

A simplicial variable dimension restart algorithm
to find economic equilibria on the unit simplex
using $n(n + 1)$ rays

93-013

R.A.M.G. Joosten¹ & A.J.J. Talman²

¹Maastricht Economic Research Institute on Innovation and Technology, (MERIT)
and Department of Mathematics, University of Limburg, P.O. Box 616, 6200 MD
Maastricht, The Netherlands.

²Department of Econometrics, Tilburg University, P.O. Box 90153, 5000 LE Tilburg,
The Netherlands

June 1993

A simplicial variable dimension restart algorithm to find economic equilibria on the unit simplex using $n(n + 1)$ rays

R.A.M.G. Joosten*, A.J.J. Talman†

June 8, 1993

Abstract

A new simplicial variable dimension restart algorithm is introduced to solve the zero point or equilibrium problem on the n -dimensional unit simplex S^n . The number of one-dimensional sets or rays along which the algorithm can leave the arbitrarily chosen starting point is $n(n + 1)$. The simplicial subdivision which underlies the algorithm differs from the triangulations of S^n used so far. In case the algorithm is applied to find a general equilibrium price vector in an exchange economy with $n + 1$ commodities, it generates a piecewise linear path of prices, connecting the arbitrarily chosen starting price vector with an approximate equilibrium price vector. At an equilibrium price vector the excess demand of all commodities is zero. Along the path the price of a commodity is kept equal to the initial price if the excess demand of that commodity is neither minimal nor maximal. If the excess demand of a commodity becomes maximal, the price of that commodity is increased from its initial value, and if the excess demand becomes minimal, the price is decreased from its initial value. Also a new intersection theorem on S^n will be derived. This intersection theorem generalizes the well-known KKM lemma and an intersection result of Freidenfelds on S^n .

Key words: Zero point problem, triangulation, intersection theorem.

*Maastricht Economic Research institute on Innovation and Technology, and Department of Mathematics, University of Limburg, P.O. Box 616, 6200 MD Maastricht, The Netherlands.

†Department of Econometrics, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands.

1 Introduction

To find an economic equilibrium in an exchange economy or, equivalently, to find a zero point of a continuous excess demand function on the n -dimensional unit simplex, a class of algorithms has been developed, the so called simplicial variable dimension restart algorithms. The n -dimensional unit simplex is the subset of $\mathbf{R}^{n+1}$ given by $S^n = \{p \in \mathbf{R}_+^{n+1} \mid \sum_j p_j = 1\}$. A function $z : S^n \rightarrow \mathbf{R}^{n+1}$ is an excess demand function if z satisfies $p^T z(p) = \sum_j p_j z_j(p) = 0$ for all $p \in S^n$, as well as $z_i(p) \geq 0$ when $p_i = 0$. The former condition is known in economics as Walras' law and the latter one as a desirability condition. A point in the unit simplex could be interpreted as a normalized price vector assigning a (relative) price to each of the $n + 1$ goods in an economy.

A continuous excess demand function results from an exchange economy where none of the goods are indivisible and where all consumers have classical preferences and have finite, positive initial endowments of all goods. The interpretation of the above is as follows. Consumers satisfy their budget constraints for any given price vector due to (strong) monotonicity of their preferences, hence the value of the excess demand is zero for every consumer. Since the value of excess demand for the economy is equal to the sum of the values of the excess demands of all consumers, it follows that the value of excess demand for the economy is zero. Thus Walras' law will hold for any price vector. Desirability means that if a good is free, then total demand for that good will not be lower than total supply of it. This follows from the (strong) monotonicity of the preferences of the consumers (or even from local nonsatiability) combined with the circumstance that any amount of this good may be purchased for free. When at a price vector the excess demand of all goods is zero, the economy is in equilibrium.

Distinction between algorithms in the class of variable dimension restart algorithms is made by the number of one-dimensional sets or rays along which a starting point can be left. The following algorithms exist: a 2-ray algorithm (Doup and Talman (1987b)), several $(n + 1)$ -ray algorithms (van der Laan and Talman (1979), Doup and Talman (1987a)) and some $(2^{n+1} - 2)$ -ray algorithms (Doup, van der Laan and Talman (1987), Dang and Talman (1990)).

