A new simplicial variable dimension algorithm to find equilibria on the product space of unit simplices

Doup, T.M.; Talman, A.J.J.

Published in:
Mathematical Programming

Publication date:
1987

Mathematical Programming 37 (1987) 319-355
North-Holland

A NEW SIMPLICIAL VARIABLE DIMENSION ALGORITHM TO FIND EQUILIBRIA ON THE PRODUCT SPACE OF UNIT SIMPLICES

T.M. DOUP* and A.J.J. TALMAN
Department of Econometrics, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands

Received 16 April 1984
Revised manuscript received 17 November 1986

A new simplicial variable dimension restart algorithm is introduced to solve the nonlinear complementarity problem on the product space S of unit simplices. The triangulation which underlies the algorithm differs from the triangulations of S used thus far. Moreover, the number of rays along which the algorithm can leave the arbitrarily chosen starting point is much larger. More precisely, there is a ray leading from the starting point to each vertex of S. In case S is the product of n one-dimensional unit simplices the alogrithm is similar to the octahedral algorithm on R^n having 2^n rays. Also, the accuracy of an approximate solution in the terminal simplex of the algorithm is in general better than for the other algorithms on S. Computational results will show that the number of iterations for the new algorithm is much less. The examples concern the computation of equilibria in noncooperative games, exchange economies and trade models.

Key words: Nonlinear complementarity problem, triangulation, vector labelling.

1. Introduction

To compute fixed points of continuous functions from the product space S of N unit simplices $S^{n_j} = \{x_j \in R_+^{n_j+1} | \sum_{k=1}^{n_j+1} x_{j,k} = 1\}$, $j = 1, \ldots, N$, in van der Laan and Talman [5] a simplicial variable dimension algorithm was introduced. This algorithm is a generalization of an algorithm introduced in van der Laan and Talman [2] solving fixed point problems on a unit simplex. Such a variable dimension algorithm subdivides the set on which the problem is defined into simplices, labels the vertices of the simplices and searches for a simplex which yields an approximate solution to the problem. Starting from an arbitrary point a path of adjacent simplices of varying dimension is generated until such a simplex is found. To guarantee the existence of such a simplex the vertices on the boundary must satisfy a so-called properness condition. Recently in van der Laan, Talman and Van der Heyden [7] the algorithm on S was generalized for general labellings in order to solve the

* This author is financially supported by the Netherlands Organisation for the Advancement of Pure Research (ZWO), Grant 46-98.

This research is part of the VF-program "Equilibrium and Disequilibrium in Demand and Supply" which has been approved by the Netherlands ministry of education and sciences.

nonlinear complementarity problem on S. Typically this problem has solutions on the boundary of S. The number of one-dimensional sets (rays) along which the algorithm in [7] can leave the initial point is equal to the number of variables, $\sum_{j=1}^{N}(n_j+1)$. In case $N=1$, $n_1=n$, there is a ray leading from the initial point to each of the $n+1$ vertices of S^n. Due to the underlying so-called Q-triangulation of S^n these rays are broken lines. It would be more natural to have straight lines from the starting point to the vertices of S^n.

The basic algorithm on S with proper labelling was adapted to a class of algorithms for computing zero points (or fixed points) of continuous functions from R^n to R^n in van der Laan and Talman [4]. The so-called $(n+1)$-ray and $2n$-ray algorithm are special cases of this class where there are $n+1$ and $2n$ rays resp. to leave the arbitrarily chosen starting point in R^n.

Subsequently Wright [13] introduced the 2^n-ray or octahedral algorithm. To solve the nonlinear complementarity problem (NLCP) with lower and upper bounds both the $2n$-ray and 2^n-ray algorithm were adapted in [6]. The set C^n on which this problem is defined can be seen as the product space of n one-dimensional simplices. In case of the 2^n-ray algorithm there is a ray from the starting point to each of the 2^n vertices of C^n. Due to the underlying K'-triangulation (see [11]) these rays are again broken lines.

In this paper we present a simplicial variable dimension algorithm to solve the nonlinear complementarity problem on S with rays leading from the starting point to each of the $\prod_{j=1}^{N}(n_j+1)$ vertices of S. The new triangulation of S which underlies the algorithm is called the V-triangulation and induces rays which are straight lines from the starting point to the vertices of S. In addition the algorithm allows for generalized labellings. The new algorithm, which we call the product-ray algorithm, generates a sequence of adjacent simplices of varying dimension of the V-triangulation until an approximating simplex is found. This path is followed by alternating replacement steps in the triangulation and linear programming pivoting steps in a system of $\sum_{j=1}^{N}(n_j+1)+1$ equations. This system differs from the system of equations used in the algorithm on S described in [7]. When $N=1$ the system is equivalent to the latter system, although the path of generated simplices differs since another triangulation underlies the algorithm. For $N>1$, however, the systems are not equivalent. As will be argued below, the accuracy of the approximate solution at the terminal simplex of the new algorithm is typically better than the accuracy of such a solution for the algorithm described in [7]. So, to find an approximate solution with a given accuracy, we may expect that the new algorithm needs fewer restarts and therefore less function evaluations. Computational results for three different applications confirm that the new algorithm performs better even in the case $N=1$. The first application concerns the computation of Nash equilibrium vectors in noncooperative N-person games. The second application deals with a pure exchange economy and the last one with an international trade model.

The paper is organized as follows. In Section 2 a description of the V-triangulation of the unit simplex is given. Section 3 describes the V-triangulation of the product

space S of unit simplices and gives the path of points followed by the algorithm to find an approximate solution of the NLCP on S. A comparison is made with the algorithm described in [7]. Section 4 describes the steps of the algorithm to follow this path. Some variants of the triangulation are discussed in Section 5. Special attention is drawn to the case $N = 1$ when S is a unit simplex. Finally, computational results are given in Section 6.

2. The V-triangulation of the unit simplex

The problem discussed in this section is the so-called nonlinear complementarity problem (NLCP) on the n-dimensional unit simplex S^n. This problem can be stated as follows:

let z be a continuous function from S^n into R^{n+1} satisfying $x^T z(x) = 0$ for all x in S^n. Find an x^* in S^n such that $z(x^*) \leq 0$.

This problem is equivalent to the Brouwer fixed point problem on S^n.

To solve the fixed point problem on S^n two so-called simplicial variable dimension restart algorithms have been developed by van der Laan and Talman, see [2, 3]. These algorithms differ in the underlying triangulation of S^n, the so-called Q- and the U-triangulation resp. The latter triangulation is a triangulation of the affine hull of S^n and does not triangulate (the boundary of) S^n. These algorithms were developed for solution points in the interior of S^n.

In general the NLCP on S^n can have solutions on the boundary of S^n. Van der Laan, Talman and Van der Heyden [7] introduced a simplicial algorithm to solve the NLCP on S^n. The Q-triangulation underlies the algorithm and the algorithm allows for movements on the boundary of S^n. A drawback lies in the underlying triangulation. Let i be the index in the set $\{1, \ldots, n+1\}$ for which $z_i(v) = \max_h z_h(v)$, where v is the arbitrary starting point of the algorithm. For the moment we assume that this index is unique. Then the i-th component of v is initially increased and exactly one other component of v is decreased with the same amount in order to stay in S^n and to utilize the Q-triangulation of S^n. More precisely, the one-dimensional piecewise linear set $\tilde{A}(\{i\})$ along which the algorithm leaves the starting point v when $z_i(v) = \max_h z_h(v)$ connects v with the vertex $e(i)$ of S^n by increasing the i-th component of v to one and by successively decreasing the j-th component until zero for $j = i+1, \ldots, n+1, 1, \ldots, i-1$. This is illustrated for $n = 2$ in Fig. 2.1.

If $z_i(x)$ is maximal for all x in S^n for some index i, the algorithm proposed in [7] ends with the vertex $e(i)$ of S^n which is an exact solution for the NLCP. An increase of the i-th component of v for which $z_i(v)$ is maximal is obvious but the decrease with the same amount of exactly one other component of v seems to be arbitrary. A decrease of all the other components of v to compensate the increase of v_i is more natural. When the U-triangulation underlies the algorithm these other

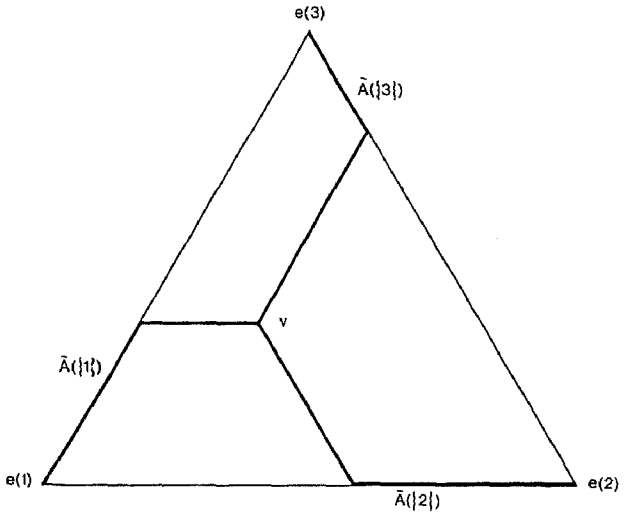

Fig. 2.1. The one-dimensional sets $\tilde{A}(\{i\})$, $i=1,\ldots,n+1$, of the algorithm proposed by van der Laan, Talman and Van der Heyden, $n=2$. $\tilde{A}(\{i\})$ connects v and $e(i)$, $i=1,\ldots,n+1$.

components are all decreased with the same amount. However this triangulation does not triangulate the boundary sets of S^n and is therefore not suitable for use in a simplicial algorithm for solving the NLCP on S^n. In this section we will propose a new triangulation which combines the advantages of both the Q- and U-triangulation in the sense that it triangulates S^n itself and therefore also the boundary of S^n and that it induces one-dimensional sets $A(\{i\})$ connecting v and the vertex $e(i)$ of S^n along which all the components v_j of v, $j\neq i$, are simultaneously decreased. This triangulation of S^n which we call the V-triangulation is generalized in Section 3 to a new triangulation of the product space S of unit simplices being the base for a new simplicial variable dimension restart algorithm for solving the NLCP on S.

The V-triangulation of S^n is such that along the induced sets $A(\{i\})$ connecting v and $e(i)$, $i=1,\ldots,n+1$, the components v_j of v, $j\neq i$, are proportionally decreased to zero so that $A(\{i\})$ is just the convex hull of v and $e(i)$, $i\in I_{n+1}=\{1,\ldots,n+1\}$. For $n=2$ the sets $A(\{i\}$ are illustrated in Fig. 2.2.

A proportional decrease of the components v_j of v, $j\neq i$, seems most natural if $z_i(v)=\max_h z_h(v)$ since if $z_i(x)=\max_h z_h(x)$ for all x in S^n a direct movement is made from v to $e(i)$ being the solution of this NLCP. Now, for $T\neq I_{n+1}$ let $A(T)$ be the convex hull of the point v and the vertices $e(i)$, $i\in T$. If not $z_i(x)=\max_h z_h(x)$ for all x in S^n the simplicial algorithm based on the V-triangulation of S^n will approximately generate points x in $A(T)$ such that $z_k(x)=\max_h z_h(x)$ for all k in T, for varying T, until an approximate solution is found. For $n=2$ the sets $A(T)$, $T\subset I_{n+1}$, are illustrated in Fig. 2.3.

We will now describe the V-triangulation of S^n in detail. The triangulation is such that each nonempty set $A(T)$, $T\subset I_{n+1}$, is subdivided into t-dimensional simplices or t-simplices where t is the cardinality of the set T, i.e. $t=|T|$.

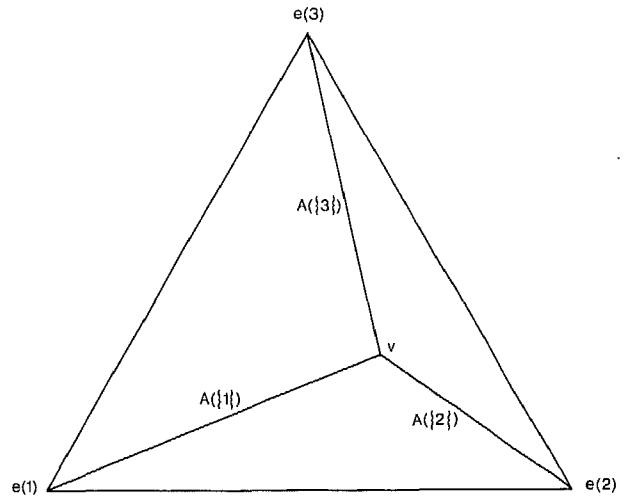

Fig. 2.2. The one-dimensional sets $A(\{i\})$ induced by the V-triangulation, $n=2$.

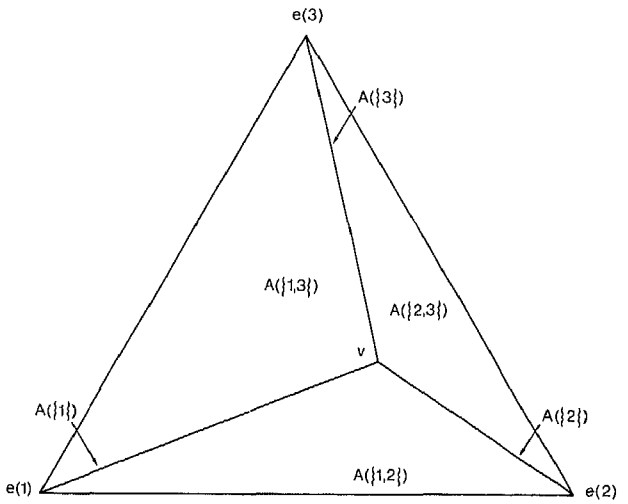

Fig. 2.3. The sets $A(T)$, $T \subset I_{n+1}$, induced by the V-triangulation of S^n, $n=2$.

Formally, for proper subsets T of I_{n+1} the set $A(T)$ is defined by

$$A(T) = \text{co}(\{v\}, \{e(i) | i \in T\})$$

where $\text{co}(\cdot)$ means the convex hull. If v lies in the convex hull of the $e(i)$'s, $i \in T$, we define $A(T)$ to be empty. Observe that the dimension of a nonempty $A(T)$ equals $t = |T|$.

To triangulate these nonempty regions $A(T)$, $T \subset I_{n+1}$, we introduce the projection $p(K)$ of v on the boundary face $S^n(K) = \{x \in S^n \,|\, x_i = 0, i \notin K\}$ of S^n, for nonempty

proper subsets K of I_{n+1}, defined by

$$p_h(K)=\begin{cases}\dfrac{1+|K^0|}{\left(\sum_{k\in K} v_k+|K^0|\right)}v_h, & h\in K\setminus K^0,\\[2ex] \dfrac{1-\sum_{k\in K} v_k}{\left(\sum_{k\in K} v_k+|K^0|\right)}, & h\in K^0,\\[2ex] 0, & h\notin K,\end{cases}$$

where K^0 is the set $\{h\in K \mid v_h=0\}$. Other projections will be discussed in Section 5. We define $p(\emptyset)=v$. Furthermore, let $\gamma(T)=(\gamma_0,\ldots,\gamma_{t-1})$ be a permutation of the t elements of T.

