set of $\mathscr{T}$-complete points in $A(\mathscr{T})$ or a set of $\mathscr{T}$-complete points in

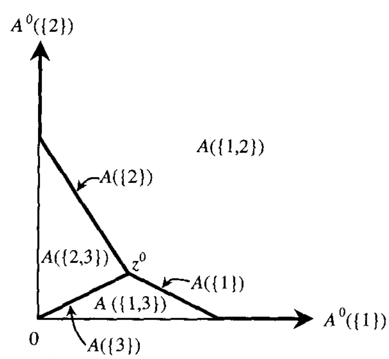

Fig. 2. Subdivision of $\mathbb{R}_+^2$ into subsets $A(\mathscr{T})$ and $A^0(\mathscr{T})$ for $\mathscr{T} \subset \{1, 2, 3\}$.

$A^0(\mathscr{T})$ for subsets $\mathscr{T}$ of $\{1, \ldots, n+1\}$. Among these sets of $\mathscr{T}$-complete points there exists a collection of adjacent sets of $\mathscr{T}$-complete points in $A(\mathscr{T})$ or $A^0(\mathscr{T})$ for varying subsets $\mathscr{T}$ of $\{1, \ldots, n+1\}$. This collection of adjacent sets is a path of $\mathscr{T}$-complete points in $A(\mathscr{T})$ or $A^0(\mathscr{T})$ for varying subsets $\mathscr{T}$ of $\{1, \ldots, n+1\}$ starting in z^0 and either ending up with a solution to LCP(q, M) or ending up in a ray towards infinity. The next section describes how to generate this path through subsequent subsets $A(\mathscr{T})$ and $A^0(\mathscr{T})$ for varying subsets $\mathscr{T}$ of $\{1, \ldots, n+1\}$ by complementary pivoting.

3. The steps of the algorithm

Theorem 2.1 implies that the algorithm generates a path of $\mathscr{T}$-complete points in $A(\mathscr{T})$ or $A^0(\mathscr{T})$ for varying subsets $\mathscr{T}$ of $\{1, \ldots, n+1\}$. This property leads to a pivot system in the following way. The $\mathscr{T}$-completeness condition at a point z is equivalent to the existence of $\mu_j \geqslant 0$ $(j \notin \mathscr{T} \cup \{n+1\})$, $\theta \geqslant 0$ if $n+1 \notin \mathscr{T}$, $\theta = 0$ if $n+1 \in \mathscr{T}$ such that

$$-Mz - q = \theta e - \sum_{j \notin \mathscr{T} \cup \{n+1\}} \mu_j e(j).$$

Combined with $z \in A(\mathscr{T})$ or $z \in A^0(\mathscr{T})$ the appropriate pivot system for $\mathscr{T}$-completeness at a point z in $A(\mathscr{T})$ or $A^0(\mathscr{T})$ is given in one of the next two lemmas where, for $j = 1, \ldots, n$, $M_{.j}$ denotes the jth column of the matrix M.

Lemma 3.1. *A point $z \in A(\mathscr{T})$ is $\mathscr{T}$-complete for some feasible $\mathscr{T} \subset \{1, \ldots, n+1\}$ if and only if the system of equations*

$$\sum_{j \in \mathscr{T}} \lambda_j M r(j) - \sum_{j \notin \mathscr{T} \cup \{n+1\}} \mu_j e(j) + \theta e = -q - Mz^0 \tag{3.1}$$

has a solution $\lambda_j \geqslant 0$ $(j \in \mathscr{T})$, $\mu_j \geqslant 0$ $(j \notin \mathscr{T} \cup \{n+1\})$, $\theta \geqslant 0$ if $n+1 \notin \mathscr{T}$, $\theta = 0$ if $n+1 \in \mathscr{T}$, such that $z = z^0 + \sum_{j \in \mathscr{T}} \lambda_j r(j)$. □

Lemma 3.2. *A point $z \in A(\mathscr{T})$ for some $\mathscr{T} \subseteq \{1, \ldots, n\}$ with $z_i^0 = 0$ for all $i \notin \mathscr{T}$ or a point $z \in A^0(\mathscr{T})$ for some $\mathscr{T} \subseteq \{1, \ldots, n\}$ is $\mathscr{T}$-complete if and only if the system of equations*

$$\sum_{j \in \mathscr{T}} \lambda_j a M_{.j} - \sum_{j \notin \mathscr{T} \cup \{n+1\}} \mu_j e(j) + \theta e = -q \tag{3.2}$$

has a solution $\lambda_j \geqslant 0$ $(j \in \mathscr{T})$, $\mu_j \geqslant 0$ $(j \notin \mathscr{T} \cup \{n+1\})$, $\theta \geqslant 0$, such that $z = \sum_{j \in \mathscr{T}} \lambda_j a e(j)$. □

Notice that the pivot systems in (3.1) and (3.2) both contain n equations in $n+1$ variables leaving us with one degree of freedom. If nonempty, the solution set of each system forms a line segment, assuming nondegeneracy. This line segment corresponds to a linear piece of $\mathscr{T}$-complete points in $A(\mathscr{T})$ or in $A^0(\mathscr{T})$ with either one or two end points. As we will show below each end point of a line segment of solutions to a system of equations for some

$\mathcal{T} \subset \{1, \ldots, n+1\}$ either corresponds to the starting point z^0 or to a solution to the linear complementarity problem or is an end point of a line segment of solutions to exactly one other system of equations possibly for a different set $\mathcal{T}$. The point z^0 will correspond to an end point of only one line segment of solutions. These properties make the path of points generated by the algorithm from z^0 to be a piecewise linear path through subsequent subsets $A(\mathcal{T})$ and $A^0(\mathcal{T})$ for varying subsets $\mathcal{T}$ of $\{1, \ldots, n+1\}$. Each linear piece can be followed by making a linear programming pivot step in the appropriate pivot system with the variable being zero (or making a binding constraint) at an end point.