For a specific algorithm in this class, the unit simplex S^n is subdivided in n -dimensional simplices such that certain t -dimensional subsets of S^n , $0 \leq t \leq n$, depending on the starting point are subdivided in t -simplices in a manner which is typical to just that algorithm. Starting from this arbitrary

point of the unit simplex a path or sequence of adjacent simplices of varying dimension is generated until a simplex is found that yields an approximating zero point of the function z . Since there exists only a finite number of simplices (of variable dimension) and provisions are taken that any simplex is visited at most once, the algorithm finds such an approximating solution in a finite number of steps. Simplicial variable dimension restart algorithms generalize Scarf's algorithm, see Scarf (1967, 1973). In Scarf's algorithm the starting point cannot be chosen arbitrarily but has to be chosen equal to one of the corner points of S^n .

If the accuracy of an approximating solution is not satisfactory, the algorithm is restarted in or close to the approximation found in the previous round, using a finer simplicial subdivision. The algorithm generally finds a better approximating solution within typically a small number of steps. Such restarts can be made until a vector in S^n is found where excess demand is close enough to zero.

An excellent survey of variable dimension restart algorithms to find economic equilibria on the unit simplex may be found in Doup (1988, part II). The original $(n + 1)$ -ray algorithms employ either Scarf's or Sperner's labelling rule. In Scarf's labelling rule, to every point in the unit simplex the index of one good with the highest excess demand at that price vector is assigned as a label. In Sperner's labelling rule the index of one good with the lowest excess demand is assigned. When Scarf's rule is used the $(n + 1)$ -ray algorithm leaves the starting point by increasing the price of the good corresponding with the index assigned to it. When Sperner's rule is used the $(n + 1)$ -ray algorithm leaves the starting point by decreasing the price of the good corresponding with the index assigned to it.

Each $(2^{n+1} - 2)$ -ray algorithm leaves the starting point by relatively decreasing the prices of all the goods with negative excess demand and increasing the prices of all the goods with positive excess demand. Since the difference between the numbers of directions to leave the starting point, $n + 1$ and $2^{n+1} - 2$, becomes relatively large when n increases, the question arises: Is there an algorithm that uses more than $n + 1$ but less than $2^{n+1} - 2$ rays?

The basic idea of the new algorithm in economic terms is to increase the price of a good from the initial price if the excess demand for it is maximal, and to decrease the price of a good from the initial price if the excess demand for it is minimal. This idea thus constitutes a synthesis of the approaches taken in the two $(n + 1)$ -ray algorithms mentioned above. Assuming that the starting point is not an equilibrium, the algorithm is such

that initially the price of precisely one good with highest excess demand will be increased and the price of precisely one good with lowest excess demand will be decreased by the same amount. When the starting point is not an equilibrium the goods with minimum and maximum excess demand are different goods. Assuming nondegeneracy, at the starting point exactly one of the $n + 1$ goods has maximum excess demand, while just one of the remaining n goods has minimum excess demand. It follows then that there exist $n(n + 1)$ different combinations, giving as many (potential) directions to leave the starting point.

The interpretation of the path followed by the algorithm is as follows. Suppose some reluctance to change prices exists among agents and the propensity to adjust prices rises only if apparent scarcity or abundance of goods occur. Then decreasing the price of the most abundant good and increasing the price of the scarcest good seems to be very natural.

Several of the simplicial variable dimension restart algorithms mentioned above, can also be used to derive and to prove certain intersection theorems. The $(n + 1)$ -ray algorithm using Sperner's labelling can be applied for an alternative proof of the well-known lemma of Knaster, Kuratowski and Mazurkiewicz (1929). The $(n + 1)$ -ray algorithm using Scarf's labelling can be used to prove an intersection result of Freidenfelds (1974). The $(2^{n+1} - 2)$ -ray algorithm can be employed to prove Shapley's lemma (1980), a generalization of the KKM lemma. The $n(n + 1)$ -ray algorithm to be presented in this paper, can be utilized to derive a new intersection theorem on S^n generalizing both the KKM lemma and the intersection result of Freidenfelds.

This paper is organized as follows. In Section 2 we give the definitions necessary in order to formulate the algorithm. Section 3 describes the algorithm and elaborates on the steps taken in the algorithm. In Section 4 we present the algorithm. In Section 5 we derive a new intersection theorem on S^n .