Definition 2.1. For $T\subset I_{n+1}$ the set $A(\gamma(T))$ is given by

$$A(\gamma(T))=\left\{x\in S^n \mid x=v+\sum_{i=0}^{t-1}\alpha_i q(\gamma_i) \text{ with } 0\leqq\alpha_{t-1}\leqq\cdots\leqq\alpha_0\leqq 1\right\}$$

where $q(\gamma_i)$, $i=0,\ldots,t-1$, is given by

$$q(\gamma_i)=p(\{\gamma_0,\ldots,\gamma_i\})-p(\{\gamma_0,\ldots,\gamma_{i-1}\}).$$

Observe that $q(\gamma_0)=e(\gamma_0)-v$. For $n=2$, some sets $A(\gamma(T))$, $T\subset I_{n+1}$, are illustrated in Fig. 2.4.

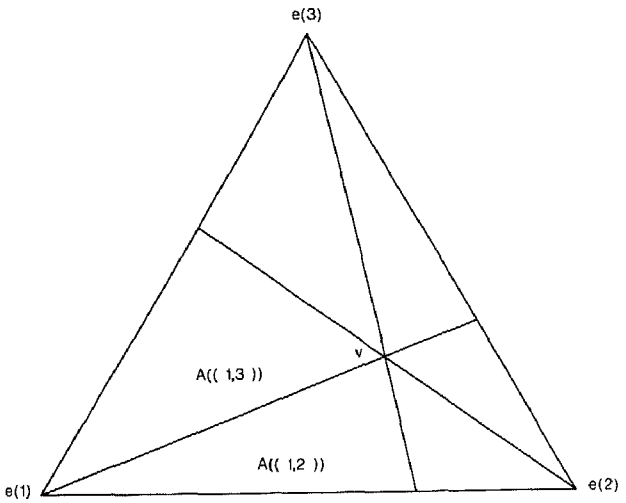

Fig. 2.4. Some regions $A(\gamma(T))$, $T\subset I_{n+1}$, in S^n, $n=2$.

$A(T)$ is the union of $A(\gamma(T))$ over all permutations $\gamma(T)$ of the t elements of T. The rank of the matrix $Q(\gamma(T)) = [q(\gamma_0), \ldots, q(\gamma_{t-1})]$ is less than t if and only if

$$v_h = 0 \quad \text{for all } h \in T.$$

The rank of this matrix is independent of the permutation $\gamma(T)$ and $A(T)$ is nonempty if and only if the rank of the matrix $Q(\gamma(T))$ is equal to t which coincides with the condition that v does not lie in the convex hull of $e(i)$, $i \in T$. The number of 1-dimensional regions $A(\{i\})$ is equal to $n+1$ but in the case that v is a vertex of S^n this number is n. In the sequel we restrict ourselves to sets $A(\gamma(T))$ having dimension t. Each nonempty $A(T)$, $T \subset I_{n+1}$, is triangulated by first triangulating $A(\gamma(T))$ in t-simplices for each permutation $\gamma(T)$ of the t elements of T and then taking the union over all permutations. Take some integer $m > 0$, m^{-1} will be the grid size of the triangulation.

Definition 2.2. For $T \subset I_{n+1}$, the set $G(\gamma(T))$ is the set of t-simplices $\sigma(y^1, \pi(T))$ with vertices $y^1, \ldots, y^{t+1}$ such that

(i) $y^1 = v + \sum_{i \in T} a(i) m^{-1} q(i)$ for nonnegative integers $a(i)$, $i \in T$, such that $0 \leqq a(\gamma_{t-1}) \leqq \cdots \leqq a(\gamma_0) \leqq m-1$;

(ii) $\pi(T) = (\pi_1, \ldots, \pi_t)$ is a permutation of the t elements of T such that for all $i = 1, \ldots, t-1$: $p' > p$ if $a(\gamma_{i-1}) = a(\gamma_i)$ where $\pi_p = \gamma_{i-1}$ and $\pi_{p'} = \gamma_i$;

(iii) $y^{i+1} = y^i + m^{-1} q(\pi_i)$, $i = 1, \ldots, t$.

It is clear that if $A(T)$ is nonempty $G(\gamma(T))$ is a triangulation of $A(\gamma(T))$ and that the union $G(T)$ of $G(\gamma(T))$ over all permutations $\gamma(T)$ of the elements of T triangulates $A(T)$ and finally that the union G of $G(T)$ over all sets T induces a triangulation of S^n. We call this triangulation the V-triangulation. For $n = 2$ and $m = 2$, G is illustrated in Fig. 2.5.

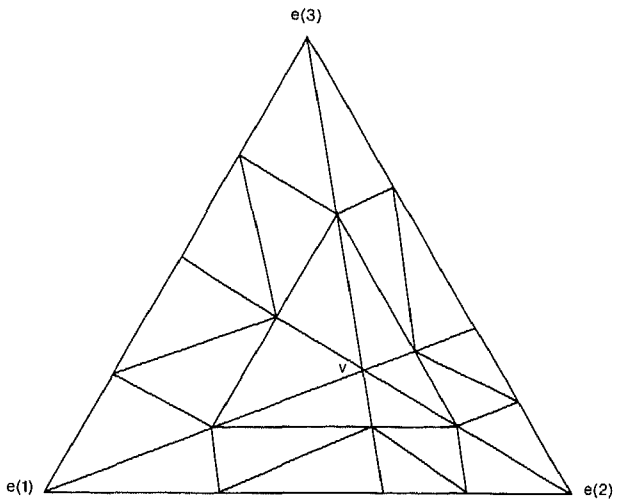

Fig. 2.5. The V-triangulation of S^n, $n = 2$, with grid size $\frac{1}{2}$.

The algorithm will now follow a sequence of adjacent t-simplices of $G(\gamma(T))$ for varying T such that their common facets are T-complete. In the next definition we allow T to be equal to I_{n+1}.

Definition 2.3. For $g = t-1$, t, where $t = |T|$, $T \subset I_{n+1}$, a g-simplex $\sigma(y^1, \ldots, y^{g+1})$ is T-complete if the system of linear equations

$$\sum_{i=1}^{g+1} \lambda_i \begin{pmatrix} z(y^i) \\ 1 \end{pmatrix} + \sum_{k \notin T} \mu_k \begin{pmatrix} e(k) \\ 0 \end{pmatrix} - \beta \begin{pmatrix} e \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{2.1}$$

where $e(k)$ denotes the k-th unit vector in R^{n+1} and $e = \sum_{k=1}^{n+1} e(k)$, has a solution λ_i^*, $i = 1, \ldots, g+1$, μ_k^*, $k \notin T$, and β^* with $\lambda_i^* \geq 0$, $i = 1, \ldots, g+1$, and $\mu_k^* \geq 0$, $k \notin T$.

A solution λ_i^*, $i = 1, \ldots, g+1$, μ_k^*, $k \notin T$, and β^* will be denoted by $(\lambda^*, \mu^*, \beta^*)$. For $g = t-1$ we assume that the system (2.1) has a unique solution with $\lambda_i^* > 0$, $i = 1, \ldots, t$, and $\mu_k^* > 0$, $k \notin T$, and that for $g = t$ at most one variable of (λ^*, μ^*) is equal to zero (nondegeneracy assumption).

Definition 2.4. A T-complete $(t-1)$-simplex $\sigma(y^1, \ldots, y^t)$, $T \subseteq I_{n+1}$, is complete if for all $x \in \sigma$: $x_h = 0$, $h \notin T$.

Observe that we again allow T to be equal to I_{n+1}. A complete simplex yields an approximate solution $x^* = \sum_{i=1}^{t} \lambda_i^* y^i$ as is shown at the end of this section.

Since a T-complete t-simplex has at most two T-complete facets and since a facet of a t-simplex in $A(\gamma(T))$ is a facet of at most one other t-simplex in $A(\gamma(T))$, we obtain that the T-complete t-simplices in $A(\gamma(T))$, for given permutation $\gamma(T)$ of T, $T \subset I_{n+1}$, $T \neq \emptyset$, determine sequences of adjacent simplices with T-complete common facets. Let $\sigma(y^1, \pi(T))$ and $\bar{\sigma}(\bar{y}^1, \bar{\pi}(T))$ be two adjacent t-simplices in $G(\gamma(T))$ with common facet τ opposite vertex y^p, $1 \leq p \leq t+1$, then $\bar{\sigma}$ can be obtained from σ as given in Table 1 where the $(n+1)$-vector a is given by $a_i = a(i)$, $i \in T$, and $a_i = 0$, $i \notin T$.

When moving from σ to $\bar{\sigma}$ we say that a replacement step is made with the vertex y^p. Each sequence is either a loop or has two end points. An end point is either a $T \cup \{k\}$-complete t-simplex in $A(\gamma(T))$ for some $k \notin T$, or a T-complete t-simplex with a T-complete facet in $\mathrm{bd}(A(\gamma(T)))$. If an end point $\sigma(y^1, \pi(T))$ in $A(\gamma(T))$ is

Table 1

p is the index of the vertex to be replaced

	$\bar{y}^1$	$\bar{\pi}(T)$	$\bar{a}$
$p = 1$	$y^1 + m^{-1} q(\pi_1)$	$(\pi_2, \ldots, \pi_t, \pi_1)$	$a + e(\pi_1)$
$1 < p < t+1$	y^1	$(\pi_1, \ldots, \pi_{p-2}, \pi_p, \pi_{p-1}, \ldots, \pi_t)$	a
$p = t+1$	$y^1 - m^{-1} q(\pi_t)$	$(\pi_t, \pi_1, \ldots, \pi_{t-1})$	$a - e(\pi_t)$

a $T\cup\{k\}$-complete t-simplex for some $k\notin T$, then either σ is complete if $v_h=0$ for all $h\notin T\cup\{k\}$ or σ is a $T\cup\{k\}$-complete facet of exactly one $T\cup\{k\}$-complete $(t+1)$-simplex $\bar{\sigma}$ in $A(T\cup\{k\})$. More precisely $\bar{\sigma}$ is the $(t+1)$-simplex $\bar{\sigma}(y^1, \pi(T\cup\{k\}))$ in $A(\gamma(T\cup\{k\}))$ where $\gamma(T\cup\{k\})=(\gamma_0,\ldots,\gamma_{t-1},k)$ and $\pi(T\cup\{k\})=(\pi_1,\ldots,\pi_t,k)$ and is therefore an end point of a sequence of adjacent $T\cup\{k\}$-complete $(t+1)$-simplices in $A(\gamma(T\cup\{k\}))$ with $T\cup\{k\}$-complete common facets. The next lemma describes when a facet of the t-simplex $\sigma(y^1,\pi(T))$ in $A(\gamma(T))$ lies in the boundary of $A(\gamma(T))$.

Lemma 2.5. *Let $\sigma(y^1,\pi(T))$ be in $G(\gamma(T))$ and τ the facet of σ opposite vertex y^p, $1\leqslant p\leqslant t+1$. Then τ lies on the boundary of $A(\gamma(T))$ if and only if*

(i) $p=1$: $\pi_1=\gamma_0$ *and* $a(\pi_1)=m-1$;

(ii) $1<p<t+1$: $\pi_{p-1}=\gamma_{i-1}$, $\pi_p=\gamma_i$ *for some* i, $1\leqslant i\leqslant t-1$, *and* $a(\pi_{p-1})=a(\pi_p)$;

(iii) $p=t+1$: $\pi_t=\gamma_{t-1}$ *and* $a(\pi_t)=0$.

The lemma follows immediately from the definitions of $G(\gamma(T))$ and $A(\gamma(T))$. If $\sigma(y^1,\pi(T))$ in $A(\gamma(T))$ is an end point of a sequence with a T-complete facet τ opposite vertex y^p, $1\leqslant p\leqslant t+1$, then in case (i) τ is a T-complete $(t-1)$-simplex in $S^n(T)=\{x\in S^n\,|\,x_h=0, h\notin T\}$ and therefore complete. In case (ii) τ is a T-complete facet of exactly one other t-simplex, say $\bar{\sigma}$, in $A(T)$. More precisely, $\bar{\sigma}$ is the t-simplex $\bar{\sigma}(y^1,\bar{\pi}(T))$ in $A(\bar{\gamma}(T))$ where $\bar{\gamma}(T)=(\gamma_0,\ldots,\gamma_{i-2},\gamma_i,\gamma_{i-1},\ldots,\gamma_{t-1})$ and $\bar{\pi}(T)=(\pi_1,\ldots,\pi_{p-2},\pi_p,\pi_{p-1},\ldots,\pi_t)$, and is therefore an end point of a sequence of adjacent T-complete t-simplices in $A(\bar{\gamma}(T))$ with T-complete common facets. In case (iii) the T-complete facet τ lies in $A(T\backslash\{h\})$, with $h=\pi_t$, so that τ is the $(t-1)$-simplex $\bar{\sigma}(y^1,\pi(T\backslash\{h\}))$ in $A(\gamma(T\backslash\{h\}))$ where $\gamma(T\backslash\{h\})=(\gamma_0,\ldots,\gamma_{t-2})$ and $\pi(T\backslash\{h\})=(\pi_1,\ldots,\pi_{t-1})$. Therefore τ is an end point of a sequence of adjacent $T\backslash\{h\}$-complete $(t-1)$-simplices in $A(\gamma(T\backslash\{h\}))$ with $T\backslash\{h\}$-complete common facets. In the case $|T|=1$ we have $\tau=\{v\}$ and there is no such sequence.

In this way the T-complete t-simplices in $A(\gamma(T))$, $T\subset I_{n+1}$, can be linked together for varying T to obtain sequences of adjacent simplices of varying dimension. Each sequence is either a loop or has two endpoints. Exactly one end point is the zero-dimensional simplex $\sigma(v)$ whereas all other end points are complete simplices. Therefore there is one sequence which connects the starting point v with a complete simplex. This sequence is followed by the algorithm, starting for $T=\emptyset$ in the point v, by performing alternating linear programming pivot steps in the linear system (2.1) and replacement steps in the triangulation. As soon as by an l.p. pivot step in (2.1) a μ_k, $k\notin T$, becomes zero, the current t-simplex $\sigma(y^1,\pi(T))$ is $T\cup\{k\}$-complete and if σ is not complete an l.p. pivot step is made in the linear system (2.1) with $(z^{\mathrm{T}}(y^{t+2}),1)^{\mathrm{T}}$ where y^{t+2} is the vertex of the unique $(t+1)$-simplex $\bar{\sigma}$ in $A(T\cup\{k\})$ having σ as a facet opposite this vertex. On the other hand, if for some $h\in T$ a T-complete facet τ in $A(T\backslash\{h\})$ is generated, then the unit vector $(e^{\mathrm{T}}(h),0)^{\mathrm{T}}$ is reintroduced in the linear system (2.1). This completes the description

of the algorithm to find, within a finite number of iterations, a complete simplex in the V-triangulation of S^n with grid size m^{-1}.