A linear piece of $\mathcal{T}$-complete points in $A(\mathcal{T})$ for some subset $\mathcal{T} \subseteq \{1, \ldots, n\}$ for which $z_i^0 = 0$ for all $i \notin \mathcal{T}$ can be generated by making a pivot step in system (3.1) or in system (3.2). Which one of these systems will be appropriate depends on the system in which the previous pivot step was made. This feature causes the algorithm to generate the path through different subsets of $\mathbb{R}^n_+$ in a more efficient way. Changing from one pivot system to the other one at an end point of a line segment requires a redefinition of the variables $\lambda_j, j \in \mathcal{T}$. The setup in Lemma 3.1 and Lemma 3.2 allows us to make as few of these changes of variables as possible.

Suppose the algorithm is following a linear piece of $\mathcal{T}$-complete points in $A(\mathcal{T})$ or in $A^0(\mathcal{T})$ for some $\mathcal{T} \subset \{1, \ldots, n+1\}$, i.e., a pivot step is made in one of the systems of equations (3.1) or (3.2) with the variable being zero at an end point of the line segment of solutions. When the linear piece has another end point, say $\bar{z}$, then, assuming nondegeneracy, exactly one of the following cases will occur for the solution $(\bar{\lambda}, \bar{\mu}, \bar{\theta})$ at this end point:

Case 1. λ_p is equal to zero for some $p \in \mathcal{T}$, while $\mathcal{T} \setminus \{p\} \neq \emptyset$. Then $\bar{z}$ is an end point lying in $A(\mathcal{T} \setminus \{p\})$ or in $A^0(\mathcal{T} \setminus \{p\})$ depending on whether $\bar{z} \in A(\mathcal{T})$ or $\bar{z} \in A^0(\mathcal{T})$ respectively. The algorithm proceeds in $A(\mathcal{T} \setminus \{p\})$ or $A^0(\mathcal{T} \setminus \{p\})$ by pivoting the column $-e(p)$ into the appropriate system of equations thereby raising μ_p from zero if $p \neq n+1$, and pivoting the column e into the appropriate system of equations thereby raising θ from zero if $p = n+1$, in order to maintain $\mathcal{T} \setminus \{p\}$-completeness.

Case 2. In system (3.1) $z_p^0 > 0$ and $\bar{\lambda}_p$ is equal to

$$\left(\sum_{j \in \mathcal{T} \setminus \{p\}} \bar{\lambda}_j - 1\right)\left(\frac{z_p^0}{a - z_p^0}\right)$$

for some $p \in \mathcal{T}$. Then $\bar{z}$ lies in the boundary of $A(\mathcal{T})$. More precisely, $\bar{z}$ lies in $A^0(\mathcal{T} \setminus \{p\})$. Let

$$\hat{\lambda}_j = \bar{\lambda}_j + \left(1 - \sum_{h \in \mathcal{T}} \bar{\lambda}_h\right)\left(\frac{z_j^0}{a}\right)$$

for $j \in \mathcal{T} \setminus \{p\}$. Then $\hat{\lambda}_j \geq 0$ $(j \in \mathcal{T} \setminus \{p\})$, $\bar{\mu}_h \geq 0$ $(h \notin \mathcal{T} \cup \{n+1\})$, $\bar{\mu}_p = 0$, and $\bar{\theta} \geq 0$ is a solution to system (3.2) and $\bar{z}$ is an end point of a linear piece of $\mathcal{T} \setminus \{p\}$-complete points in $A^0(\mathcal{T} \setminus \{p\})$. The algorithm proceeds in $A^0(\mathcal{T} \setminus \{p\})$ by changing system (3.1) into system (3.2) and pivoting the column $-e(p)$ into the new system (3.2) thereby raising μ_p from zero in order to maintain $\mathscr{T}\setminus\{p\}$-completeness.

Case 3. $\sum_{j\in\mathscr{T}}\lambda_j$ is equal to 1 either in system (3.1) while $n+1\in\mathscr{T}$ or $z_h^0>0$ for some $h\notin\mathscr{T}$, or in system (3.2) while $z_h^0>0$ for some $h\notin\mathscr{T}$. Suppose $n+1\in\mathscr{T}$, then $\bar{w}_j=0$ and $\bar{z}_j=\bar{\lambda}_j a\geqslant 0$ for $j\in\mathscr{T}$ while $\bar{w}_j=\bar{\mu}_j\geqslant 0$ and $\bar{z}_j=(1-\sum_{j\in\mathscr{T}}\bar{\lambda}_j)z_j^0=0$ for $j\notin\mathscr{T}$, leaving us with a solution to the linear complementarity problem in $\bar{z}$. Suppose $n+1\notin\mathscr{T}$, then it holds that $z_h^0>0$ for some $h\notin\mathscr{T}$. Hence, $\bar{z}$ is an end point of a linear piece of $\mathscr{T}$-complete points in $A(\mathscr{T})$ as well as in $A^0(\mathscr{T})$. So, if $\bar{z}$ were the end point of a linear piece of $\mathscr{T}$-complete points in $A(\mathscr{T})$, then the algorithm proceeds by generating a linear piece of $\mathscr{T}$-complete points in $A^0(\mathscr{T})$. This linear piece of $\mathscr{T}$-complete points in $A^0(\mathscr{T})$ is generated by changing system (3.1) into system (3.2) and pivoting the column $aM_{\cdot k}$ or $-e(k)$ into the new system (3.2), depending on whether $Mr(k)$ or $-e(k)$ was the last pivot column in (3.1). Notice that $\sum_{j\in\mathscr{T}}\lambda_j$ is then raised from 1. Conversely, if $\bar{z}$ were the end point of a linear piece of $\mathscr{T}$-complete points in $A^0(\mathscr{T})$, then the algorithm proceeds by generating a linear piece of $\mathscr{T}$-complete points in $A(\mathscr{T})$. This linear piece of $\mathscr{T}$-complete points in $A(\mathscr{T})$ is generated by changing system (3.2) into system (3.1) and pivoting the column $Mr(k)$ or $-e(k)$ into the new system (3.1), depending on whether $aM_{\cdot k}$ or $-e(k)$ was the last pivot column in the system (3.2). Hence $\sum_{j\in\mathscr{T}}\lambda_j$ is lowered from 1.