2 Preliminaries

Let $z : S^n \rightarrow \mathbb{R}^{n+1}$ be a continuous excess demand function, i.e. $p^T z(p) = 0$ for all $p \in S^n$, and $z_j(p) \geq 0$ whenever $p_j = 0$. To find a zero point of z , let $v \in \text{int } S^n$ be an arbitrary point in the relative interior of S^n . The point v will be the starting point of the algorithm. Then the algorithm approximately follows a set of points in S^n such that for each point p of this

set it holds that

$$\begin{aligned} p_j &\leq v_j & \text{if } z_j(p) &= \min_h z_h(p), \\ p_j &= v_j & \text{if } \min_h z_h < z_j(p) < \max_h z_h(p), \\ p_j &\geq v_j & \text{if } z_j(p) &= \max_h z_h(p). \end{aligned} \quad (1)$$

Assuming regularity and nondegeneracy the set of points in S^n satisfying (1) consists of piecewise smooth loops and paths. Exactly one path connects the starting point v with a solution to the zero point problem. This path, to be denoted by P , can be interpreted as follows in economic terms in case at price vector p the vector $z(p)$ is the excess demand in an exchange economy with $n+1$ commodities. Along the first (linear) piece of the path the price of the commodity with highest excess demand is increased and the price of the commodity with lowest excess demand is decreased. In general, along the path the price of a commodity is increased from the initial price value when its excess demand becomes maximal and it is decreased from the initial price value when its excess demand becomes minimal, while the prices of the commodities not having minimum or maximum excess demand are kept equal to the initial prices. When along the path the price of a commodity with maximal (minimal) excess demand becomes equal to the initial price value of that commodity, the excess demand is allowed to decrease from the maximal value (to increase from the minimal value) while its price is kept equal to the initial price. The path of points satisfying (1) and leading from v to a zero point of z , is followed by the algorithm by generating a piecewise linear path in a simplicial subdivision of S^n . Each linear piece of the latter path lies in some simplex and is generated by making a linear programming pivot step in a system of linear equations.

In the sequel we use the following notations. For every positive integer k let I^k be the set of integers $\{1, \dots, k\}$. For $A \subset I^{n+1}$, $-A$ is defined to be the set $\{-j | j \in A\}$. Furthermore, let T be a subset of $I^{n+1} \cup -I^{n+1}$ such that $T \cap I^{n+1} \neq \emptyset$, $T \cap -I^{n+1} \neq \emptyset$, $T \cap (-T) = \emptyset$. Let $T^+ = \{j \in I^{n+1} | j \in T\}$, $T^- = \{i \in I^{n+1} | -i \in T\}$, and $T(c) = \{h \in I^{n+1} | h \notin T^+ \cup T^-\}$.

To describe the $n(n+1)$ -ray algorithm we introduce the following sets in S^n .

Definition 2.1 For T as described above the set $A(T)$ is given by

$$A(T) = \{p \in S^n \mid \begin{aligned} p_i &\leq v_i & \text{if } i &\in T^-, \\ p_i &= v_i & \text{if } i &\in T(c), \\ p_i &\geq v_i & \text{if } i &\in T^+. \end{aligned}\}.$$

Notice that if a point p in S^n lies on the path P , then there exists a set T such that $p \in A(T)$, $z_i(p) = \max_h z_h(p)$ if $i \in T^+$, and $z_i(p) = \min_h z_h(p)$ if $i \in T^-$. The simplicial subdivision underlying the algorithm must be such that it subdivides each subset $A(T)$ into t -dimensional simplices where t , $t = |T| - 1$ is the dimension of $A(T)$. To describe such a triangulation we introduce for every set T a projection $p(T)$ of v on $\text{bd } S^n$, the boundary of S^n , see Figure 1 for $n = 2$.