The algorithm in fact traces a piecewise linear path of points in S^n from the starting point v to an approximate solution which lies in the complete simplex generated by the algorithm. More precisely, let $\bar{z}$ be the piecewise linear approximation of z with respect to the underlying V-triangulation, i.e. if $x \in \sigma(y^1, \ldots, y^{t+1})$ so that $x = \sum_{i=1}^{t+1} \lambda_i y^i$ for certain nonnegative numbers λ_i, $i = 1, \ldots, t+1$, such that $\sum_{i=1}^{t+1} \lambda_i = 1$, then

$$\bar{z}(x) = \sum_{i=1}^{t+1} \lambda_i z(y^i).$$

The set of solutions (λ, μ, β) of the linear system (2.1) with respect to a t-simplex σ in $A(T)$ determine a line segment of points $\bar{x}$, $\bar{x} = \sum_i \lambda_i y^i$, in σ satisfying

$$\begin{aligned} &\bar{z}_k(\bar{x}) = \beta \qquad \text{if } k \in T, \\ &\bar{z}_k(\bar{x}) = \beta - \mu_k \quad \text{if } k \notin T. \end{aligned} \tag{2.2}$$

The end point of a line segment of points in σ is characterized by either $\lambda_p = 0$ for some p, $1 \leq p \leq t+1$, or $\mu_k = 0$ for some $k \notin T$. In the first case $x = \sum_i \lambda_i y^i$ lies in the facet τ opposite the vertex y^p of σ. Then x is an approximate solution if τ is complete or an end point of a line segment in either the unique t-simplex in $A(T)$ sharing τ with σ or in τ itself if τ lies in $A(T \backslash \{h\})$ for some $h \in T$. On the other hand, if at an end point x in σ $\mu_k = 0$ for some $k \notin T$, then x is an approximate solution if σ is complete or x is an end point of a line segment of points in the unique $(t+1)$-simplex in $A(T \cup \{k\})$ having σ as a facet. Therefore, the line segments of points induced by (2.1) for the sequence of adjacent simplices generated by the algorithm can be linked together to a piecewise linear (p.l.) path from v to an approximate solution. For an $\bar{x}$ on this path holds that for some $T \subset I_{n+1}$ $\bar{x}$ lies in $A(T)$ and satisfies according to (2.2)

$$\bar{z}_h(\bar{x}) = \max_j \bar{z}_j(\bar{x}) \quad \text{for all } h \in T.$$

Since $A(T)$ is the set of points x in S^n for which the components x_k, $k \notin T$, are (relatively to the point v) equal to each other and are relatively smaller than the other components of x, the p.l. path followed by the algorithm can be interpreted as follows. Initially the component v_h of v having maximal z-value is increased and all other components of v are proportionally decreased. In general, points $\bar{x}$ are generated for which the components $\bar{x}_k$ not having maximal z-value are (relatively to v) equal to each other and are relatively smaller than the components $\bar{x}_h$ of $\bar{x}$ for which $\bar{z}_h(\bar{x}) = \max_j \bar{z}_j(\bar{x})$. As soon as (on the path) $\bar{z}_k(\bar{x})$ becomes equal to $\max_j \bar{z}_j(\bar{x})$, $z_k(x)$ is kept equal to $\max_j \bar{z}_j(x)$ while x_k is relatively increased away from the components x_j of x not having maximal $\bar{z}$-value. If, however, $\bar{x}_h$ for which $\bar{z}_h(\bar{x}) = \max_j \bar{z}_j(\bar{x})$ becomes (relatively to v) equal to the $\bar{x}_j$'s not having maximal $\bar{z}$-value, x_h is kept relatively equal to these x_j's and $\bar{z}_h(x)$ is decreased away from $\max_j \bar{z}_j(x)$. An approximate solution x^* has been found when $\bar{z}_k(x^*) < \max_j \bar{z}_j(x^*)$

implies $x_k^* = 0$ for all $k \in I_{n+1}$. Such a point can therefore be considered as an approximate solution. If the accuracy of the approximate solution is not good enough, the algorithm can be restarted in x^* with a smaller grid size in the hope that within a few iterations a better approximation is found. This process of restarting can be continued until the desired accuracy has been reached. We remark that if the Q-triangulation of S^n underlies the algorithm a restart has to be made in a grid point of this triangulation. Typically, an approximate solution is not a grid point and must be rounded off for example to the nearest grid point. For a further comparison with the algorithm of [7] we refer to the end of Section 3.

3. The V-triangulation of the product space S of unit simplices and the path of points followed by the product-ray algorithm

The problem to be considered in the rest of the paper is the Non-Linear Complementarity Problem on the product space S of N unit simplices (NLCP on S). More precisely, let $S^{n_1}, \ldots S^{n_N}$ be N unit simplices, then S is equal to $\prod_{j=1}^{N} S^{n_j}$. Furthermore, let z be a continuous function from S into R^{N+M}, where $M = \sum_{j=1}^{N} n_j$, satisfying $x_j^T z_j(x) = 0$, $j = 1, \ldots, N$, for all $x = (x_1, \ldots, x_N)$ in S, with $x_j \in S^{n_j}$, $j \in I_N$. The NLCP on S consists of finding a point x^* for which $z(x^*) = (z_1(x^*), \ldots, z_N(x^*)) \leqq 0$. Again this problem is equivalent to the Brouwer fixed point problem on S. To solve the latter problem, in van der Laan and Talman [5] simplicial variable dimensional restart algorithms with $\sum_{j=1}^{N} (n_j + 1)$ rays to leave the arbitrary starting point v were introduced based on the Q-triangulation as well as the U-triangulation of S. However both triangulations have the same disadvantages as on S^n. Therefore we will generalize the V-triangulation introduced in Section 2 to a triangulation of S. This V-triangulation of S will give rise to a new simplicial variable dimension restart algorithm on S with $\prod_{j=1}^{N} (n_j + 1)$ one-dimensional sets (rays) to leave the arbitrary starting point v. More precisely, there is exactly one ray to each vertex of S and the algorithm leaves v along one of these rays depending on which component of $z_j(v)$, $j = 1, \ldots, N$, is maximal. Initially the corresponding components of v are increased whereas all the other components of v are proportionally decreased. Note the similarity with the algorithm described in section 2. The algorithm which we call the $\prod_{j=1}^{N} (n_j + 1)$-ray or simply the product-ray algorithm is described in Section 4 while this section will describe in detail the V-triangulation of S which underlies the algorithm. To do so we need some more notation. The $(\sum_{i=1}^{j-1} (n_i + 1) + k)$-th component of a point x in S or the k-th component of a vector x_j in S^{n_j} will be denoted by $x_{j,k}$. Moreover, the index $\sum_{i=1}^{j-1} (n_i + 1) + k$ will be denoted by (j, k) for simplicity. For $j = 1, \ldots, N$ the index set $I(j)$ is the set $\{(j, 1), \ldots, (j, n_j + 1)\}$, and I is the union of $I(j)$ over all j in I_N. A vertex of S is denoted by $e(I^0)$ where $I_j^0 = I(j) \cap I^0$ consists of exactly one element of $I(j)$, $j \in I_N$, and is defined by

$$e_{j,h}(I^0) = \begin{cases} 1 & \text{if } (j, h) \in I^0, \\ 0 & \text{otherwise.} \end{cases}$$

For such an index set I^0 the one-dimensional set $A(I^0)$ is defined by $A(I^0) = \text{co}(\{v, e(I^0)\})$ unless v equals $e(I^0)$. The sets $A(I^0)$ are the rays connecting v and all the vertices of S along which the product-ray algorithm can leave v. Notice that the number of rays is indeed equal to $\prod_{j=1}^{N}(n_j+1)$ unless v equals a vertex of S in which case the number of rays is one less. The algorithm will leave v along the ray from v to the vertex $e(T^0)$ of S if $z_{j,k}(v) = \max_{(j,h)\in I(j)} z_{j,h}(v)$ for all $j \in I_N$, where $T_j^0 = \{(j, k)\}$. In general the algorithm to be described in section 4 will approximately follow points x in S such that for some subset T of I with $T_j = T \cap I(j) \neq \emptyset$ for all $j \in I_N$ holds that both $z_{j,h}(x) = \max_k z_{j,k}(x)$ for all $(j, h) \in T$, $j \in I_N$, and x lies in $A(T)$ being the convex hull of v and the vertices $e(T^0)$ of S for which T^0 is a subset of T.

Definition 3.1. For $T \subsetneq I$, with $|T_j| \geq 1$, $j \in I_N$, the set $A(T)$ is equal to

$$A(T) = \text{conv}(\{v\}, \{e(T^0) | T^0 \subset T, |T_j^0| = 1, j \in I_N\}).$$

For the case $S = S^1 \times S^1$ some sets $A(T)$ are illustrated in Fig. 3.1. When v lies in the convex hull of vertices $e(T^0)$, $T^0 \subset T$, we define $A(T)$ to be empty. The dimension of a nonempty set $A(T)$ is equal to $t = |T| - N + 1$. To describe the V-triangulation of S we first subdivide each nonempty $A(T)$ in subsets $A(T^0, \gamma(T^1))$ where T^0 and T^1 partition T such that $|T_j^0| = 1$ for all $j \in I_N$ and where $\gamma(T^1)$ is a permutation vector $(\gamma_1(T^1), \ldots, \gamma_N(T^1))$ of the set T^1 with $\gamma_j(T^1) = ((j, k_1^j), \ldots, (j, k_{t(j)}^j))$ a permutation of the $t(j)$ elements of the set T_j^1. For simplicity,

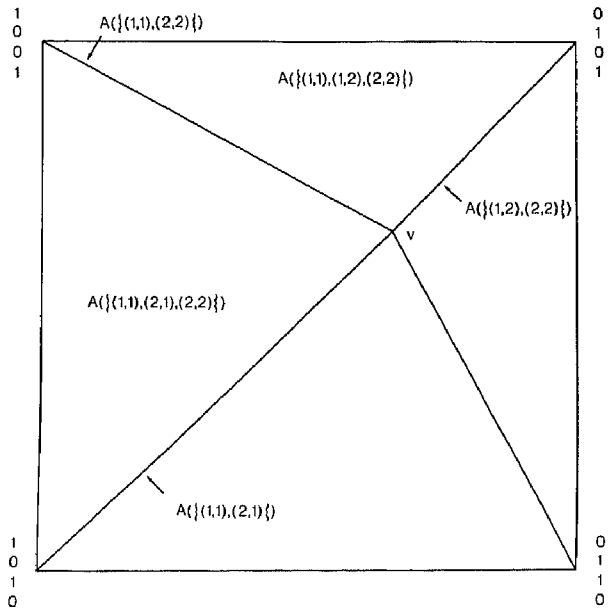

Fig. 3.1. The sets $A(T)$, $|T_j| \geq 1$, of $S = S^1 \times S^1$.

the unique element of T_j^0 is denoted by (j, k_0^j). The subsets $A(T^0, \gamma(T^1))$ are determined by first defining projections of (the starting point) v on the boundary faces of S.

For $K \subset I$ the projection $p(K)$ of v on the boundary set $S(K) = \{x \in S \mid x_{j,k} = 0, (j,k) \notin K\}$ is defined as follows. For $j = 1, \ldots, N$,

$$p_j(K) = v_j \quad \text{if } K_j = \emptyset,$$

and otherwise, if $\sum_{(j,k)\in K_j} v_{j,k} < 1$,

$$p_{j,h}(K) = \begin{cases} \dfrac{1+|K_j^0|}{\sum_{(j,k)\in K_j} v_{j,k} + |K_j^0|}\, v_{j,h}, & (j,h) \in K_j \setminus K_j^0, \\ \dfrac{1 - \sum_{(j,k)\in K_j} v_{j,k}}{\sum_{(j,k)\in K_j} v_{j,k} + |K_j^0|}, & (j,h) \in K_j^0, \\ 0, & (j,h) \notin K_j, \end{cases}$$

and if $\sum_{(j,k)\in K_j} v_{j,k} = 1$,

$$p_{j,h}(K) = \begin{cases} \dfrac{v_{j,h}}{1+|K_j^0|}, & (j,h) \in K_j \setminus K_j^0, \\ \dfrac{1}{1+|K_j^0|}, & (j,h) \in K_j^0, \\ 0, & (j,h) \notin K_j, \end{cases}$$

where $K_j = I(j) \cap K$ and $K_j^0 = \{(j,h) \in I(j) \mid v_{j,h} = 0\}$. Other projections are discussed in Section 5.

Definition 3.2. For $T \subset I$, the set $A(T^0, \gamma(T^1))$ is given by

$$A(T^0, \gamma(T^1)) = \left\{ x \in S \mid x = v + \beta q(T^0) + \sum_{(i,h)\in T^1} \alpha(i,h) q(i,h) \right\}$$

for nonnegative numbers β and $\alpha(i,h)$, $(i,h) \in T^1$, such that for each $j \in I_N$,

$$0 \leq \alpha(j, k_{t(j)}^j) \leq \cdots \leq \alpha(j, k_1^j) \leq \beta \leq 1,$$

where the $(N+M)$-vectors $q(T^0)$ and $q(i,h)$, $(i,h) \in T^1$, are given by

$$q(T^0) = e(T^0) - v$$

and for $h = 1, \ldots, t(j)$, $j \in I_N$,

$$q(j, k_h^j) = p(T^0 \cup \{(j,k_1^j), \ldots, (j,k_h^j)\}) - p(T^0 \cup \{(j,k_1^j), \ldots, (j,k_{h-1}^j)\}).$$

Observe that $q_i(j, k_h^j) = 0$ for all $i \neq j$. The set $A(T)$ is the union of $A(T^0, \gamma(T^1))$ over all T^0 and permutation vectors $\gamma(T^1)$ of $T^1 = T \setminus T^0$. Some subsets $A(T^0, \gamma(T^1))$ are illustrated in Fig. 3.2 for the case $N = 2$, $n_1 = 1$ and $n_2 = 1$.

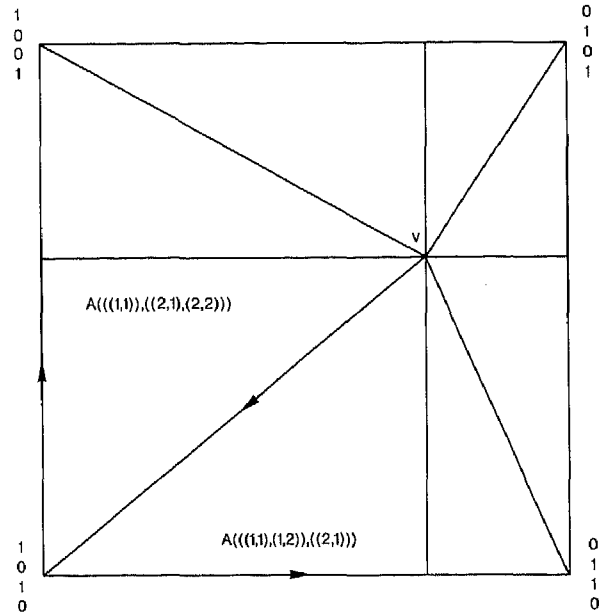

Fig. 3.2. The subsets $A(T^0, \gamma(T^1))$ of $S = S^1 \times S^1$.

The dimension of $A(T^0, \gamma(T^1))$ is less than $t = |T| - N + 1$ if and only if $(j, k) \in T$ implies $v_{j,k} = 0$. The dimension of $A(T^0, \gamma(T^1))$ is independent of T^0 and $\gamma(T^1)$ and is equal to the rank of the matrix $Q(T^0, \gamma(T^1))$ consisting of the t columns $q(T^0)$ and $q(i, h)$, $(i, h) \in T^1$. In the sequel we will restrict ourselves to the sets $A(T^0, \gamma(T^1))$ having dimension t.