Case 4. In system (3.2) it holds that $z_p^0>0$ and $\bar{\lambda}_p$ is equal to

$$z_p^0\left(\frac{1-\sum_{i\in\mathscr{T}\setminus\{p\}}\bar{\lambda}_i}{a-\sum_{j\in\mathscr{T}}z_j^0}\right)$$

for some $p\in\mathscr{T}$. Then $\bar{z}$ lies in the boundary of $A(\mathscr{T})$. More precisely, $\bar{z}$ lies in $A(\mathscr{T}\setminus\{p\})$. Let

$$\hat{\lambda}_h=\bar{\lambda}_h-z_h^0\left(\frac{1-\sum_{j\in\mathscr{T}\setminus\{p\}}\bar{\lambda}_j}{a-\sum_{j\in\mathscr{T}}z_j^0}\right)$$

for $h\in\mathscr{T}\setminus\{p\}$. Then $\hat{\lambda}_h\geqslant 0$ $(h\in\mathscr{T}\setminus\{p\})$, $\bar{\mu}_h\geqslant 0$ $(h\notin\mathscr{T}\cup\{n+1\})$, $\bar{\mu}_p=0$, and $\bar{\theta}\geqslant 0$ is a solution to (3.1), and $\bar{z}$ is an end point of a linear piece of $\mathscr{T}\setminus\{p\}$-complete points in $A(\mathscr{T}\setminus\{p\})$. The algorithm proceeds by changing system (3.2) into system (3.1) and pivoting the column $-e(p)$ into system (3.1) thereby raising μ_p from zero in order to maintain $\mathscr{T}\setminus\{p\}$-completeness.

Case 5. $\bar{\mu}_k$ is zero for some $k\notin\mathscr{T}\cup\{n+1\}$. Suppose $\bar{z}\in A(\mathscr{T})$. If $z_h^0=0$ for all $h\notin\mathscr{T}\cup\{k\}$ while $n+1\in\mathscr{T}$, or if $\mathscr{T}\cup\{k\}=\{1,\dots,n+1\}$, then $\bar{z}$ is a solution to the linear complementarity problem. Otherwise, $\bar{z}$ is an end point of a linear piece of $\mathscr{T}\cup\{k\}$-complete points in $A(\mathscr{T}\cup\{k\})$. The algorithm proceeds by pivoting the column $Mr(k)$ into the system (3.1) or $aM_{\cdot k}$ into the system (3.2) thereby raising λ_k from zero in order to maintain $\mathscr{T}\cup\{k\}$-completeness.

Suppose $\bar{z}\in A^0(\mathscr{T})$. If $z_h^0=0$ for all $h\notin\mathscr{T}\cup\{k\}$, then $\bar{z}$ is an end point of a linear piece of $\mathscr{T}\cup\{k\}$-complete points in $A(\mathscr{T}\cup\{k\})$. Otherwise, $\bar{z}$ is an end point of a linear piece of $\mathscr{T}\cup\{k\}$-complete points in $A^0(\mathscr{T}\cup\{k\})$. The algorithm proceeds in both cases by pivoting the column $aM_{\cdot k}$ into the system (3.2) thereby raising λ_k from zero in order to maintain $\mathcal{T}\cup\{k\}$-completeness.

Case 6. $\bar{\theta}$ is zero. Then $\bar{z}$ is a solution to the linear complementarity problem if $\bar{z}\in A^0(\mathcal{T})$ or if $\bar{z}\in A(\mathcal{T})$ and $z_h^0=0$ for all $h\notin\mathcal{T}$. Otherwise, $\bar{z}$ is an end point of a linear piece of $\mathcal{T}\cup\{n+1\}$-complete points in $A(\mathcal{T}\cup\{n+1\})$. The algorithm proceeds by pivoting the column $-Mz^0$ into (3.1) thereby raising λ_{n+1} from zero in order to maintain $\mathcal{T}\cup\{n+1\}$-completeness.

The cases 1 to 6 describe the performance of the algorithm at the end points of all possible line segments generated by the algorithm except at z^0 where the algorithm is initiated. To show that z^0 is an end point of a (unique) linear piece of $\mathcal{T}$-complete points in $A(\mathcal{T})$ for some subset $\mathcal{T}$ of $\{1, \ldots, n+1\}$ let us denote Mz^0+q by w^0. If $\min_h w_h^0<0$, let k be such that $w_k^0=\min_h w_h^0$. Then the starting point z^0 is $\mathcal{T}^0$-complete with $\mathcal{T}^0=\{k\}$ and the system of equations

$$-\sum_{j\neq k,\ n+1}\mu_j e(j)+\theta e=-q-Mz^0 \tag{3.3}$$

has a unique solution $\mu_j^0=w_j^0-w_k^0>0$ $(j\neq k,\ n+1)$ and $\theta^0=-w_k^0>0$. So, assuming nondegeneracy, z^0 is an end point of a linear piece of $\{k\}$-complete points in $A(\{k\})$. In order to follow this linear piece the algorithm starts by pivoting the column $Mr(k)$ into (3.3) thereby raising λ_k from zero.

If $\min_h w_h^0\geqslant 0$, then the starting point z^0 solves the linear complementarity problem if $z^0=0$. Otherwise z^0 is $\mathcal{T}^0$-complete with $\mathcal{T}^0=\{n+1\}$ and the system of equations

$$-\sum_{j=1}^{n}\mu_j e(j)=-q-Mz^0 \tag{3.4}$$

has a unique solution $\mu_j^0=w_j^0>0$ $(j\in\{1, \ldots, n\})$. Assuming nondegeneracy, z^0 is the end point of a linear piece of $\{n+1\}$-complete points in $A(\{n+1\})$. In order to follow this linear piece the algorithm starts by pivoting the column $-Mz^0$ into (3.4) thereby raising λ_{n+1} from zero.