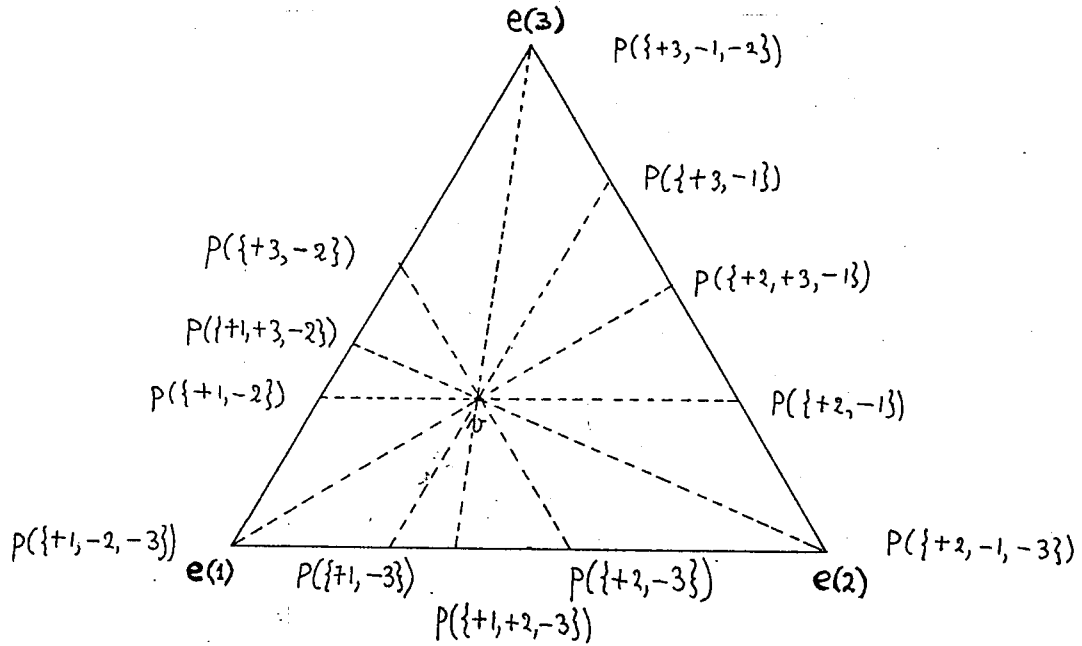


Figure 1: The projections $p(K)$ of v on $\text{bd } S^n$, $n = 2$.

Definition 2.2 Let K be a subset of $I^{n+1} \cup -I^{n+1}$, satisfying $K^+ \neq \emptyset$, $K^- \neq \emptyset$, $K^+ \cap K^- = \emptyset$, then the projection $p(K)$ of v on the intersection of $A(K)$ and $\text{bd } S^n$ is given by

$$\begin{aligned} p_j(K) &= 0 & \text{if } j \in K^-, \\ p_j(K) &= v_j & \text{if } j \in K(c), \\ p_j(K) &= v_j(1 - \sum_{k \in K(c)} v_k)(\sum_{i \in K^+} v_i)^{-1} & \text{if } j \in K^+. \end{aligned}$$

With these projections we define specific subsets of the sets $A(T)$ as follows, see Figure 2 for $n = 2$.

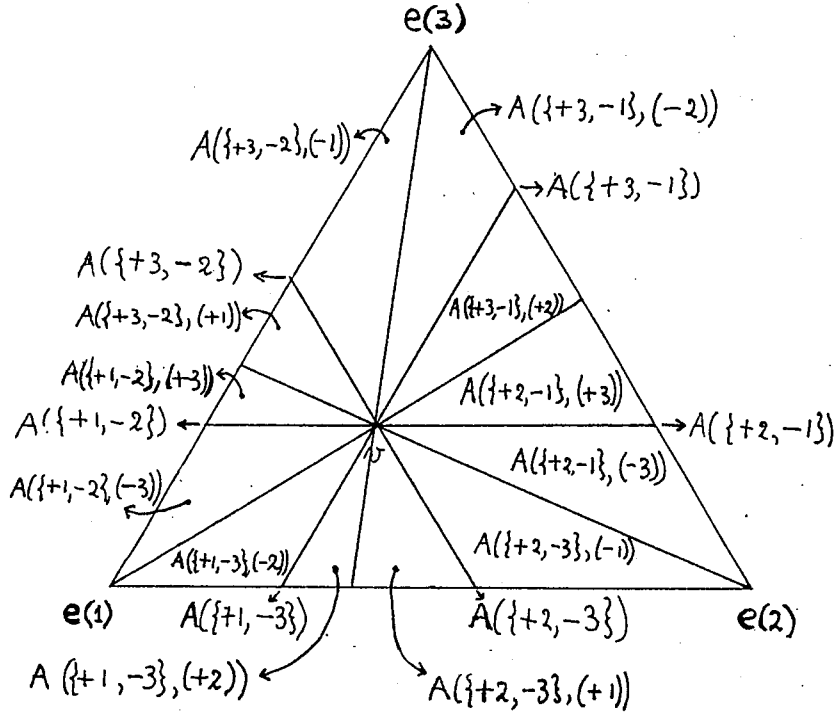


Figure 2: The sets $A(T^0, \Gamma(T^1))$, $n = 2$.