The V-triangulation of S is obtained by first triangulating each $A(T^0, \gamma(T^1))$ in t-simplices. Let m^{-1} again be the grid size, with m a positive integer.

Definition 3.3. The set $G(T^0, \gamma(T^1))$ is the collection of t-simplices $\sigma(y^1, \pi(T))$ with vertices $y^1, \ldots, y^{t+1}$ such that

(i) $y^1 = v + bm^{-1}q(T^0) + \sum_{(i,h)\in T^1} a(i, h)m^{-1}q(i, h)$ for nonnegative integers b and $a(i, h)$, $(i, h) \in T^1$, such that for all $j \in I_N$, $0 \leq a(j, k^j_{t(j)}) \leq \cdots \leq a(j, k^j_1) \leq b \leq m-1$;

(ii) $\pi(T) = (\pi_1, \ldots, \pi_t)$ is a permutation of the t elements consisting of T^0 and the $t-1$ elements of T^1 such that: if, for some $j \in I_N$, $a(j, k^j_1) = b$ this implies $p' > p$ where $\pi_{p'} = (j, k^j_1)$ and $\pi_p = T^0$, and if, for some $j \in I_N$, $a(j, k^j_i) = a(j, k^j_{i-1})$ for some i, $2 \leq i \leq t(j)$, this implies $p' > p$ where $\pi_{p'} = (j, k^j_i)$ and $\pi_p = (j, k^j_{i-1})$;

(iii) $y^{i+1} = y^i + m^{-1}q(\pi_i)$, $i = 1, \ldots, t$.

The set $G(T^0, \gamma(T^1))$ is a triangulation of $A(T^0, \gamma(T^1))$ and the union $G(T)$ of $G(T^0, \gamma(T^1))$ over all T^0 and $\gamma(T^1)$ triangulates $A(T)$. Finally, the union G of $G(T)$ over all subsets T of I such that $|T_j| \geq 1$, $j \in I_N$, induces a triangulation of S. We call this triangulation G the V-triangulation of S. For $N = 2$, $n_1 = 1$, $n_2 = 1$ and $m = 2$ the V-triangulation is illustrated in Fig. 3.3.

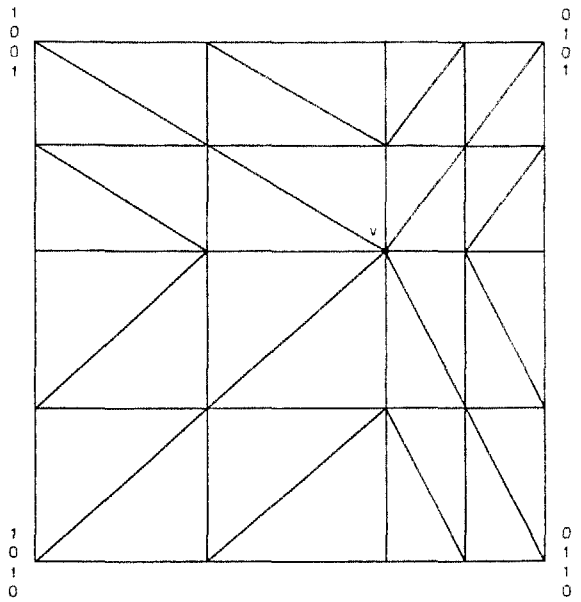

Fig. 3.3. The V-triangulation of $S = S^1 \times S^1$ when $m^{-1} = \frac{1}{2}$.

We remark that the V-triangulation of S is similar to the K'-triangulation of R^n proposed in Todd [11] in case $n_j = 1$ for all $j \in I_N$. The product-ray algorithm generates a path of points $\bar{x}$ in S, starting in v and terminating with an approximate solution, such that for some $T \subset I$, $\bar{x}$ lies in $A(T)$ and

$$\bar{z}_{j,h}(\bar{x}) = \max_{(j,l) \in I(j)} \bar{z}_{j,l}(\bar{x}) \quad \text{for all } (j,h) \in T$$

where $\bar{z}$ is the piecewise linear approximation of z with respect to the V-triangulation with grid size m^{-1}. This path is followed by alternating replacement steps in the triangulation and linear programming pivot steps in a linear system. This system is induced by the equations above. Let $\sigma(y^1, \ldots, y^{g+1})$, $g = t-1$, t, be a g-simplex in $A(T)$ containing $\bar{x}$ so that there are $\lambda_1, \ldots, \lambda_{g+1} \geq 0, \sum_{i=1}^{g+1} \lambda_i = 1$, for which $\bar{x} = \sum_i \lambda_i y^i$, then the equations above are equivalent to

$$\sum_{i=1}^{g+1} \lambda_i z_{j,h}(y^i) = \beta_j \quad \text{if } (j,h) \in T,$$

$$\sum_{i=1}^{g+1} \lambda_i z_{j,h}(y^i) \leq \beta_j \quad \text{if } (j,h) \notin T,$$

where $\beta_j = \max_l \bar{z}_{j,l}(\bar{x})$. We call such a g-simplex T-complete.

Definition 3.4. For $g = t-1$, t, where $t = |T| - N + 1$, a g-simplex $\sigma(y^1, \ldots, y^{g+1})$ is T-complete if the system of linear equations

$$\sum_{i=1}^{g+1} \lambda_i \begin{pmatrix} z(y^i) \\ 1 \end{pmatrix} + \sum_{(j,k) \notin T} \mu_{j,k} \begin{pmatrix} e(j,k) \\ 0 \end{pmatrix} - \sum_{j=1}^{N} \beta_j \begin{pmatrix} \bar{e}(j) \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \tag{3.1}$$

where $e(j,k)$ denotes the $(\sum_{i=1}^{j-1}(n_i+1)+k)$-th unitvector in R^{N+M} and $\bar{e}(j)=\sum_{h=1}^{n_j+1} e(j,h)$, has a solution $\lambda_i^* \geq 0$, $i=1,\ldots,g+1$, $\mu_{j,k}^* \geq 0$, $(j,k) \notin T$, and β_j^*, $j=1,\ldots,N$.

Observe that the system (3.1) has $(g+1)+(N+M-|T|)+N$ columns, so when $g=t-1=|T|-N$, the system has $N+M+1$ columns and for $g=t$ one column more. A solution λ_i^*, $i=1,\ldots,g+1$, $\mu_{j,k}^*$, $(j,k) \notin T$, β_j^*, $j \in I_N$, will be denoted by $(\lambda^*, \mu^*, \beta^*)$.

Nondegeneracy assumption 3.5. For $g=t-1$ the system (3.1) has a unique solution $(\lambda^*, \mu^*, \beta^*)$ with $\lambda_i^* > 0$, $i=1,\ldots,t$, and $\mu_{j,k}^* > 0$, $(j,k) \notin T$, and for $g=t$ at most one variable of (λ^*, μ^*) is equal to zero.

Under this assumption $\sigma(v)$ is T^0-complete with $T_j^0=\{(j,k_0^j)\}$ with (j,k_0^j) the unique index for which $z_{j,k_0^j}(v)=\max_h z_{j,h}(v)$, $j=1,\ldots,N$.

The algorithm now starts with $\sigma(v)$ for $T=T^0$ and follows a sequence of adjacent t-simplices in $A(T)$ for varying T, $T \subset I$, such that their common facets are T-complete. The algorithm terminates as soon as a complete simplex has been generated.

Definition 3.6. A T-complete $(t-1)$-simplex $\sigma(y^1,\ldots,y^t)$ is complete if for all x in σ $x_{i,h}=0$ for all $(i,h) \notin T$.

A complete simplex σ yields an approximate solution $x^*=\sum_{i=1}^t \lambda_i^* y^i$ where $(\lambda^*, \mu^*, \beta^*)$ is the corresponding solution to the linear system (3.1) with respect to σ. At such a point x^* we have from (3.1) that $\bar{z}_{j,k}(x^*)=\beta_j^*$ if $x_{j,k}^* > 0$ and $\bar{z}_{j,k}(x^*) \leq \beta_j^*$ if $x_{j,k}^*=0$.

Lemma 3.7. *Let $\varepsilon > 0$ be such that $\max_{(i,h)}|z_{i,h}(x)-z_{i,h}(y)| < \varepsilon$ for all x and y in the same simplex σ of the V-triangulation of S. Let σ^* be a complete simplex and let x^* be as above, then we have*

$$-\varepsilon < \beta_j^* < \varepsilon, \quad j \in I_N$$

$$\beta_j^* - \varepsilon < z_{j,k}(x^*) < \beta_j^* + \varepsilon \quad \text{when } x_{j,k}^* > 0,$$

$$z_{j,k}(x^*) < \beta_j^* + \varepsilon \quad \text{when } x_{j,k}^* = 0.$$

The proof follows from the fact that $x_j^{*T} \bar{z}_j(x^*)=\beta_j^*$ for all $j \in I_N$. The steps of the algorithm and the proof that the algorithm indeed finds a complete simplex within a finite number of steps are given in Section 4.

There it will be shown that the T-complete t-simplices $\sigma(y^1, \pi(T))$ in $A(T)$ form sequences of adjacent simplices having T-complete common facets. Each sequence not being a loop has two end points. An end point of a sequence in $A(T)$ is either a $T \cup \{(j,k)\}$-complete simplex σ or a simplex with a T-complete facet τ in the

boundary of $A(T)$. In the first case τ is either complete or a facet of a unique $(t+1)$-simplex in $A(T \cup \{(j,k)\})$ being an end point of a sequence of adjacent simplices in $A(T \cup \{(j,k)\})$ with $T \cup \{(j,k)\}$-complete common facets. In the second case the facet τ, if not complete, is a $(t-1)$-simplex in $A(T \backslash \{(i,h)\})$ for some $(i,h) \in T$ being therefore an end point of adjacent $(t-1)$-simplices in $A(T \backslash \{(i,h)\})$ with $T \backslash \{(i,h)\}$-complete common facets. The 0-simplex $\sigma(v)$ is an end point of a sequence of adjacent 1-simplices in $A(T^0)$ with T^0-complete common facets. The sequences in $A(T)$ for varying $T \subset I$, $|T_j| \geq 1$ for all $j \in I_N$, can therefore be linked together to form sequences of adjacent simplices of varying dimension. Exactly one sequence connects $\sigma(v)$ with a complete simplex and this sequence is followed by the product-ray algorithm starting in v.

The set of solutions (λ, μ, β) of (3.1) with respect to a T-complete t-simplex $\sigma(y^1, \ldots, y^{t+1})$ in $A(T)$ yields a line segment of points $\bar{x}$ in σ for which $\bar{x} = \Sigma_i \lambda_i y_i$ and

$$\bar{z}_{j,k}(\bar{x}) = \beta_j \quad \text{if } (j,k) \in T,$$

$$\bar{z}_{j,k}(\bar{x}) = \beta_j - \mu_{j,k} \quad \text{if } (j,k) \notin T,$$

where $\mu_{j,k} \geq 0$, $(j,k) \notin T$. An end point x of such a line segment is characterized by $\lambda_p = 0$ for some p, $1 \leq p \leq t+1$, or by $\mu_{j,k} = 0$ for some $(j,k) \notin T$. In the first case x lies in the facet τ of σ opposite the vertex y^p. Then x is an approximate solution if τ is complete, otherwise x is an end point of a line segment either in τ if τ lies in $A(T \backslash \{(i,h)\})$ for some $(i,h) \in T$ or in the unique t-simplex $\bar{\sigma}$ in $A(T)$ also having τ as a facet. If at an end point x $\mu_{j,k} = 0$ for some $(j,k) \notin T$ then x is an approximate solution if σ is complete or an end point of a line segment in the unique $(t+1)$-simplex in $A(T \cup \{(j,k)\})$ having σ as a facet.

The algorithm therefore in fact generates a piecewise linear path of points $\bar{x}$ in $A(T)$ for varying T, starting with $T = T^0$ and leading to an approximate solution, satisfying $\bar{z}_{j,h}(\bar{x}) = \max_l \bar{z}_{j,l}(\bar{x})$ for all $(j,h) \in T$. Notice that for each T, $T \subset I$, the set $A(T)$ is the set of points x in S for which the components $x_{j,k}$, $(j,k) \notin T$, are (relatively to the point v) equal to each other and relatively smaller than the other components of x. The p.l. path of the new algorithm can therefore be interpreted as follows. Initially, for each j the component $v_{j,h}$ of v for which $z_{j,h}(v) = \max_l z_{j,l}(v)$ is increased whereas all the other components of v are (simultaneously) proportionally decreased. In general the product-ray algorithm generates points $\bar{x}$ in S for which the components $\bar{x}_{j,k}$ not having maximal $\bar{z}_j$-value are (relatively to the starting point v) equal to each other and relatively smaller than the components $\bar{x}_{j,h}$ of $\bar{x}$ having maximal $\bar{z}_j$-value. When $\bar{z}_{j,k}(\bar{x})$ becomes equal to $\max_l \bar{z}_{j,l}(\bar{x})$ for some (j,k), then $\bar{z}_{j,k}(x)$ is kept equal to this maximum and $x_{j,k}$ is (relatively to v) increased away from the components $x_{i,l}$ of x not having maximal $\bar{z}_i$-value. If, however, at $\bar{x}$ on the path a component $\bar{x}_{i,h}$ having maximal $\bar{z}_i$-value becomes relatively equal to the components $\bar{x}_{j,k}$ of $\bar{x}$ not having maximal $\bar{z}_j$-value, $x_{i,h}$ is kept relatively equal to these $\bar{x}_{j,k}$'s and $\bar{z}_{i,h}(x)$ is decreased away from $\max_l \bar{z}_{i,l}(x)$.

Finally, if $\bar{x}_{i,h} = 0$ for all indices $(i, h) \in I$ for which $\bar{z}_{i,h}(\bar{x}) < \max_k \bar{z}_{i,k}(\bar{x})$, then $\bar{x}$ is an approximate solution and the p.l. path terminates. If the accuracy of the approximation is not good enough the algorithm can be restarted in $\bar{x}$ with a smaller grid size to obtain a better approximation.

Since $\bar{z}$ approximates z the p.l. path followed by the algorithm approximates (a path of) points x in S such that for some $T \subset I$ x lies in $A(T)$ and satisfies

$$z_{j,h}(x) = \max_l z_{j,l}(x) \quad \text{for all } (j, h) \in T.$$

An apparent difference between the product-ray algorithm and the algorithm of [7] is that the new algorithm starts by increasing for each $j \in I_N$ the component $v_{j,h}$ of v_j having the largest z_j-value. These increases are combined with a proportional decrease of all the other components of v. The algorithm of [7], however, only increases the component $v_{i,h}$ of the whole $(M+N)$-vector v that has the largest z-value (instead of the largest z_i-value). This initial increase of one component is combined with a decrease with the same amount of exactly one other (arbitrary) component of v_i. In other words, in the new algorithm more information of the z-value is used to adapt the variables and the adaptation of the variables is much more natural.