4. Convergence issues

The cases described in the previous section show that each end point of a line segment of solutions to (3.1) or (3.2) either corresponds to the starting point z^0 or to a solution of LCP(q, M) or is an end point of a line segment of solutions to exactly one other system of equations. The point z^0 corresponds to an end point of exactly one line segment of solutions to (3.1). To find a solution to problem (1.1), the algorithm starts in z^0 and generates a sequence of adjacent line segments of solutions to (3.1) or (3.2) for varying subsets $\mathcal{T}$ of $\{1, \ldots, n+1\}$. Each line segment of solutions is generated by making a pivot step in either (3.1) or (3.2) with one of the variables being zero at an end point until one of the cases described in the previous section occurs. Such a line segment of solutions to (3.1) or (3.2) corresponds to a linear piece of $\mathscr{T}$-complete points in either $A(\mathscr{T})$ or $A^0(\mathscr{T})$.

These properties make the path of points generated by the algorithm a piecewise linear path through subsequent subsets $A(\mathscr{T})$ and $A^0(\mathscr{T})$ of $\mathbb{R}^n_+$ for varying subsets $\mathscr{T}$ of $\{1, \ldots, n+1\}$. This path either ends in a solution to the linear complementarity problem or in a ray of solutions to (3.2) since it cannot cycle as can be shown by arguments similar to Lemke [5]. The end points giving rise to a solution to the linear complementarity problem have already been described during the enumeration of the cases in Section 3. Lemma 4.1 summarizes all the cases in which the algorithm ends up in a solution.

Lemma 4.1. *Let z be an end point of a linear piece of $\mathscr{T}$-complete points on the path generated by the algorithm in $A(\mathscr{T})$ or in $A^0(\mathscr{T})$ for some subset $\mathscr{T}$ of $\{1, \ldots, n+1\}$. Then z is a solution to the linear complementarity problem if one of the following cases holds*:

(i) $z \in A(\mathscr{T})$, $n+1 \in \mathscr{T}$, $\mu_k = 0$ *for some* $k \notin \mathscr{T}$, *and* $z^0_h = 0$ *for all* $h \notin \mathscr{T} \cup \{k\}$ *or* $\mathscr{T} \cup \{k\} = \{1, \ldots, n+1\}$;

(ii) $z \in A^0(\mathscr{T})$ *and* $\theta = 0$;

(iii) $z \in A(\mathscr{T})$, $z^0_h = 0$ *for all* $h \notin \mathscr{T}$, *and* $\theta = 0$;

(iv) $z \in A(\mathscr{T})$, $n+1 \in \mathscr{T}$, *and* $\sum_{j \in \mathscr{T}} \lambda_j = 1$,

where the variables $\lambda_j \geqslant 0$ $(j \in \mathscr{T})$, $\mu_j \geqslant 0$ $(j \notin \mathscr{T} \cup \{n+1\})$, $\theta \geqslant 0$ *are the solution to the appropriate pivot system* (3.1) *or* (3.2) *at* z. □

The possibility of divergence urges us to impose a convergence condition on the problem. Notice that divergence can only occur when the algorithm is generating a path of points in $A^0(\mathscr{T})$ or in $A(\mathscr{T})$ with $\mathscr{T}$ such that $z^0_i = 0$ for all $i \notin \mathscr{T}$ and $n+1 \notin \mathscr{T}$, i.e., when the system of equations in Lemma 3.2 is appropriate. Therefore we can restrict our attention to the possible occurrence of a ray of solutions to system (3.2) for some $\mathscr{T} \subseteq \{1, \ldots, n\}$. System (3.2) however is equivalent to the pivot system used in Lemke [5] to solve the linear complementarity problem. So, any convergence theorem on Lemke's algorithm can be used for our algorithm.

In Gowda and Pang [4] an existence theorem is given based on the Basic Theorem of Complementarity as stated in Eaves [1]. They relate their result on the existence of a solution to the stationary point problem on a polytope which takes our homotopy set $\mathscr{H}(t)$, $t \geqslant 0$, as a special case. Therefore the algorithm converges under the same condition.

What remains is to choose a value of the number a. In case $z^0 = 0$, the algorithm coincides with Lemke's algorithm. In this case pivot steps need only to be made in system (3.2) and the number a can be set equal to one. Suppose now that $z^0 \neq 0$. In that case we already put one limitation on a, $a > e^{\mathrm{T}} z^0$, being independent of the data of the problem as defined in (1.1) but guaranteeing that each $A(\mathscr{T})$, $\mathscr{T} \subset \{1, \ldots, n+1\}$, is convex. To make the choice of a dependent on the data of the problem, i.e., on M and q, we suggest to choose a such that for all $j \in \{1, \ldots, n\}$ no $\{j\}$-complete points in $A^0(\{j\})$ can be found.

Let a be such that for $j \in \{1, \ldots, n\}$ no j-complete points in $A^0(\{j\})$ exist. This implies that for every j the system (3.2) for $\mathscr{T}$ equal to $\{j\}$,

$$\lambda_j a M_{\cdot j} - \sum_{h \neq j,\, n+1} \mu_h e(h) + \theta e = -q, \tag{4.1}$$

may not have a solution $\lambda_j \geqslant 1$, $\mu_h \geqslant 0$ $(h \neq j, n+1)$, $\theta \geqslant 0$. The following condition on a assures that for any j at a solution to (4.1) for $\lambda_j > 1$ it holds that $\theta < 0$ or $\mu_h < 0$ for some $h \neq j, n+1$. Take $a > \max\{e^{\mathrm{T}} z^0, a_1, \ldots, a_n\}$ where a_j, $j = 1, \ldots, n$, are chosen such that

$$\begin{aligned}
&\text{if } M_{jj} > 0, \text{ then } a_j > \min\left\{\frac{-q_j}{M_{jj}}, \min_{h : M_{hj} < M_{jj}} \left\{\frac{q_h - q_j}{M_{jj} - M_{hj}}\right\}\right\},\\
&\text{if } M_{jj} = 0, \text{ then } a_j > \min_{h : M_{hj} < 0} \left\{\frac{q_j - q_h}{M_{hj}}\right\},\\
&\text{if } M_{jj} < 0, \text{ then } a_j \geqslant \min_{h : M_{hj} > M_{jj}} \left\{\frac{q_h - q_j}{M_{hj} - M_{jj}}\right\},
\end{aligned} \tag{4.2}$$

for every $j \in \{1, \ldots, n\}$. If these conditions do not hold it is possible that for some $j \in \{1, \ldots, n\}$ the number a_j can not be calculated according to (4.2). In that case by choosing a arbitrarily larger than $e^{\mathrm{T}} z^0$ one knows in advance that the algorithm could diverge and that the linear complementarity problem might not even have a solution.