Definition 2.3 Let T be as described, above. Let T^0 be a subset of T , with exactly one positive and one negative element, let $T^1 = T \setminus T^0$, and let $\Gamma(T^1) = (\Gamma_1, \dots, \Gamma_{t-1})$ be a permutation of the $t - 1$ elements of T^1 . Then $A(T^0, \Gamma(T^1))$ is given by

$$A(T^0, \Gamma(T^1)) = \{p \in S^n | p = v + \sum_{j=1}^t \alpha(j)q(j), 0 \leq \alpha(t) \leq \dots \leq \alpha(1) \leq 1\},$$

where $q(1) = p(T^0) - v$ and $q(h) = p(T^0 \cup \{\Gamma_1, \dots, \Gamma_{h-1}\}) - p(T^0 \cup \{\Gamma_1, \dots, \Gamma_{h-2}\})$, $h = 2, \dots, t$.

$A(T)$ is the union of all possible sets $A(T^0, \Gamma(T^1))$ where T^0 , T^1 and $\Gamma(T^1)$ are defined as above. Each $A(T)$ is triangulated by subdividing every

subset $A(T^0, \Gamma(T^1))$ of $A(T)$ in t -simplices as follows. Let m be some positive integer.

Definition 2.4 The subset $A(T^0, \Gamma(T^1))$ of $A(T)$ is subdivided into t -simplices $\sigma(a, \pi)$ with vertices $y^1, \dots, y^{t+1}$ according to the QV -triangulation with grid size m^{-1} such that:

- 1) $y^1 = v + m^{-1} \sum_{j=1}^t a(j)q(j)$ with $a = (a(1), \dots, a(n+1))$ an integer vector satisfying $a(n+1) = \dots = a(t+1) = 0 \leq a(t) \leq \dots \leq a(1) \leq m-1$.
- 2) $y^{i+1} = y^i + m^{-1}q(\pi_i)$ for $i = 1, \dots, t$ with $\pi = (\pi_1, \dots, \pi_t)$ a permutation of the elements of I^t such that $p > p'$ if $a(\pi_p) = a(\pi_{p'})$ and $\pi_p = k$ and $\pi_{p'} = k-1$ for some $k \in \{2, \dots, t\}$.

The union of the simplicial subdivisions of the sets $A(T^0, \Gamma(T^1))$ where T^0, T^1 and $\Gamma(T^1)$ are defined as above is a simplicial subdivision with grid size m^{-1} of $A(T)$, whereas the union of these subdivisions of $A(T)$ over all T induces a simplicial subdivision with grid size m^{-1} of S^n , see Figure 3 for $n = 2$ and $m = 2$.

This simplicial subdivision is called the QV -triangulation because in some sense, it is a combination of the Q -triangulation and the V -triangulation, see Scarf (1973) and Doup and Talman (1987a). We remark that if all components of v are nonnegative multiples of m^{-1} , then the Q -triangulation with grid size m^{-1} also simplicially subdivides every subset $A(T)$.

Definition 2.5 Let T be as described above. For $g \in \{t, t-1\}$, let σ be a g -simplex with vertices $y^1, \dots, y^{g+1}$ in S^n , then σ is T -complete if the system of linear equations

$$\sum_{i=1}^{g+1} \lambda_i \begin{bmatrix} z(y^i) \\ 1 \end{bmatrix} - \sum_{h \in T(c)} \mu_h \begin{bmatrix} e(h) \\ 0 \end{bmatrix} - \alpha \begin{bmatrix} e(T^-) \\ 0 \end{bmatrix} - \beta \begin{bmatrix} e(T^+) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad (2)$$

where $e(h)$ is the h -th unit vector in $\mathbb{R}^{n+1}$ and $e(Q) = \sum_{j \in Q} e(j)$ for $Q = T^+$ or $Q = T^-$, has a solution satisfying $\lambda_i \geq 0$ for all $i \in I^{g+1}$, $\alpha \leq 0$, $\beta \geq 0$, $\alpha \leq \mu_h \leq \beta$ for all $h \in T(c)$.