A second difference of both algorithms concerns the accuracy of the $z_j(x^*)$'s at an approximate solution x^*. As shown in Lemma 3.7 the estimation of the accuracy of the $z_j(x^*)$'s is the same for all j. This result differs from the estimated accuracy of $z(x^*)$ at an approximate solution obtained from the algorithm of [7]. For the latter algorithm only for one $j \in I_N$ $z_j(x^*)$ can be shown to be accurate. More precisely, the algorithm of [7] follows a piecewise linear path of points $\bar{x}$ in $|T|$-dimensional regions $\tilde{A}(T)$ based on the Q-triangulation of S for varying index sets T in I with $|T_j| < n_j + 1$, $j \in I_N$, such that

$$\bar{z}_{j,k}(\bar{x}) = \max_{(i,h) \in I} \bar{z}_{i,h}(\bar{x}) \quad \text{for all } (j, k) \in T.$$

The algorithm starts in v with T consisting of the (unique) index (j, k) for which $z_{j,k}(v) = \max_{(i,h)} z_{i,h}(v)$, and terminates with an approximate solution x^* as soon as for some $j \in I_N$ either $\bar{z}_{j,k}(x^*) = \max \bar{z}_{i,h}(x^*)$ or $x^*_{j,k} = 0$ for all $(j, k) \in I(j)$. So, at x^* there is for each g in I_N, $g \neq j$, at least one index (g, l) in $I(g)$ with

$$x^*_{g,l} > 0 \quad \text{and} \quad \bar{z}_{g,l}(x^*) < \max_{(i,h) \in I} \bar{z}_{i,h}(x^*).$$

This might give a poor accuracy, especially if $x^*_{g,l}$ is rather small. Recall that for the new algorithm we have at an approximate solution x^*

$$\bar{z}_{j,k}(x^*) = \max_h \bar{z}_{j,h}(x^*) \quad \text{for all } (j, k) \text{ with } x^*_{j,k} > 0.$$

Both features, a better accuracy at an approximation solution and a more natural adaptation of the variables, should make the new algorithm more attractive than the algorithm of [7], especially when a good accuracy is required and several restarts have to be made. These theoretical arguments are confirmed by the computational results given in Section 6.

4. The steps of the product-ray algorithm on the product space S of unit simplices

In this section we describe in detail the steps of the product-ray algorithm for finding a complete simplex which approximates a solution to the NLCP on S. Starting with the zero-dimensional simplex $\sigma(v)$ the algorithm generates a sequence of adjacent T-complete t-simplices in $A(T)$ for varying $T \subset I$, $|T_j| \geq 1$, $j \in I_N$, until a complete simplex is generated. Under Assumption 3.5 the T-complete t-simplices $\sigma(y^1, \pi(T))$ in $A(T^0, \gamma(T^1))$ form for given T^0 and $\gamma(T^1)$ sequences of adjacent t-simplices with common T-complete facets. Let $\sigma(y^1, \pi(T))$ and $\bar{\sigma}(\bar{y}^1, \bar{\pi}(T))$ be two adjacent t-simplices in $A(T^0, \gamma(T^1))$, then $\bar{y}^1$, $\bar{\pi}(T)$ and $\bar{\sigma}$ are obtained from y^1, $\pi(T)$ and σ according to Table 1, described in Section 2, where p is the index of the vertex of σ opposite the common facet of σ and $\bar{\sigma}$, $a_{i,h} = a(i,h)$ for $(i,h) \in T^1$, $a_{i,h} = b$ for $(i,h) \in T^0$, and $a_{i,h} = 0$ for $(i,h) \notin T$. Each sequence is either a loop or has two end points. An end point of such a sequence is either a $T \cup \{(j,k)\}$-complete t-simplex for some $(j,k) \notin T$ or a T-complete t-simplex with a T-complete facet in the boundary of $A(T^0, \gamma(T^1))$. If an end point $\sigma(y^1, \pi(T))$ of a sequence in $A(T^0, \gamma(T^1))$ is a $T \cup \{(j,k)\}$-complete t-simplex for some $(j,k) \notin T$, then σ is either complete or a $T \cup \{(j,k)\}$-complete facet of exactly one $(t+1)$-simplex $\bar{\sigma}$ in $A(T \cup \{(j,k)\})$.

Lemma 4.1. *A $T \cup \{(j,k)\}$-complete t-simplex $\sigma(y^1, \pi(T))$ in $A(T^0, \gamma(T^1))$ is complete if and only if $v_{i,h} = 0$ for all $(i,h) \notin T \cup \{(j,k)\}$.*

When σ is not complete, σ is a $T \cup \{(j,k)\}$-complete facet of exactly one $(t+1)$-simplex $\bar{\sigma}$ in $A(T \cup \{(j,k)\})$. More precisely, $\bar{\sigma}$ is the $(t+1)$-simplex $\bar{\sigma}(y^1, \pi(T \cup \{(j,k)\}))$ in $A(T^0, \gamma(T^1 \cup \{(j,k)\}))$ with

$$\gamma_h(T^1 \cup \{(j,k)\}) = \begin{cases} \gamma_h(T^1), & h \neq j, \\ ((j,k_1^j), \ldots, (j,k_{t(j)}^j), (j,k)), & h = j, \end{cases}$$

and

$$\pi(T \cup \{(j,k)\}) = (\pi_1, \ldots, \pi_t, (j,k)).$$

Proof. Suppose that σ is a $T \cup \{(j,k)\}$-complete t-simplex in $A(T^0, \gamma(T^1))$ such that $v_{i,h} = 0$ for all $(i,h) \notin T \cup \{(j,k)\}$. We have to show that σ is complete, i.e. that for all x in σ $x_{i,h} = 0$ holds for all $(i,h) \notin T \cup \{(j,k)\}$. Let x lie in σ, then x also lies in $A(T^0, \gamma(T^1))$, so that there exist nonnegative numbers β and $\alpha(g,l)$, $(g,l) \in T^1$, such that

$$x = v + \beta q(T^0) + \sum_{(g,l) \in T^1} \alpha(g,l) q(g,l) \tag{4.1}$$

with $0 \leq \alpha(g, k_{t(g)}^g) \leq \cdots \leq \alpha(g, k_1^g) \leq \beta \leq 1$ for all $g \in I_N$. Since $q_{i,h}(g,l) = 0$, $(g,l) \in T^1$, and $q_{i,h}(T^0) = -v_{i,h}$ for all $(i,h) \notin T \cup \{(j,k)\}$ we obtain, for all $(i,h) \notin T \cup \{(j,k)\}$,

$$x_{i,h} = v_{i,h}(1-\beta). \tag{4.2}$$

Therefore $v_{i,h}=0$ for all $(i,h)\notin T\cup\{(j,k)\}$ implies $x_{i,h}=0$ for all $(i,h)\notin T\cup\{(j,k)\}$ and x in σ.

Now suppose that σ is complete, i.e., for all x in σ, $x_{i,h}=0$, $(i,h)\notin T\cup\{(j,k)\}$. In particular $y^1_{i,h}=0$ for all $(i,h)\notin T\cup\{(j,k)\}$. For $x=y^1$ equation (4.1) holds with $\beta=b/m<1$, so that, according to equation (4.2), $v_{i,h}=0$ for all $(i,h)\notin T\cup\{(j,k)\}$. This completes the first part of the proof.

Now suppose that σ is not complete. Then there is an index $(g,l)\notin T\cup\{(j,k)\}$ such that $v_{g,l}>0$ and $A(T\cup\{(j,k)\})$ is non-empty. Consequently, there is a unique $(t+1)$-simplex $\bar\sigma$ in $A(T\cup\{(j,k)\})$ having σ as a facet. From the definition of $G(T\cup\{(j,k)\}$ it follows immediately that $\bar\sigma=\sigma(y^1,(\pi(T),(j,k)))$ and lies in $A(T^0,\gamma(T^1\cup\{(j,k)\}))$.

So, if σ is not complete, it is a facet of an end point of a sequence of adjacent $(t+1)$-simplices in $A(T^0,\gamma(T^1\cup\{(j,k)\}))$ with $T\cup\{(j,k)\}$-complete common facets.

If an end point of a sequence of adjacent t-simplices in $A(T^0,\gamma(T^1))$ with T-complete common facets is a t-simplex $\sigma(y^1,\pi(T))$ with a T-complete facet in the boundary of $A(T^0,\gamma(T^1))$ then this facet is either a facet of exactly one other t-simplex $\bar\sigma(y^1,\bar\pi(T))$ in $A(T)$ or it lies in the boundary of $A(T)$. Notice that the boundary of $A(T)$ is equal to

$$\operatorname{bd}A(T)=\bigcup_{\substack{(i,h)\in T\\ |T_i|>1}}A(T\backslash\{(i,h)\})\cup S(T).$$

Lemma 4.2. *A facet τ opposite the vertex y^p of a t-simplex $\sigma(y^1,\pi(T))$ in $A(T^0,\gamma(T^1))$ lies in the boundary of $A(T^0,\gamma(T^1))$ iff*

(a) $p=1$: $\pi_1=T^0$ *and* $b=m-1$;

(b) $1<p<t+1$: $\pi_{p-1}=(j,k^j_{i-1})$, $\pi_p=(j,k^j_i)$ *for certain* $j\in I_N$ *and* i, $2\leqslant i\leqslant t(j)$, *and* $a(\pi_{p-1})=a(\pi_p)$; or $\pi_{p-1}=T^0$, $\pi_p=(j,k^k_1)$ *for certain* $j\in I_N$ *and* $b=a(\pi_p)$;

(c) $p=t+1$: $\pi_t=(j,k^j_{t(j)})$ *for certain* $j\in I_N$ *and* $a(\pi_t)=0$.

The lemma follows immediately from the definitions of $G(T^0,\gamma(T^1))$ and $A(T^0,\gamma(T^1))$.

If the T-complete facet τ opposite vertex y^1 of $\sigma(y^1,\pi(T))$ lies in $\operatorname{bd}A(T^0,\gamma(T^1))$, then τ lies in $S(T)$ since $\pi_1=T^0$ and $b=m-1$, and is therefore complete.

Lemma 4.3. *If, for some p, $1<p<t+1$, the facet τ opposite vertex y^p, of $\sigma(y^1,\pi(T))$ lies in* $\operatorname{bd}A(T^0,\gamma(T^1))$, *then τ is a facet of exactly one other t-simplex in $A(T)$. More precisely, if $i>1$, with i given by Lemma* 4.2b, *$\bar\sigma$ is the t-simplex $\bar\sigma(y^1,\bar\pi(T))$ in $A(T^0,\bar\gamma(T^1))$ where $\bar\gamma(T^1)$ and $\bar\pi(T)$ are given by*

$$\bar\gamma_h(T^1)=\begin{cases}\gamma_h(T^1), & h\neq j,\\ ((j,k^j_1),\ldots,(j,k^j_{i-2}),(j,k^j_i),(j,k^j_{i-1}),\ldots,(j,k^j_{t(j)})), & h=j,\end{cases}$$

and

$$\bar{\pi}(T) = (\pi_1, \ldots, \pi_{p-2}, \pi_p, \pi_{p-1}, \ldots, \pi_t).$$

If $i = 1$, *then* $\bar{\sigma}$ *is the* t*-simplex* $\bar{\sigma}(y^1, \bar{\pi}(T))$ *in* $A(\bar{T}^0, \gamma(\bar{T}^1))$ *where* $\bar{T}^0$, $\gamma(\bar{T}^1)$ *and* $\bar{\pi}(T)$ *are given by*

$$\bar{T}^0_h = \begin{cases} T^0_h, & h \neq j, \\ \{(j, k^j_1)\}, & h = j, \end{cases}$$

$$\gamma_h(\bar{T}^1) = \begin{cases} \gamma_h(T^1), & h \neq j, \\ ((j, k^j_0), (j, k^j_2), \ldots, (j, k^j_{t(j)})), & h = j, \end{cases}$$

and

$$\bar{\pi}(T) = (\pi_1, \ldots, \pi_{p-2}, \bar{T}^0, (j, k^j_0), \pi_{p+1}, \ldots, \pi_t).$$

The proof of this lemma also follows immediately from the definitions of $A(T^0, \gamma(T^1))$ and $G(T^0, \gamma(T^1))$. In the case $i > 1$, $\bar{\sigma}$ is an end point of a sequence of adjacent t-simplices in $A(T^0, \bar{\gamma}(T^1))$ having T-complete common facets whereas in the case $i = 1$, σ is an end point of a sequence of adjacent t-simplices in $A(\bar{T}^0, \gamma(\bar{T}^1))$ having T-complete common facets.

Lemma 4.4. *If the facet* τ *opposite vertex* y^{t+1} *of* $\sigma(y^1, \pi(T))$ *lies in* bd $A(T^0, \gamma(T^1))$ *and* $T^1 \neq \emptyset$, *then* τ *lies in* $A(T \backslash \{(i, h)\})$ *with* $(i, h) = \pi_t$. *More precisely,* τ *is the* $(t-1)$*-simplex* $\bar{\sigma}(y^1, \pi(T \backslash \{(i, h)\}))$ *in* $A(T^0, \gamma(T^1 \backslash \{(i, h)\}))$ *where* γ *and* π *are given by*

$$\gamma_j(T^1 \backslash \{(i, h)\}) = \begin{cases} \gamma_j(T^1), & j \neq i, \\ ((i, k^i_1), \ldots, (i, k^i_{t(i)-1})), & j = i, \end{cases}$$

and

$$\pi(T^1 \backslash \{(i, h)\}) = (\pi_1, \ldots, \pi_{t-1}).$$

The $(t-1)$-simplex $\bar{\sigma}$ is therefore an end point of a sequence of adjacent $(t-1)$-simplices in $A(T^0, \gamma(T^1 \backslash \{(i, h)\}))$ having $T \backslash \{(i, h)\}$-complete common facets. In the special case that $T^1 = \emptyset$ so that the boundary of $A(T^0, \gamma(\emptyset))$ consists of the points v and $e(T^0)$, the facet τ is the 0-dimensional simplex $\sigma(v)$ being T^0-complete with $T^0_j = \{(j, k^j_0)\}$, $j \in I_N$, where (j, k^j_0) is the unique index in $I(j)$ for which

$$z_{k, k^j_0}(v) = \max_{h \in I_{n_j+1}} z_{j,h}(v), \quad j \in I_N.$$

In case $v = e(T^0)$, the 0-simplex $\sigma(v)$ is complete and v solves the NLCP on S. If v is not equal to $e(T^0)$, then $\sigma(v)$ is a T^0-complete facet of exactly one 1-simplex $\bar{\sigma}$ in $A(T^0)$. More precisely, $\bar{\sigma}$ is the 1-simplex $\bar{\sigma}(y^1, \pi(T^0))$ in $A(T^0, \gamma(\emptyset))$ with $y^1 = v$ and $\pi(T^0) = (T^0)$.