We remark that instead of a uniform number a for all j we could also choose different numbers a_j for every j and take $\bar{r}(j) = a_j e(j) - z^0$, e.g. by choosing a_j as calculated in (4.2). In that case the algorithm follows a path of stationary points of g on $\bar{\mathscr{H}}(t) \cap \mathbb{R}^n_+$ for a varying parameter $t \geqslant 0$ starting for $t = 0$, where

$$\bar{\mathscr{H}}(t) = \left\{ z^0 + \sum_{j=1}^{n+1} \lambda_j \bar{r}(j) \,\middle|\, \lambda_j \geqslant 0 \text{ for } j \in \{1, \ldots, n+1\}, \text{ and } \sum_{j=1}^{n+1} \lambda_j \leqslant t \right\}.$$

5. Some examples

In this section the algorithm is illustrated by three examples. In each of these examples a possible path to be followed by the algorithm is given and the relation between the points on this path and the parametrized set is described.

In Section 2 we explained that every point on the path followed by the algorithm is a stationary point of the affine function g on the parametrized set $\mathscr{H}(t) \cap \mathbb{R}^n_+$ for some value of the parameter $t \geqslant 0$. In Figure 3 we have drawn the parametrized set $\mathscr{H}(t) \cap \mathbb{R}^n_+$ for some value of the parameter $t > 0$ in the two-dimensional case. If a point $\bar{z}$ is a stationary point of the affine function g on $\mathscr{H}(t) \cap \mathbb{R}^2_+$, then $\bar{z}$ maximizes $z^{\mathrm{T}} g(\bar{z})$ over $\mathscr{H}(\bar{t}) \cap \mathbb{R}^2_+$ where $\bar{t}$ is such that $\bar{z} \in \mathrm{bd}(\mathscr{H}(\bar{t}) \cap \mathbb{R}^2_+)$. Figure 3 shows the relation between $g(\bar{z})$ and such a stationary point $\bar{z}$ in $\mathrm{bd}(\mathscr{H}(\bar{t}) \cap \mathbb{R}^2_+)$ for $\bar{t} > 0$. Notice that $\bar{z} \in [c, d]$ when $\bar{t} > 1$.

If $\bar{z} = b$ for some $1 \geqslant \bar{t} > 0$, then $g(\bar{z})$ has to lie in the cone spanned by the vectors $-e(1)$ and $-e(2)$, i.e. there exists a $\mu_1 \geqslant 0$ and $\mu_2 \geqslant 0$ such that $g(\bar{z}) = -\mu_1 e(1) - \mu_2 e(2)$. If $\bar{z} = c$ for some $\bar{t} > 0$, then $g(\bar{z})$ has to lie in the cone spanned by the vectors $-e(1)$ and e,

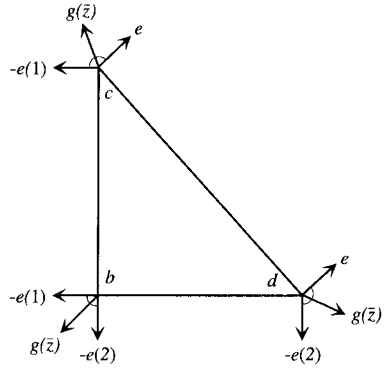

Fig. 3. Stationary points of g on $\mathscr{H}(\bar{t}) \cap \mathbb{R}^2_+$.

i.e. there exists a $\mu_1 \geqslant 0$ and $\theta \geqslant 0$ such that $g(\bar{z}) = -\mu_1 e(1) + \theta e$. If $\bar{z} \in (b, c)$ for some $1 \geqslant \bar{t} > 0$, then $g(\bar{z})$ has to lie in the cone spanned by the vector $-e(1)$, i.e. there exists a $\mu_1 \geqslant 0$ such that $g(\bar{z}) = -\mu_1 e(1)$. If $\bar{z} = d$ for some $\bar{t} > 0$, then $g(\bar{z})$ has to lie in the cone spanned by the vectors $-e(2)$ and e, i.e. there exists a $\mu_2 \geqslant 0$ and $\theta \geqslant 0$ such that $g(\bar{z}) = -\mu_2 e(2) + \theta e$. If $\bar{z} \in (c, d)$ for some $\bar{t} > 0$, then $g(\bar{z})$ has to lie in the cone spanned by the vector e, i.e. there exists a $\theta \geqslant 0$ such that $g(\bar{z}) = \theta e$. Finally, if $\bar{z} \in (b, d)$ for some $1 \geqslant \bar{t} > 0$, then $g(\bar{z})$ has to lie in the cone spanned by the vector $-e(2)$, i.e. there exists a $\mu_2 \geqslant 0$ such that $g(\bar{z}) = -\mu_2 e(2)$.

Example 5.1. Let LCP(q, M) be given by the following data.

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} -1 & -9 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} + \begin{pmatrix} 14 \\ -1 \end{pmatrix},$$

$$w^{\mathrm{T}} z = 0, \tag{5.1}$$

$$w \geqslant 0, \qquad z \geqslant 0.$$

To solve LCP() *in* (5.1), *the algorithm starts in* $z^0 = (3, 2)^{\mathrm{T}}$. Choose $a = 7$. Then the algorithm follows the path given by the bold-faced line in Figure 4.