Let $\lambda = (\lambda_1, \dots, \lambda_{g+1})$, $\mu = (\mu_h, h \in T(c))$. It is easy to see that there are $n+3$ variables if $g = t$ and $n+2$ variables if $g = t-1$. There are $n+2$ equations to solve. A solution $(\lambda, \mu, \alpha, \beta)$ therefore is (in general) not unique if $g = t$. In the latter case a solution $(\lambda, \mu, \alpha, \beta)$ is called a basic solution if

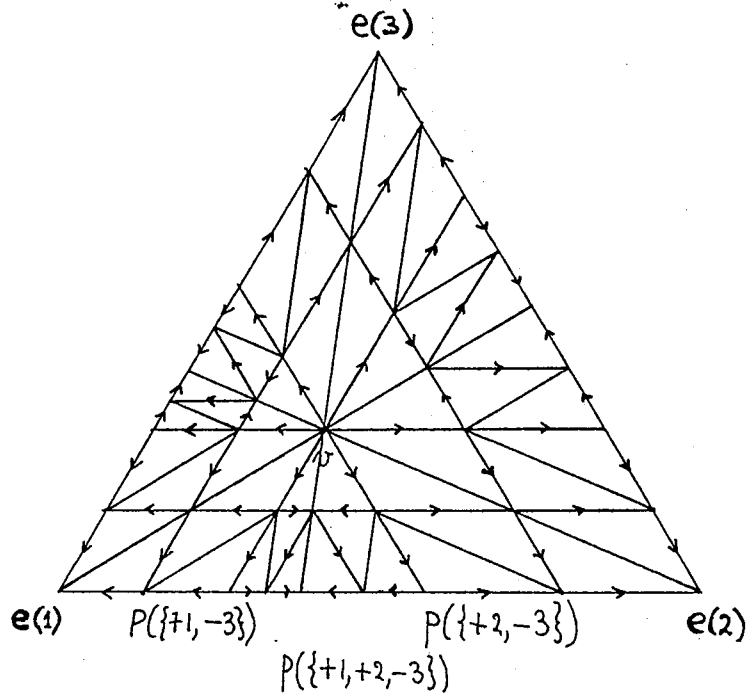


Figure 3: The QV -triangulation of S^n and of the sets $A(T^0, \Gamma(T^1))$, $n = 2$, $m = 2$.

at least one of the variables α, β, λ_i , $i \in I^{g+1}$, equals zero or at least one of the μ_h 's, $h \in T(c)$, equals either α or β .

Assumption 2.1 (Nondegeneracy assumption) Suppose $\sigma(y^1, \dots, y^{g+1})$ is a T -complete g -simplex of the QV -triangulation with grid size m^{-1} in $A(T)$. If $g = t - 1$, then system (2) has *exactly one* solution $(\lambda, \mu, \alpha, \beta)$, satisfying $\lambda_j > 0$ for all $j \in I^t$, $\alpha < \mu_h < \beta$ for all $h \in T(c)$, $\alpha < 0$, $\beta > 0$. Furthermore, if $g = t$, then at every solution $(\lambda, \mu, \alpha, \beta)$ either *at most one* variable α, β, λ_j for $j \in I^{t+1}$ equals zero, or *at most one* μ_h , $h \in T(c)$, equals either α or β .

For a T -complete t -simplex $\sigma(a, \pi)$ with vertices $y^1, \dots, y^{t+1}$ in a given $A(T)$, the set of solutions of system (2) determines under Assumption 2.1 a line segment of solutions $(\lambda, \mu, \alpha, \beta)$. A line segment of solutions induces a linear piece of points $p = \sum_{i=1}^{t+1} \lambda_i y^i$ in $\sigma(a, \pi)$. This linear piece has two

different end points corresponding with the two basic solutions of system (2). Therefore this linear piece of points in $\sigma(a, \pi)$ can be followed by making a linear programming pivot step in the system of linear equations above with the variable that is binding at one of the end points. Let the *piecewise linear approximation* Z of z with respect to the QV -triangulation be defined as

$$Z(p) = \sum_{i=1}^{t+1} \lambda_i z(y^i)$$

where $p = \sum_{i=1}^{t+1} \lambda_i y^i \in \sigma(a, \pi)$, $\sum_{i=1}^{t+1} \lambda_i = 1$, $\lambda_i \geq 0$ for all $i \in I^{t+1}$.