Therefore, the T-complete t-simplices in $A(T)$, $T \subset I, |T_j| \geq 1, j \in I_N$, form sequences of adjacent t-simplices with T-complete common facets for varying T. Exactly one sequence connects the 0-dimensional simplex $\sigma(v)$ with a complete simplex whereas all other sequences are either loops or connect two complete simplices. To find a complete simplex, the product-ray algorithm follows the sequence which connects v with a complete simplex by starting in v and making alternating linear programming pivot steps in (3.1) and replacement steps in the triangulation. When, by an l.p. pivot step in (3.1), $\mu_{j,k}$ becomes zero for some $(j,k) \notin T$, the current $\sigma(y^1, \pi(T))$ is either complete or a facet of a $(t+1)$-simplex $\bar{\sigma}$ in $A(T \cup \{(j,k)\})$ as described above. In the latter case the algorithm continues by determining $\bar{\sigma}$ and making an l.p. pivot step in (3.1) with the vector $(z^T(y^{t+2}), 1)^T$ where y^{t+2} is the vertex of $\bar{\sigma}$ opposite σ. If, by an l.p. pivot step, λ_p becomes zero for some p, $1 \leq p \leq t+1$, the facet τ of σ opposite the vertex y^p is T-complete. Then τ is either a facet of exactly one other t-simplex $\bar{\sigma}$ in $A(T)$ or τ lies in the boundary of $A(T)$. In the first case the algorithm continues by determining $\bar{\sigma}$ as described above and making an l.p. pivot step with $(z^T(\bar{y}), 1)^T$ where $\bar{y}$ is the vertex of $\bar{\sigma}$ opposite τ. In the second case τ is either complete or a $(t-1)$-simplex $\bar{\sigma}$ in $A(T \setminus \{\pi_t\})$. If τ is not complete the algorithm continues by determining $\bar{\sigma}$ as described above and by making an l.p. pivot step with the vector $(e^T(\pi_t), 0)^T$.

Since the number of simplices and faces of the triangulation is finite and no simplex is generated more than once, the algorithm generates, starting at v, within a finite number of steps a complete simplex. When the accuracy of the approximate solution induced by the complete simplex is not good enough, the algorithm can be restarted in this point with a finer grid size for the V-triangulation in the hope that the next approximation will be found in a small number of iterations and is more accurate (see also Lemma 3.7).

The steps of the algorithm described above can be summarized as follows.

Step 0: Let (j, k_0^j) be the (unique) index for which $z_{j,k_0^j}(v) = \max_k z_{j,k}(v)$, $j \in I_N$. Set $T^0 = T = \{(1, k_0^1), \ldots, (N, k_0^N)\}$. If $v = e(T^0)$ then $\sigma(v)$ is complete and the algorithm terminates with an exact solution, else set $T^1 = \emptyset$, $t = 1$, $y^1 = v$, $\pi(T) = (T^0)$, $\sigma = \sigma(y^1, \pi(T))$, $\bar{p} = 2$, $b = 0$, $a_{i,h} = 0$, $(i,h) \in I$, $\lambda_1 = 1$, $\mu_{j,k} = z_{j,k_0^j}(v) - z_{j,k}(v)$, $(j,k) \notin T$, and $\beta_j = z_{j,k_0^j}(v)$, $j \in I_N$.

Step 1: Calculate $z(y^{\bar{p}})$. Perform an l.p. pivot step by bringing $(z^T(y^{\bar{p}}), 1)^T$ in the system

$$\sum_{\substack{i=1 \\ i \neq \bar{p}}}^{t+1} \lambda_i \begin{pmatrix} z(y^i) \\ 1 \end{pmatrix} + \sum_{(j,k) \notin T} \mu_{j,k} \begin{pmatrix} e(j,k) \\ 0 \end{pmatrix} - \sum_{j=1}^{N} \beta_j \begin{pmatrix} \bar{e}(j) \\ 0 \end{pmatrix} = \begin{pmatrix} \underline{0} \\ 1 \end{pmatrix}.$$

If, for some $(j,k) \notin T$, $\mu_{j,k}$ becomes equal to zero then go to step 3. Otherwise λ_p becomes equal to zero for some p, $p \neq \bar{p}$.

Step 2: If $p = 1$, $b = m-1$ and $\pi_1 = T^0$ then the facet of σ opposite the vertex y^1 is complete and the algorithm terminates.

If $1<p<t+1$, and if for some $j\in I_N$ either $\pi_{p-1}=(j,k^j_{i-1})$, $\pi_p=(j,k_i)$ and $a(\pi_{p-1})=a(\pi_p)$ for some $i\in\{2,\dots,t(j)\}$ or $\pi_{p-1}=T^0$, $\pi_p=(j,k^j_1)$ and $b=a(\pi_p)$, then σ becomes $\bar\sigma$ and $\gamma(T^1)$, T^0 and T^1 are adapted as given in Lemma 4.3, and return to step 1 with $\bar p$ the index of the new vertex of σ.

If $p=t+1$, $\pi_t=(j,k^j_{t(j)})$ and $a(\pi_t)=0$ then set $t=t-1$, $T^1=T^1\backslash\{(j,k^j_{t(j)})\}$, $(i,h)=(j,k^j_{t(j)})$, $\gamma(T^1)$ becomes $\gamma(T^1\backslash\{(j,k^j_{t(j)})\})$ and σ becomes $\bar\sigma$ as given in Lemma 4.4, and go to step 4.

In all other cases $\sigma(y^1,\pi(T))$ and a are adapted according to Table 1. Return to step 1 with $\bar p$ the index of the new vertex of σ.

Step 3: If $v_{i,h}=0$ for all $(i,h)\notin T\cup\{(j,k)\}$ then σ is a complete simplex and the algorithm terminates. Otherwise set $t=t+1$, $T^1=T^1\cup\{(j,k)\}$, while $\gamma(T^1)$ becomes $\gamma(T^1\cup\{(j,k)\})$ and σ becomes $\bar\sigma$ as given in Lemma 4.1. Return to step 1 with $\bar p$ the index of the new vertex of σ.

Step 4: Perform an l.p. pivot step by bringing $(e^T(i,h),0)^T$ in the system

$$\sum_{i=1}^{t+1}\lambda_i\begin{pmatrix}z(y^i)\\1\end{pmatrix}+\sum_{\substack{(j,k)\notin T\\(j,k)\neq(i,h)}}\mu_{j,k}\begin{pmatrix}e(j,k)\\0\end{pmatrix}-\sum_{j=1}^{N}\beta_j\begin{pmatrix}\bar e(j)\\0\end{pmatrix}=\begin{pmatrix}\underline{0}\\1\end{pmatrix}.$$

If, for some $(j,k)\notin T$, $(j,k)\neq(i,h)$, $\mu_{j,k}$ becomes equal to zero then go to step 3. Otherwise return to step 2 with p the index of the vertex for which λ_p becomes zero.

5. Some variants of the V-triangulation of S and S^n

In Section 3 the subsets $A(T)$, $T\subset I$ and $|T_j|\geq 1$, $j\in I_N$, were defined as the convex hull of the point v and the vertices $e(T^0)$ of S with $T^0\subset T$ and $|T^0_j|=1$, $j=1,\dots,N$. In fact $A(T)$ is the convex hull of $\{v\}$ and the boundary face $S(T)=\{x\in S\,|\,x_{jk}=0$ for all $(j,k)\notin T\}$. To triangulate $A(T)$ this set is subdivided in subsets $A(T^0,\gamma(T^1))$ with $T^0\cap T^1=\emptyset$, $T^0\cup T^1=T$, $|T^0_j|=1$ for all $j\in I_N$, and with $\gamma(T^1)$ a permutation vector of the elements of T^1. Each set $A(T^0,\gamma(T^1))$ is determined by the projections $p(K)$ of the point v on $S(K)$, $K\subset T$, defined by $p_j(K)=v_j$ if $K_j=\emptyset$ and otherwise, if $\sum_{(j,k)\in K_j}v_{j,k}<1$,

$$p_{j,h}(K)=\begin{cases}\dfrac{1+|K^0_j|}{\sum_{(j,k)\in K_j}v_{j,k}+|K^0_j|}\,v_{j,h}, & (j,h)\in K^0_j\backslash K_j,\\[2ex] \dfrac{1-\sum_{(j,k)\in K_j}v_{j,k}}{\sum_{(j,k)\in K_j}v_{j,k}+|K^0_j|}, & (j,h)\in K^0_j,\\[2ex] 0, & (j,h)\notin K_j,\end{cases}\tag{5.1}$$

and, in the case $\sum_{(j,h)\in K_j}v_{j,k}=1$,

$$p_{j,h}(K)=\begin{cases}\dfrac{v_{j,h}}{1+|K_j^0|}, & (j,h)\in K_j^0\backslash K_j,\\ \dfrac{1}{1+|K_j^0|}, & (j,h)\in K_j^0,\\ 0, & (j,h)\notin K_J,\end{cases}$$

where $K_j^0=\{(j,h)\in K_j \mid v_{j,h}=0\}$, $j=1,\ldots,N$. We call this projection the relative projection of v on $S(K)$ since for each j the ratio between $p_{j,h}(K)$ and $v_{j,h}$ is independent of h, $(j,h)\in K_j\backslash K_j^0$, whereas the $p_{j,h}(K)$'s, $(j,h)\in K_j^0$, are all equal to each other. However, one can take any point in $S(K)$ as the projection of v on $S(K)$. Interesting projections different from (5.1) are the following three projections, where again $p_j^i(K)=v_j$ if $K_j=\emptyset$, $i=1,2,3$:

(1) $$p_{j,h}^1(K)=\begin{cases}1/|K_j| & \text{if } (j,h)\in K,\\ 0 & \text{if } (j,h)\notin K,\end{cases}$$

(2) $$p_{j,h}^2(K)=\begin{cases}v_{j,h}+(1-\sum\limits_{K_j} v_{j,k})/|K_j| & \text{if } (j,h)\in K \text{ and } \sum\limits_{(j,k)\in K_j} v_{j,k}<1,\\ 1/|K_j| & \text{if } (j,h)\in K \text{ and } \sum\limits_{(j,k)\in K_j} v_{j,k}=1,\\ 0 & \text{if } (j,h)\notin K,\end{cases}$$

and

(3) $$p_{j,h}^3(K)=\begin{cases}1 & \text{if } (j,h)=\text{lexicomin}(K_j),\\ 0 & \text{otherwise,}\end{cases}$$

where lexicomin(K_j) is equal to (j,k) if k is the first index in the set $\{1,\ldots,n_j+1\}$ with (j,k) in K_j, $j=1,\ldots,N$. The projection $p^1(K)$ projects the point v on the barycenter of $S(K)$ and p^1 is therefore called the barycenter projection. The projection $p^2(K)$ projects v on the point in $S(K)$ which is nearest to v and is called the orthogonal projection. Finally, the projection $p^3(K)$ projects v on a certain vertex $e(K^1)$ of S with $K^1\subset K$. We call p^3 the vertex projection. Similarly to the definition of the $A(T^0,\gamma(T^1))$'s given in Section 3 we can define subsets $A^i(T^0,\gamma(T^1))$ of $A(T)$ based on the projections $p^i(K)$, $i=1,2,3$.

Lemma 5.1. *The dimension of $A^i(T^0,\gamma(T^1))$, $i=1,2$, is less than $t=|T|-N+1$ if and only if $v_{j,k}=0$ for all $(j,k)\notin T$. The dimension of $A^3(T^0,\gamma(T^1))$ is less than $t=|T|-N+1$ if and only if $v_{j,k}=0$ for all $(j,k)\notin T$ or for some $j\in I_N$ there is at least one index i, $1\leqslant i\leqslant t(j)$, such that $k_{i-1}^j<k_i^j$. Moreover, $A(T)=A^3(T^0,\gamma(T^1))$ with T^0 and $\gamma(T^1)$ such that for all j $k_0^j>k_1^j>\cdots>k_{t(j)}^j$.*

In general each projection yields a different triangulattion of the sets $A(T)$ and therefore a different triangulation of S. The triangulation induced by the projection p^i will be called the V^i-triangulation of S, $i=1,2,3$. For $N=1$, $n_1=2$ and $m=2$, the V-, V^1-, V^2-, and the V^3-triangulation of S are illustrated in the Figures 5.1–5.4. Observe that the V^3-triangulation of S^n coincides with the triangulation of the unit simplex proposed in Tuy, Thoai and Muu [12].

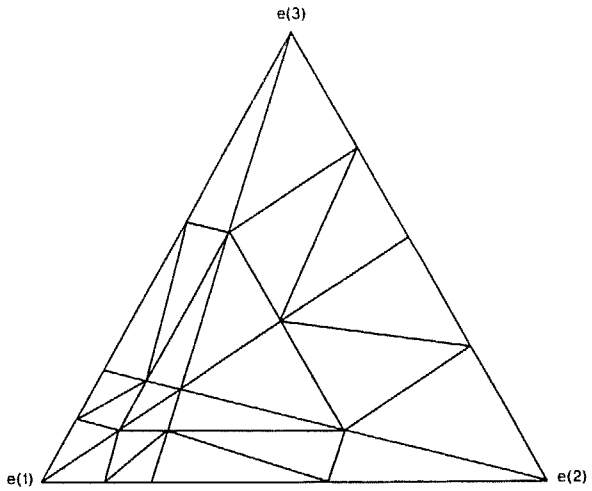

Fig. 5.1. The V-triangulation of S^n, $n=2$, having relative projections and grid size $m^{-1}=\frac{1}{2}$.

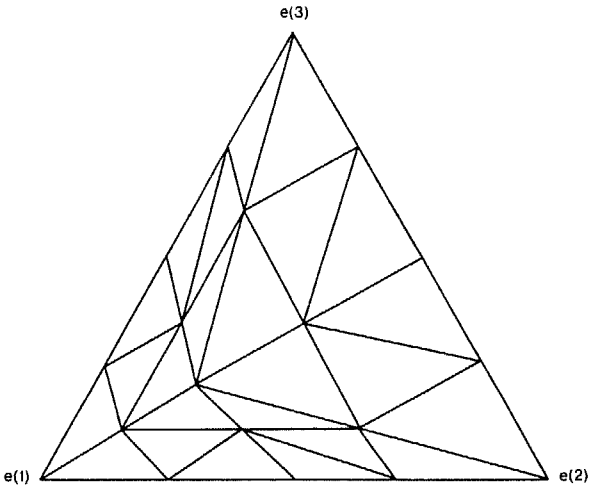

Fig. 5.2. The V^1-triangulation of S^n, $n=2$, having barycentrical projections and grid size $m^{-1}=\frac{1}{2}$.

The relative projections, however, seem to be the most natural ones. It follows from Section 2 that when $S=S^n$ the set $A(T)$, $T\subset I_{n+1}$, is equal to

$$A(T)=\{x\in R^{n+1}\,|\,x=(1-b)v+by \text{ with } y\in S^n(T),\ 0\leqslant b\leqslant 1\},$$

so that

$$A(T)=\left\{x\in S^n\,|\,x=v+\sum_{i\in T}\lambda_i v(i),\ \sum_i\lambda_i\leqslant 1,\ \lambda_i\geqslant 0,\ i\in T\right\}$$

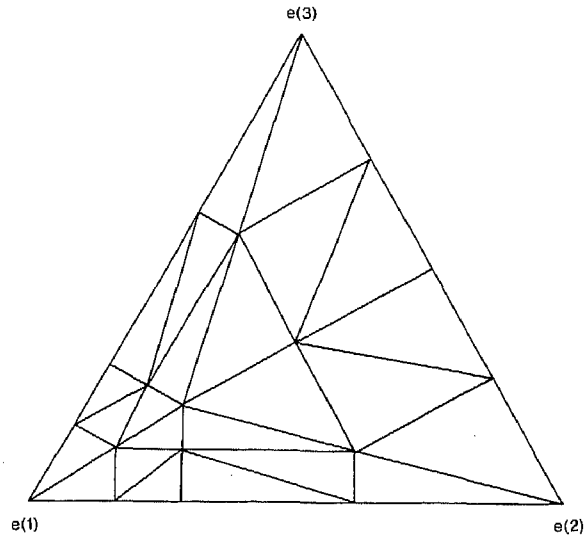

Fig. 5.3. The V^2-triangulation of S^n, $n=2$, having orthogonal projections and grid size $m^{-1}=\frac{1}{2}$.