In z^0 it holds that $g_1(z^0) = 7 > 0$ and $g_2(z^0) = -4 < 0$. With $w^0 = Mz^0 + q$ this implies that $w_1^0 = -7 = \min_h w_h^0 < 0$ and $w_2^0 = 4 > \min_h w_h^0$ causing the algorithm to leave the starting point z^0 by increasing z_1 from z_1^0 and lowering z_2 from z_2^0 along the direction $r(1) = ae(1) - z^0 = (4, -2)^{\mathrm{T}}$. Leaving z^0 along $r(1)$ implies that the algorithm generates points z in $A(\{1\})$. Such points z are points on $\mathrm{bd}(\mathscr{H}(t) \cap \mathbb{R}^2_+)$ for some $1 \geqslant t > 0$ comparable to the point d in Figure 3. Hence, according to the stationary point condition, the algorithm starts to generate the points z in $A(\{1\})$ for which $g(z) = \theta e - \mu_2 e(2)$ for some $\theta \geqslant 0$ and $\mu_2 \geqslant 0$. In Figure 4 these are all the points between z^0 and $z^1 = (5, 1)^{\mathrm{T}}$.

In z^1 it holds that $g(z^1) = (0, -5)^{\mathrm{T}}$. Hence, $\theta = 0$ and $g(z^1) = -\mu_2 e(2)$ for some $\mu_2 > 0$, namely $\mu_2 = -5$. Moving further in $A(\{1\})$ would imply $\theta < 0$ thereby violating the stationary point condition. Since $g(z^1) = -\mu_2 e(2)$ for some $\mu_2 > 0$, z^1 can be compared to

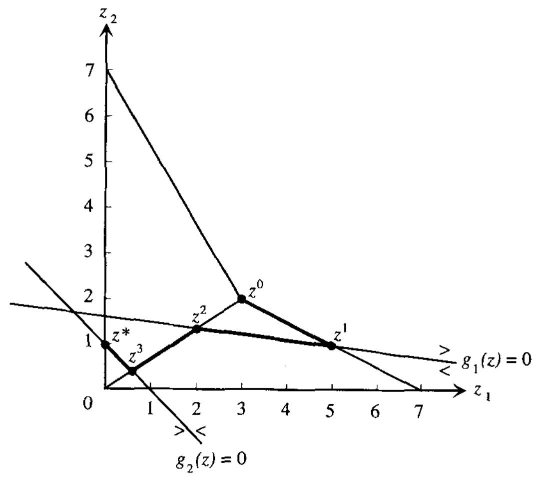

Fig. 4. The path followed by the algorithm to solve LCP(q, M) in (5.1) with $a=7$. It starts in $z^0=(3, 2)^T$. The path is given by the bold-faced line.

the end point d of $[d, b]$ in Figure 3. The algorithm therefore proceeds by generating points z represented by the points $[d, b]$ of bd$(\mathscr{H}(t) \cap \mathbb{R}^2_+)$ for $1 \geqslant t > 0$ in Figure 3, i.e. points z in $A(\{1, 3\})$ such that $g_1(z) = 0$ and $g_2(z) < 0$. In Figure 4 these are all the points between z^1 and $z^2 = (2, \frac{4}{3})^T$ where $g(z^2) = (0, -\frac{7}{3})^T$.

If the algorithm would generate points z beyond z^2 satisfying $g_1(z) = 0$ and $z_1 < z_1^2$, then it generates points on bd$(\mathscr{H}(t) \cap \mathbb{R}^2_+)$ for $t > 0$ comparable to points on $[b, c]$ in Figure 3 thereby violating the stationary point condition. Since z^2 can be compared to the point b in Figure 3, it holds that z^2 lies in $A(\{3\})$. In order to maintain the condition that each point z on the path generated by the algorithm is a stationary point of g on $\mathscr{H}(t) \cap \mathbb{R}^2_+$ for some $t > 0$, the algorithm next generates points in $A(\{3\})$ such that $g(z) = -\mu_1 e(1) - \mu_2 e(2)$ for $\mu_1 \geqslant 0$ and $\mu_2 \geqslant 0$, by raising μ_1 from zero. This implies that $g_1(z)$ is decreased from zero. Hence, the algorithm proceeds by generating the points between z^2 and $z^3 = (\frac{3}{5}, \frac{2}{5})^T$ in Figure 4.

In z^3 it holds that $g(z^3) = (-9\frac{4}{5}, 0)^T$, so the point z^3 can be compared to the point b on bd$(\mathscr{H}(t) \cap \mathbb{R}^2_+)$ for some $1 \geqslant t > 0$ in Figure 3. Since in z^3 it holds that $g_2(z^3) = 0$ it follows that $g(z^3) = -\mu_1 e(1)$ for some $\mu_1 > 0$, namely $\mu_1 = 9\frac{4}{5}$. The algorithm proceeds by generating points $z \in A(\{2, 3\})$ such that $g(z) = -\mu_1 e(1)$ for $\mu_1 \geqslant 0$. These are all the points between z^3 and $z^* = (0, 1)^T$ and they correspond to points between b and c in Figure 3.

The point z^* is a point on bd$(\mathscr{H}(1) \cap \mathbb{R}^2_+)$ comparable to a point between b and c in Figure 3. In z^* it holds that $g(z^*) = (-5, 0)^T$. Hence, the algorithm stops with the solution, $w^* = (5, 0)^T$ and $z^* = (0, 1)^T$ to LCP(q, M) in (5.1).

Example 5.2. Let LCP(q, M) be given by the following data.

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} 3 & -3 \\ 5 & -2 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} + \begin{pmatrix} -6 \\ -12 \end{pmatrix},$$
$$w^{\mathrm{T}} z = 0, \tag{5.2}$$
$$w \geqslant 0, \qquad z \geqslant 0.$$

To solve LCP() *in* (5.2), *the algorithm starts in* $z^0 = (1, 2)^{\mathrm{T}}$. Choose $a = 5$. Then the algorithm follows the path given by the bold-faced line in Figure 5.