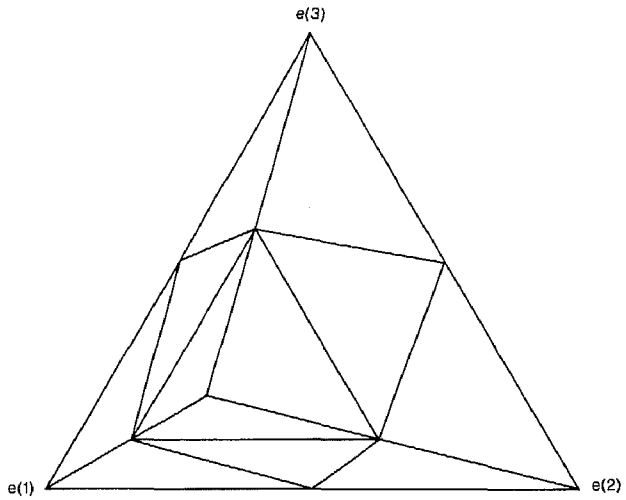

Fig. 5.4. The V^3-triangulation of S^n, $n=2$, having vertex projections and grid size $m^{-1}=\frac{1}{2}$.

with $v(i)=e(i)-v$, $i\in I_{n+1}$. Therefore the regions $A(T)$, $T\subset I_{n+1}$, are similar to the regions $A(T)$ defined in van der Laan and Talman [2] for their basic algorithm on S^n with respect to the matrix Q. This matrix induces the so-called Q-triangulation of S^n. In the same way we could define a triangulation with respect to the matrix $V=[v(1),\ldots,v(n+1)]$. However this triangulation does not triangulate S^n but only its affine hull. When v is equal to the barycenter of S^n, this triangulation coincides with the U-triangulation of the affine hull of S^n proposed in [10] (see also [3]). The disadvantage of these triangulations is that approximate solutions

can only be obtained from full dimensional simplices even if the (real) solution lies on the boundary of S^n.

The V-triangulation presented in Section 2 is however a triangulation of S^n itself so that each $S^n(T)$, $T \subset I_{n+1}$, is also triangulated. This makes it possible to terminate with lower-dimensional simplices on the boundary of S^n. In general this will yield a better approximate solution. Moreover, if the starting point lies on the boundary of S^n fast movements on the boundary are possible. Finally we remark that the steps of the algorithm presented in Section 4, applied to $S = S^n$, are similar to the steps of the basic algorithm on S^n with the Q- or the U-triangulation. The only difference lies in the definition of the regions $A(T)$, $T \subset I_{n+1}$, and the triangulation of each $A(T)$ (see also Section 2). Theoretically, any consistent triangulation of the $A(T)$'s could underly the algorithm. However, triangulations which are easy to implement and which are not based on projections are not known to the authors. The same statements can be made for the general case $N > 1$. If $n_j = 1$ for all j, the V^2-triangulation coincides with the V-triangulation. This case is illustrated in Figure 3.3. The V^1-triangulation and the V^3-triangulation of $S^1 \times S^1$ are illustrated in the Figures 5.5 and 5.6 respectively, for $m^{-1} = \frac{1}{2}$.

6. Computational results

The algorithm presented in Section 4 has been applied to the noncooperative N-person game, to a pure exchange economy, and to an international economy.

We will first give a short description of the noncooperative N-person game. Let $\Gamma = (I_{n_1+1}, \ldots, I_{n_N+1}, a_1, \ldots, a_N)$ be a $2N$-tuple with I_{n_j+1} the index set of strategies of person j and $a_j : I' \to R_+$, where $I' = \prod_{j=1}^{N} I_{n_j+1}$, $j \in I_N$. A vector $s = (s_1, \ldots, s_N)$ in I' denotes for player $j \in I_N$ that he plays his s_j-th pure strategy; $a_j(s)$ is the loss for player j if strategy $s \in I'$ is played. A vector $x = (x_1, \ldots, x_N) \in S$ with $S = \prod_{j=1}^{N} S^{n_j}$ denotes a mixed strategy vector of the game. Let $p_j(x)$ be the expected loss of player j with mixed strategy vector $x \in S$, then we have

$$p_j(x) = \sum_{s \in I'} a_j(s) \prod_{i=1}^{N} x_{i,s_i}, \quad j \in I_N. \tag{6.1}$$

The expected loss $m_{j,k}(x)$ of player j when he plays his k-th pure strategy is given by

$$m_{j,k}(x) = \sum_{\substack{s \in I' \\ s_j = k}} a_j(s) \prod_{\substack{i=1 \\ i \neq j}}^{N} x_{i,s_i}, \quad k \in I_{n_j+1}, \; j \in I_N. \tag{6.2}$$

It follows from (6.1) and (6.2) that

$$p_j(x) = \sum_{k=1}^{n_j+1} x_{j,k} m_{j,k}(x). \tag{6.3}$$

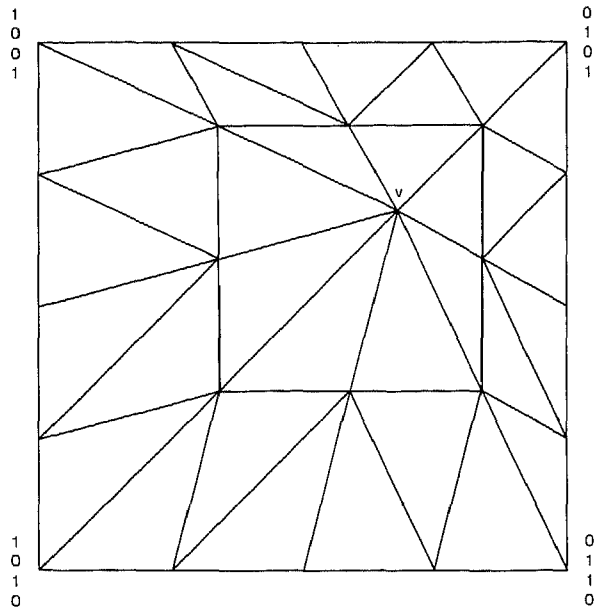

Fig. 5.5. The V^1-triangulation of $S^1 \times S^1$ having barycentrical projections and grid size $m^{-1} = \frac{1}{2}$.

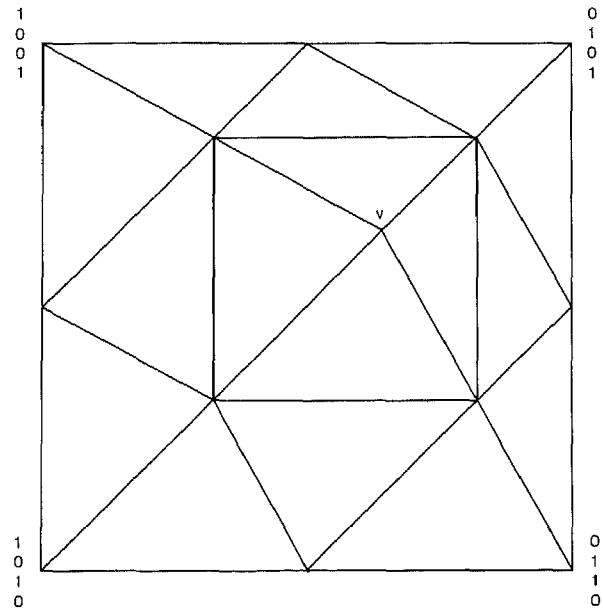

Fig. 5.6. The V^3-triangulation of $S^1 \times S^1$ having vertex projections and grid size $m^{-1} = \frac{1}{2}$.

A mixed strategy vector $x^* \in S$ is a Nash equilibrium of Γ if $p_j(x^*) - m_{j,k}(x^*) \leq 0$ for all $k \in I_{n_j+1}$, $j \in I_N$. Let z be the function from S to $\prod_{j=1}^{N} R^{n_j+1}$ defined by

$$z_{j,k}(x) = p_j(x) - m_{j,k}(x), \quad k \in I_{n_j+1}, \ j \in I_N, \tag{6.4}$$

then according to (6.3) we have, for all $x \in S$,

$$x_j^T z_j(x) = 0, \quad j \in I_N.$$

A solution $x^* \in S$ of the NLCP with the function z defined as in (6.4) is a Nash equilibrium of Γ and conversely. The algorithm presented in Section 4 has been applied to three noncooperative N-person games and compared to the algorithm given in [7]. In each application both algorithms are started in the barycenter of S with a gridsize of $m^{-1} = 1$ for the product-ray algorithm and with a gridsize vector $(1/(n_1+1), \ldots, 1/(n_N+1))$ for the algorithm described in [7], the sum-ray algorithm. When a complete simplex is found the grid is refined with a factor of two and the algorithm is restarted in the approximate solution for the V-triangulation and in the grid point closest to the approximate solution for the Q-triangulation. The grid refinement is stopped when the accuracy of the approximate solution is less than 10^{-10}. For each round with the new algorithm the total number of linear programming steps, the total number of function evaluations up to that round and the accuracy of the approximate solution are given in a table. Since for the algorithm in [7] the number of rounds is much larger to obtain the same accuracy, only the results with corresponding accuracy are given.

Throughout this section we will use the following notations,

ν: round number of the algorithm.

LP: accumulated number of linear programming steps.

FE: accumulated number of function evaluations.

E: accuracy of the approximate solution:

$$E^\nu = \max_{j,k} z_{j,k}(x^\nu) \quad \text{with } x^\nu \text{ the approximate solution in round } \nu.$$

Game 1. Three players with each player two strategies, $N = 3$, $n_j = 1$, $j \in I_N$. The elements $a_j(s)$ are given in Table 2.

Table 2

The elements $a_j(s)$, $s \in I'$, $j \in I_N$. The number in the (j, k)-th row and the (i_{k_1}, i_{k_2})-th column denotes the loss for player j when he plays his k-th pure strategy and for $h = 1, 2$ player k_h plays his i_{k_h}-th pure strategy with $k_1, k_2 \neq j$ and $k_1 < k_2$

Game 1	(1, 1)	(1, 2)	(2, 1)	(2, 2)
(1, 1)	1	2	8	5
(1, 2)	8	8	2	2
(2, 1)	4	2	2	1
(2, 2)	2	6	1	3
(3, 1)	4	1	4	2
(3, 2)	8	8	2	1

The solution of this game is $x^* = (\frac{1}{5}, \frac{4}{5}; \frac{3}{7}, \frac{4}{7}; \frac{2}{3}, \frac{1}{3})^T$. The computational results are given in Tables 3 and 4.

Table 3.
Results of game 1 for the product-ray algorithm

ν	LP	FE	E
1	12	12	$6 \cdot 10^{-2}$
2	21	20	$3 \cdot 10^{-3}$
3	38	34	$9 \cdot 10^{-5}$
4	46	42	$7 \cdot 10^{-7}$
5	54	50	$5 \cdot 10^{-9}$
6	70	64	$<10^{-10}$

Table 4
Results of game 1 for the sum-ray algorithm of [7]

ν	LP	FE	E
3	22	24	$3 \cdot 10^{-2}$
4	34	35	$3 \cdot 10^{-3}$
7	66	69	$2 \cdot 10^{-4}$
11	125	126	$8 \cdot 10^{-7}$
14	163	162	$9 \cdot 10^{-9}$
18	206	205	$<10^{-10}$

Game 2. Three players with each player three strategies, $N = 3$, $n_j = 2, j \in I_N$. The elements $a_j(s)$ are given in Table 5 (see van der Laan and Talman [5]).

The game has as solution $x^* = (\frac{3}{7}, \frac{4}{7}, 0; 0, 1, 0; 0, \frac{2}{3}, \frac{1}{3})^T$. The results are given in Table 6 and 7.

Table 5
The elements $a_j(s)$, $s \in I'$, $j \in I_N$ (see Table 2)

Game 2	(1, 1)	(1, 2)	(1, 3)	(2, 1)	(2, 2)	(2, 3)	(3, 1)	(3, 2)	(3, 3)
(1, 1)	2	3	4	2	3	3	4	1	5
(1, 2)	1	1	4	3	4	1	6	8	2
(1, 3)	4	7	2	4	5	5	3	6	4
(2, 1)	5	6	7	4	8	9	3	5	1
(2, 2)	1	1	3	3	2	1	2	2	4
(2, 3)	2	3	6	5	3	6	7	5	8
(3, 1)	1	3	5	1	6	2	1	2	4
(3, 2)	2	6	5	3	3	7	8	5	5
(3, 3)	5	2	2	4	6	5	8	1	3

Table 6

Results of game 2 for the product-ray algorithm

ν	LP	FE	E
1	14	15	$<10^{-10}$

Table 7

Results of game 2 for the sum-ray algorithm

ν	LP	FE	E
1	22	21	$4 \cdot 10^{-10}$
2	29	29	$1 \cdot 10^{-10}$
3	33	34	$<10^{-10}$

Game 3. Four players with each player two strategies, $N=4$, $n_j=1$, $j \in I_N$. The elements $a_j(s)$ are given in Table 8.

The solution found for this game is $x^* = (\frac{1}{5}, \frac{4}{5}; 1, 0; 1, 0; \frac{2}{3}, \frac{1}{3})^T$.

Table 8

The elements $a_j(s)$, $s \in I'$, $j \in I_N$. The number in the (j, k)-th row and the $(i_{k_1}, i_{k_2}, i_{k_3})$-th column denotes the loss for player j when he plays his k-th pure strategy and for $h=1, 2, 3$ player k_h plays his i_{k_h}-th pure strategy with $k_1, k_2, k_3 \neq j$ and $k_1 < k_2 < k_3$

Game 3	(1, 1, 1)	(1, 1, 2)	(1, 2, 1)	(1, 2, 2)	(2, 1, 1)	(2, 1, 2)	(2, 2, 1)a	(2, 2, 2)
(1, 1)	3	3	4	2	3	3	4	1
(1, 2)	4	1	4	3	1	1	6	8
(2, 1)	4	6	2	4	5	3	3	6
(2, 2)	5	2	7	4	8	6	3	5
(3, 1)	1	6	3	3	3	3	1	2
(3, 2)	2	2	6	5	4	6	3	5
(4, 1)	6	3	5	1	3	2	3	2
(4, 2)	2	6	5	3	4	7	1	5

The results are given in Tables 9 and 10.