In z^0 it holds that $g(z^0) = (9, 11)^{\mathrm{T}}$. Hence, $g_1(z^0) > 0$ and $g_2(z^0) > g_1(z^0)$. With $w^0 = Mz^0 + q$ this implies that $w_1^0 = -9 > -11 = \min_h w_h^0$ and $w_2^0 = -11 = \min_h w_h^0 < 0$. Hence, the algorithm leaves the starting point z^0 by lowering z_1 from z_1^0 and raising z_2 from z_2^0 along the direction $r(2) = ae(2) - z^0 = (-1, 3)^{\mathrm{T}}$. It thereby generates points in $A(\{2\})$. Each of these points is comparable to the point c on $\mathrm{bd}(\mathscr{H}(t) \cap \mathbb{R}_+^2)$ for some $1 \geqslant t > 0$ in Figure 3. The points z in $A(\{2\})$ are followed by the algorithm as long as z is a stationary point of g on $\mathscr{H}(t) \cap \mathbb{R}_+^2$ for some $t > 0$, i.e. as long as there exists a $\theta \geqslant 0$ and $\mu_1 \geqslant 0$ such that $g(z) = \theta e - \mu_1 e(1)$. This is the case for all z between z^0 and $z^1 = (0, 5)^{\mathrm{T}}$.

In $z^1 = ae(2)$ the boundary of $A(\{2\})$ has been reached since $z_1^1 = 0$. Moving on in the direction $r(2)$ would generate infeasible points. Therefore the algorithm proceeds along the boundary of $\mathbb{R}_+^2$ by generating points in $A^0(\{2\})$. Each of these points is comparable to the point c of $\mathscr{H}(t) \cap \mathbb{R}_+^2$ for some $t \geqslant 1$ in Figure 3. From the stationary point condition it follows that for all points $z \in A^0(\{2\})$ generated by the algorithm there exists a $\theta \geqslant 0$ and $\mu_1 \geqslant 0$ such that $g(z) = \theta e - \mu_1 e(1)$. This is the case for all $z \in A^0(\{2\})$ between z^1 and $z^2 = (0, 6)^{\mathrm{T}}$.

In z^2 it holds that $g(z^2) = (24, 24)^{\mathrm{T}}$. Hence, $g_1(z^2) = g_2(z^2)$, implying that μ_1 has become zero in z^2. Moving beyond z^2 in $A^0(\{2\})$ would cause μ_1 to become negative.

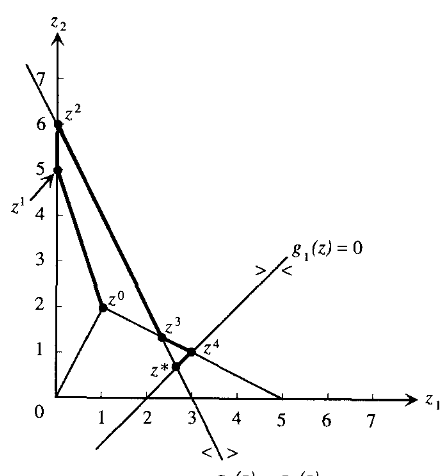

Fig. 5. The path followed by the algorithm to solve LCP(q, M) in (5.2) with $a = 5$. It starts in $z^0 = (1, 2)^{\mathrm{T}}$. The path is given by the bold-faced line.

Since μ_1 becoming zero implies that $g(z^2)=\theta e$ for some $\theta>0$, the point z^2 is comparable to the end point in c in $[c, d]$ on $\mathrm{bd}(\mathscr{H}(t)\cap\mathbb{R}^2_+)$ for some $t>0$ in Figure 3. The stationary points of g comparable to a point in $[c, d]$ on $\mathrm{bd}(\mathscr{H}(t)\cap\mathbb{R}^2_+)$ for some $t>0$ in Figure 3 are all the points $z\in A(\{1,2\})$ such that $g(z)=\theta e$ for $\theta\geqslant 0$. These are the points between z^2 and $z^3=(\frac{7}{3},\frac{4}{3})^{\mathrm{T}}$ in Figure 5, where $g(z^3)=(3,3)^{\mathrm{T}}$.

Moving at z^3 further along the line of points z on which $g_1(z)=g_2(z)$ by generating points between z^3 and $z^*=(\frac{8}{3},\frac{2}{3})^{\mathrm{T}}$ would imply that the algorithm generates points z which can be compared to points in $[d, b]$ on $\mathrm{bd}(\mathscr{H}(t)\cap\mathbb{R}^2_+)$ for $t>0$ in Figure 3 satisfying $g(z)=\theta e$ for some $\theta\geqslant 0$, violating the stationary point condition. However, since $z^3\in A(\{1\})$, z^3 is a point which can be compared to the point d on $\mathrm{bd}(\mathscr{H}(t)\cap\mathbb{R}^2_+)$ for some $t>0$ in Figure 3. The algorithm therefore proceeds by generating points $z\in A(\{1\})$ such that $g(z)=\theta e-\mu_2 e(2)$ for $\theta\geqslant 0$ and $\mu_2\geqslant 0$ by raising μ_2 from zero. These are the points between z^3 and $z^4=(3,1)^{\mathrm{T}}$ where $g(z^4)=(0,-1)^{\mathrm{T}}$.

In z^4 we obtain a similar situation as in z^1 in Example 5.1 since $g_1(z^4)=0$. As in Example 5.1 the algorithm proceeds in $A(\{1,3\})$ by generating the points from z^4 to $z^*=(\frac{8}{3},\frac{2}{3})^{\mathrm{T}}$. In z^* it holds that $g_1(z^*)=g_2(z^*)=0$ implying the algorithm to stop in z^* with the solution $w^*=(0,0)^{\mathrm{T}}$ and $z^*=(\frac{8}{3},\frac{2}{3})^{\mathrm{T}}$ to the linear complementarity problem.

Notice that the transition at z^1 from $A(\{2\})$ to $A^0(\{2\})$ causes a change of pivot systems as described in Case 3 of Section 3. At the point z^2 the algorithm starts to follow the line segment of points z in $A(\{1,2\})$ such that $g_1(z)=g_2(z)\geqslant 0$. In our example the algorithm generates these points in $A(\{1,2\})$ as linearly independent combinations of $e(1)$ and $e(2)$. The transition from $A(\{1,2\})$ to $A(\{1\})$ causes a change of variables as presented in Case 4 of Section 3.

In case z^2 would be lying in $A(\{2\})$ no change of variables at z^1 or z^3 is needed. If, however, in this case z^3 would lie in $A^0(\{1\})$, then the points generated by the algorithm from z^3 or in $A^0(\{1\})$ require a change of variables at z^3 as presented in Case 2 of Section 3 since the points in $A^0(\{1\})$ should be generated as multiples of $e(1)$. Notice further that if we apply (4.2) the number a would have been chosen at least equal to 6, which would avoid making a pivoting step in $A^0(\{2\})$.

To illustrate the algorithm introduced in the previous sections we constructed Example 5.1 and Example 5.2 in such a way that the algorithm has to perform as many iterations as possible. Consequently, Example 5.1 and Example 5.2 do not give much credit to the speed with which the algorithm finds a solution to a linear complementarity problem. In fact, both examples are solved faster by the Lemke complementary pivoting algorithm. Example 5.1 is solved by the Lemke complementary pivoting algorithm in one iteration while Example 5.2 is solved by the Lemke complementary pivoting algorithm in two iterations.

On the other hand situations where the Lemke complementary pivoting algorithm is much slower than our algorithm can be constructed very easily. In Murty [7] an example is given for which the Lemke complementary pivoting algorithm needs the maximum number of 2^n-1 steps to find the solution, n being the dimension of the linear complementarity problem. In Example 5.3 we give the two-dimensional version of Murty's example.

Example 5.3. In Murty [7] the linear complementarity problem to be solved in the two-dimensional case is given by

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} + \begin{pmatrix} -4 \\ -6 \end{pmatrix},$$

$$w^{\mathrm{T}} z = 0, \tag{5.3}$$

$$w \geqslant 0, \qquad z \geqslant 0.$$

The linear complementarity problem in (5.3) has a solution $w^* = (0, 2)^{\mathrm{T}}$ and $z^* = (4, 0)^{\mathrm{T}}$.

The Lemke complementary pivoting algorithm needs three steps to solve this two-dimensional linear complementarity problem. In Figure 6 we have drawn the path followed by the Lemke complementary pivoting algorithm. If we apply the algorithm introduced in this paper to solve the linear complementarity problem in (5.3) then we are able to use an arbitrarily chosen starting point. Suppose we take $z^0 = (3, 1)^{\mathrm{T}}$ as the starting point of our algorithm. To choose the intersection of the rays $r(1)$ and $r(2)$ with the axes of $\mathbb{R}^2_+$ we

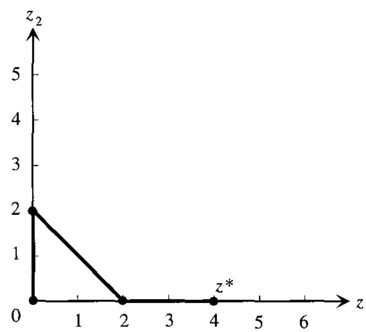

Fig. 6. The path followed by the Lemke complementary pivoting algorithm to solve the linear complementarity problem in (5.3).

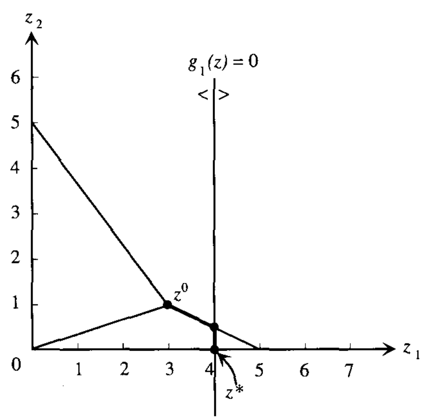

Fig. 7. The path followed by the new algorithm to solve the linear complementarity problem in (5.3).

have to make a suitable choice for a according to (4.2). This means we have to choose a such that $a > \max\{4, 4, 2\} = 4$. Hence, we may choose $a = 5$. Figure 7 gives the path followed by our algorithm to solve the linear complementarity problem in (5.3). In this case the algorithm only needs two steps to solve the linear complementarity problem in (5.3). Notice also that if we start the new algorithm in, for example, $(3, 0)^T$ then it finds the solution in only one step. The latter result holds in fact for any dimension of the problem.

This example gives the opposite results to the first two examples when comparing our algorithm to the Lemke complementary pivoting algorithm. It is this kind of linear complementarity problem which might often be met when solving a nonlinear problem by a sequence of linear complementarity problems. In case of convergence of this sequence, the approximation of the solution to the nonlinear problem obtained from a linear complementarity problem in the sequence will in due time be near to the solution itself. If one is able to start the algorithm for solving the next linear complementarity problem in the sequence in this approximating solution then the new approximation is typically found within a few iterations, probably contrary to when using the Lemke complementary pivoting algorithm. That method is forced to restart in the origin all over again. It is this kind of problem we had in mind when introducing the new algorithm.

References

[1] B.C. Eaves, "On the basic theorem of complementarity," *Mathematical Programming* 1 (1971) 68–75.

[2] B.C. Eaves, "Computing stationary points," *Mathematical Programming Study* 7 (1978) 1–14.

[3] B.C. Eaves, "Computing stationary points, again," in: O.L. Mangasarian, R.R. Meyer and S.M. Robinson, eds., *Nonlinear Programming* 3 (Academic Press, New York, 1978) pp. 391–405.

[4] M.S. Gowda and J.S. Pang, "The basic theorem of complementarity revisited," *Mathematical Programming* 58 (1993) 149–160.

[5] C.E. Lemke, "Bimatrix equilibrium points and mathematical programming," *Management Science* 11 (1964) 681–689.

[6] L. Mathiesen, "Computation of economic equilibria by a sequence of linear complementarity problems," *Mathematical Programming* 23 (1985) 144–162.

[7] K.G. Murty, "Computational complexity of complementary pivot methods," *Mathematical Programming Study* 7 (1978) 61–73.

[8] A.J.J. Talman and L. Van der Heyden, "Algorithms for the linear complementarity problem which allow an arbitrary starting point," in: B.C. Eaves et al., eds., *Homotopy Methods and Global Convergence* (Plenum Press, New York, 1983) pp. 267–285.