The second problem concerns a pure exchange economy. Suppose that $n+1$ is the number of commodities and H the total number of economic agents. Let $p \in R^{n+1} \setminus \{0\}$, then the j-th agent is assumed to respond to this vector of prices by a vector of excess demands

$$(z_1^j(p), \ldots, z_{n+1}^j(p)), \quad j=1, \ldots, H.$$

Table 9

Results game 3 for the sum-ray algorithm

ν	LP	FE	E
1	9	11	$2 \cdot 10^{-1}$
2	24	25	$<10^{-10}$

Table 10
Results game 3 for the sum-ray algorithm

ν	LP	FE	E
1	5	6	$4 \cdot 10^{-1}$
18	117	127	$<10^{-10}$

We assume that each z_i^j, $i=1,\ldots,n+1$, $j=1,\ldots,H$, is homogeneous of degree zero; this permits us to restrict ourselves to prices on the n-dimensional unit simplex. For each individual we have the budget constraint

$$p^{\mathrm{T}} z^j(p) = 0, \quad j=1,\ldots,H,$$

and furthermore we assume that each excess demand function is continuous on the simplex S^n. It is quite easy to show that an economy model satisfying the assumptions mentioned above has a price vector p^* such that

$$z_i(p^*) = \sum_{j=1}^{H} z_i^j(p^*) \leqslant 0, \quad i=1,\ldots,n+1.$$

Such a price vector is called an equilibrium price vector. An equilibrium price vector p^* is a solution of the NLCP on S^n with z the excess demand function and conversely.

We will consider the following exchange model. The j-th agent has an initial endowment $w_{j,i} > 0$ of commodity i, $i=1,\ldots,n+1$, and has a utility function given by

$$u_j(x) = \left(\sum_{i=1}^{n+1} a_{j,i}^{(1-a_j)} x_i^{a_j} \right)^{1/a_j}, \quad x \in R^{n+1},$$

with $a_{j,i} > 0$, $j=1,\ldots,H$, $i=1,,\ldots,n+1$, and $a_j < 1$, $j=1,\ldots,H$. This yields the following excess demand functions

$$z_i^j(p) = \frac{a_{j,i} \sum_{k=1}^{n+1} w_{j,k} p_k}{p_i^{b_j} \sum_{k=1}^{n+1} a_{j,k} p_k^{1-b_j}} - w_{j,i}, \quad j=1,\ldots,H, \ i=1,\ldots,n+1,$$

with $b_j = (1-a_j)^{-1}$, $j=1,\ldots,H$. Note that b_j is strictly positive for all $j \in I_H$.

We have applied the algorithm presented in Section 2 to the three pure exchange economies given in Scarf [9] and again compared it to the algorithm described in [7] with the Q-triangulation. Both algorithms were run in exactly the same way as for the previous example and with terminating accuracy 10^{-9}. The data of the three economies can be found in Scarf [9].

Economy 1: 5 commodities and 3 consumers, i.e. $n=4$ and $H=3$.

The results are given in Tables 11 and 12.

Economy 2: 8 commodities and 5 consumers, i.e. $n=7$ and $H=5$.

The results are given in Tables 13 and 14.

Economy 3: 10 commodities and 5 consumers, i.e. $n=9$ and $H=5$. The results are given in Tables 15 and 16.

Table 11

Results of economy 1 with the V-triangulation of S^n

ν	LP	FE	E
1	4	6	$2 \cdot 10^{1}$
2	9	12	$3 \cdot 10^{0}$
3	20	24	$2 \cdot 10^{-1}$
4	25	30	$9 \cdot 10^{-3}$
5	29	35	$3 \cdot 10^{-4}$
6	33	40	$7 \cdot 10^{-6}$
7	37	45	$7 \cdot 10^{-8}$
8	41	50	$<10^{-9}$

Table 12

Results of economy 1 with the Q-triangulation of S^n

ν	LP	FE	E
1	5	6	$2 \cdot 10^{1}$
2	19	21	$3 \cdot 10^{0}$
4	40	44	$2 \cdot 10^{-1}$
6	56	62	$1 \cdot 10^{-2}$
9	77	86	$2 \cdot 10^{-4}$
11	91	102	$7 \cdot 10^{-6}$
14	115	128	$5 \cdot 10^{-8}$
18	138	155	$<10^{-9}$

Table 13

Results of economy 2 with the V-triangulation of S^n

ν	LP	FE	E
1	7	9	$9 \cdot 10^{0}$
2	16	19	$2 \cdot 10^{0}$
3	36	40	$9 \cdot 10^{-2}$
4	45	50	$3 \cdot 10^{-3}$
5	52	58	$4 \cdot 10^{-4}$
6	59	66	$1 \cdot 10^{-5}$
7	67	75	$2 \cdot 10^{-7}$
8	76	85	$1 \cdot 10^{-9}$
9	86	96	$<10^{-10}$

We will now give a short description of an economy with a block diagonal pattern mentioned earlier by Mansur and Whalley [8] and more recently by van der Laan [1]. The notation has been changed slightly to make it applicable on the product space $S=\prod_{j=1}^{N} S^{n_j}$ of unit simplices.

Table 14
Results of economy 2 with the Q-triangulation of S^n

ν	LP	FE	E
1	8	9	$9 \cdot 10^{0}$
2	36	38	$1 \cdot 10^{0}$
3	53	56	$1 \cdot 10^{-1}$
6	101	107	$5 \cdot 10^{-3}$
8	130	138	$3 \cdot 10^{-4}$
10	160	170	$1 \cdot 10^{-5}$
13	203	216	$2 \cdot 10^{-7}$
17	261	278	$1 \cdot 10^{-9}$
18	279	297	$<10^{-9}$

Table 15
Results of economy 3 with the V-triangulation of S^n

ν	LP	FE	E
1	9	11	$2 \cdot 10^{1}$
2	21	24	$4 \cdot 10^{0}$
3	36	40	$6 \cdot 10^{-1}$
4	49	54	$1 \cdot 10^{-1}$
5	65	71	$6 \cdot 10^{-3}$
6	83	90	$2 \cdot 10^{-4}$
7	97	105	$4 \cdot 10^{-6}$
8	114	123	$2 \cdot 10^{-8}$
9	125	135	$1 \cdot 10^{-9}$
10	138	149	$<10^{-9}$

Table 16
Results of economy 3 with the Q-triangulation of S^n

ν	LP	FE	E
1	10	11	$1 \cdot 10^{1}$
2	27	29	$1 \cdot 10^{0}$
3	44	47	$4 \cdot 10^{-1}$
4	63	67	$2 \cdot 10^{-1}$
7	118	125	$3 \cdot 10^{-3}$
9	167	176	$1 \cdot 10^{-4}$
12	242	254	$4 \cdot 10^{-6}$
15	317	332	$5 \cdot 10^{-8}$
17	350	367	$2 \cdot 10^{-9}$
18	373	391	$<10^{-9}$

There are $N-1$ countries with $N-1$ subsets of consumers (agents) denoted by $A_1, \ldots, A_{N-1}$, with $|A_j| < \infty$, $j = 1, \ldots, N-1$. There are $N-1$ groups G_j, $j \in I_{N-1}$, of non-common goods and there is one group G_N of common goods. The non-common goods in group G_j are owned by the agents in A_j and are traded only among the agents in A_j, $j = 1, \ldots, N-1$. We may think of an international trade model with $N-1$ countries, in which case the commodities in group G_j are the domestic goods of country j, $j = 1, \ldots, N-1$, and the commodities of group G_N are traded internationally.

Suppose there are n_j non-common goods in G_j, indexed by $(j, 1), \ldots, (j, n_j)$, $j \in I_{N-1}$, and $(n_N + 1)$ common goods indexed by $(N, 1), \ldots, (N, n_N + 1)$. So the total number of goods is $M+1$ with $M = \sum_{j=1}^{N} n_j$. Now for any $p \in R_+^{M+1} \setminus \{0\}$, let $z^j(p)$ be the total excess demand of the agents in A_j, given the price vector p. In the following we write $p = (p_1, \ldots, p_{N-1}, p_N)$. Then

$$z_{i,k}^j(p) = 0 \quad \text{for } k = 1, \ldots, n_i \text{ and } i \neq j, N$$

and $z^j(p) = z^j(p_j, p_N)$ for all $j = 1, \ldots, N-1$. Now the equilibrium problem (EP) is to find an equilibrium price vector, i.e. a vector $p^* = (p_1^*, \ldots, p_{N-1}^*, p_N^*) \in \bar{R}^{M+1}$, where $\bar{R}^{M+1}$ is the subset of R_+^{M+1} such that for all $p \in \bar{R}^{M+1}$, (p_j, p_N) has at least one positive element for each $j = 1, \ldots, N-1$, such that

$$\text{(a)} \quad z_j^j(p^*) = z_j^j(p_j^*, p_N^*) = 0, \quad j = 1, \ldots, N-1,$$

and

$$\text{(b)} \quad \sum_{j=1}^{N-1} z_N^j(p^*) = \sum_{j=1}^{N-1} z_N^j(p_j^*, p_N^*) = 0.$$

Under certain assumptions, see e.g. van der Laan [1], we can prove the existence of such an equilbrium price vector. Note that for all $j \in I_{N-1}$, $p_j^{\mathrm{T}} z_j^j(p_j, p_N) + p_N^{\mathrm{T}} z_N^j(p_j, p_N) = 0$, $p \in \bar{R}^{M+1}$. We will use the structure of (EP) to formulate the problem on the product space of unit simplices $S = \prod_{j=1}^{N} S^{n_j}$. Let $x = (x_1, \ldots, x_N)$ be a vector in S then we define price vectors $y^j = (y_j^j, y_N^j) \in R^{n_j} \times R^{N_N+1}$, $j = 1, \ldots, N-1$, by

$$y_{j,k}^j = x_{j,k}, \quad k = 1, \ldots, n_j,$$

and

$$y_{N,k}^j = x_{j,n_j+1} x_{N,k}, \quad k = 1, \ldots, n_N + 1.$$

Now we define $\tilde{z}(x) = (\tilde{z}_1(x), \ldots, \tilde{z}_N(x)) \in \prod_{j=1}^{N} R^{n_j+1}$ by

$$\tilde{z}_{j,k}(x) = \begin{cases} z_{j,k}^j(y^j), & k = 1, \ldots, n_j, \\ \sum_{h=1}^{n_N+1} x_{N,h} z_{N,h}^j(y^j), & k = n_j + 1, \end{cases} \qquad j = 1, \ldots, N-1$$

and

$$\tilde{z}_{N,k}(x) = \alpha \sum_{j=1}^{N-1} z^j_{N,k}(y^j), \quad k = 1, \ldots, n_N + 1,$$

for some $0 < \alpha < (N-1)^{-1}$.

Clearly, $\tilde{z}(x^*) = 0$ if and only if $(\bar{x}^*_1, \ldots, \bar{x}^*_{N-1}, x^*_N)$ in $\prod_{j=1}^{N-1} R^{n_j} \times S^{n_N}$ is an equilibrium price vector, where $\bar{x}^*_{j,k} = x^*_{j,k} / x^*_{j,n_j+1}$, $k = 1, .., n_j$, $j = 1, \ldots, N-1$.

We will apply the algorithm in Section 3 to the same problems given in van der Laan [1] and again compare the results with the algorithm described in [7] for the Q-triangulation of S.

Both algorithms were run in exactly the same way as for the noncooperative N-person game problem except that the initial gridsize of the V-triangulation is set to $m^{-1} = \frac{1}{2}$, the initial gridsize of the Q-triangulation is set to $(1/(n_1+1), \ldots, 1/(n_N+1))/2$, and the terminating accuracy is 10^{-7}. The computational results are given in Table 17, where the first column denotes the number of common goods (n_N+1) and the second column the number of countries. Each country has two non-common goods. The third and fourth column give the accumulated numbers of l.p. steps and function evaluations for the new algorithm with the V-triangulation, while the last two columns give the same numbers for the algorithm described in [7] with the Q-triangulation.

Table 17

Results international trade model for the product-ray and the sum-ray algorithm

Number of commong goods	Number of countries	Product-ray alg.		Sum-ray alg.	
		FE	LP	FE	LP
2	2	54	47	130	117
	3	85	78	196	184
	4	97	90	284	271
	5	128	121	340	328
3	2	56	49	170	158
	3	87	80	258	245
	4	95	88	296	285
	5	109	102	414	403
4	2	67	60	242	229
	3	107	100	315	302
	4	118	111	378	365
	5	145	138	471	458
5	2	79	72	286	273
	3	99	92	415	402
	4	145	138	406	393
	5	182	175	587	574
6	2	89	82	426	412
	3	144	136	458	445
	4	195	188	580	569
	5	221	214	784	773

Concluding, the computational results suggest that the number of function evaluations and linear programming steps for the new algoritm are substantially lower than for the algorithm described in [7] although the results for the latter algorithm can be improved by using the so-called U-triangulation of the affine hull of S^n or S in case of the two economic problems. Then also the approximate solution can be taken as the starting point of the next round. Even then the number of function evaluations is larger when compared with the new algorithm (see Talman [10, p. 84]).

Acknowledgement

The authors would like to thank Gerard van der Laan for his helpful comments.

References

[1] G. van der Laan, "The computation of general equilibrium in economies with a block diagonal pattern", *Econometrica* 53 (1985) 659-666.

[2] G. van der Laan and A.J.J. Talman, "A restart algorithm for computing fixed points without an extra dimension," *Mathematical Programming* 17 (1979) 74-84.

[3] G. van der Laan and A.J.J. Talman, "An improvement of fixed point algorithms by using a good triangulation," *Mathematical Programming* 18 (1980) 274-285.

[4] G. van der Laan and A.J.J. Talman, "A class of simplicial restart fixed point algorithms without an extra dimension," *Mathematical Programming* 20 (1981) 33-48.

[5] G. van der Laan and A.J.J. Talman, "On the computation of fixed points in the product space of unit simplices and an application to noncooperative N-person games," *Mathematics of Operations Research* 7 (1982) 1-13.

[6] G. van der Laan and A.J.J. Talman, "Simplicial approximation of solutions to the nonlinear complementarity problem with lower and upper bounds," Research Memorandum 137, Department of Economics, Tilburg University, Tilburg, The Netherlands (December 1983), to appear in *Mathematical Programming*.

[7] G. van der Laan, A.J.J. Talman and L. Van der Heyden, "Variable dimensions algorithms for unproper labellings," Research Memorandum 147, Department of Economics, Tilburg University, Tilburg, The Netherlands (April 1984), to appear in *Mathematics of Operations Research*.

[8] A. Mansur and J. Whalley, "A decomposition algorithm for general equilibrium computation with application to international trade models," *Econometrica* 50 (1982) 1547-1557.

[9] H. Scarf, "The approximation of fixed points of a continuous mapping," *SIAM Journal of Applied Mathematics* 15 (1967) 1328-1343.

[10] A.J.J. Talman, "Variable dimension fixed point algorithms and triangulations," Mathematical Centre Tracts 128, Mathematisch Centrum, Amsterdam, The Netherlands, 1980.

[11] M.J. Todd, "Improving the convergence of fixed point algorithms," Mathematical Programming Studies 7 (1978) 151-169.

[12] H. Tuy, Ng. van Thoai and Le d. Muu, "A modification of Scarf's algorithm allowing restarting," *Mathematische Operationsforschung und Statistik Series Optimization* 9 (1978) 357-372.

[13] A.H. Wright, "The octahedral algorithm, a new simplicial fixed point algorithm," *Mathmetical Programming* 21 (1981) 47-69.