For $p = \sum_{i=1}^{t+1} \lambda_i y^i$ in a T -complete t -simplex $\sigma(a, \pi)$ in $A(T)$ with solution $(\lambda, \mu, \alpha, \beta)$ it holds that

$$\begin{aligned} Z_i(p) &= \alpha & \text{and} & \quad p_i \leq v_i & \text{if} & \quad i \in T^-, \\ \alpha \leq Z_i(p) \leq \beta & & \text{and} & \quad p_i = v_i & \text{if} & \quad i \in T(c), \\ Z_i(p) &= \beta & \text{and} & \quad p_i \geq v_i & \text{if} & \quad i \in T^+. \end{aligned} \quad (3)$$

Starting at v the algorithm generates a piecewise linear path of points such that each point p on the path satisfies (3), i.e., the algorithm approximately follows the piecewise smooth path P by replacing in (1) the function z by its piecewise linear approximation Z . To do so, it induces a sequence of adjacent T -complete t -simplices $\sigma(a, \pi)$ in $A(T)$ with T -complete common facets, for varying sets T . Each simplex of the sequence contains a line segment as piece of the path, being generated by making a linear programming pivot step in system (2), as described in the next section.

3 The algorithm

The algorithm may start in an arbitrary interior point v of the unit simplex and assuming that this point is not a zero point of z , we can determine which component of $z(v)$ is maximal and which component is minimal. The algorithm will leave v by increasing p_i if $z_i(v) = \max_h z_h(v)$, and by decreasing p_j if $z_j(v) = \min_h z_h(v)$. We assume that both indices i and j are uniquely determined. So, the starting point v is left along the one-dimensional set $A(\{+i, -j\})$. The algorithm will generate a path of connected line segments contained in a sequence of adjacent T -complete t -simplices $\sigma(a, \pi)$ having T -complete common facets in sets $A(T)$ for varying sets T , by going for each simplex in the sequence from one basic solution of (2) to the other one. This is accomplished by performing a linear programming pivot step in (2) with the variable that is binding at the first basic solution.

For the set of solutions $(\lambda, \mu, \alpha, \beta)$ of (2) with respect to a T -complete t -simplex $\sigma(a, \pi)$ of the QV -triangulation with grid size m^{-1} in a given set $A(T^0, \Gamma(T^1))$, the line segment of points $p = \sum_{j=1}^{t+1} \lambda_j y^j$ connects exactly two points corresponding with the two basic solutions. Each point p on this line segment satisfies (3). If $\sigma(a, \pi)$ is generated by the algorithm, it moves in $\sigma(a, \pi)$ from one point corresponding with a basic solution, say p^1 , to the other one, say p^2 . Then one of the following two situations may occur:

- 1) p^2 is an element of a T -complete facet of $\sigma(a, \pi)$ and due to the non-degeneracy assumption exactly one λ_i , $i \in I^{t+1}$, equals zero,
- 2) p^2 is in the interior of $\sigma(a, \pi)$ and due to the nondegeneracy assumption, either exactly one μ_h , $h \in T(c)$, equals either α or β , or either α or β equals zero.

In Situation 1 the algorithm distinguishes two cases. Suppose at p^2 that $\lambda_s = 0$, then p^2 is a point in the T -complete facet τ opposite vertex y^s :

- 1a) τ lies in the interior of $A(T^0, \Gamma(T^1))$,
- 1b) τ lies in the boundary of $A(T^0, \Gamma(T^1))$.

In Situation 2 the algorithm again distinguishes two cases:

- 2a) either $\mu_h = \alpha$ or $\mu_h = \beta$ for some $h \in T(c)$,
- 2b) either $\alpha = 0$ or $\beta = 0$.

In Situation 1a, the algorithm moves from $\sigma(a, \pi)$ through the facet τ to the unique t -simplex $\sigma(\underline{a}, \underline{\pi})$ in $A(T^0, \Gamma(T^1))$ having the facet τ in common with $\sigma(a, \pi)$. The parameters $\underline{a}$ and $\underline{\pi}$ of the new simplex are given in Table 3.1. By making the adaptations as described above, a new vertex is generated, say $\underline{y}^{s'}$, for some $s' \in I^{t+1}$. This new vertex of $\sigma(\underline{a}, \underline{\pi})$ lies opposite the facet τ in common with $\sigma(a, \pi)$ and replaces y^s in (2). Next a linear programming pivot step is performed in (2) with $(z^T(\underline{y}^{s'}), 1)^T$ by increasing $\lambda_{s'}$ from zero. Thus a line segment in $\sigma(\underline{a}, \underline{\pi})$ is followed and the algorithm finds a new basic solution.

s	$\underline{\pi}$	$\underline{a}$
$s = 1$	$(\pi_2, \pi_3, \dots, \pi_t, \pi_1)$	$a + e(\pi_1)$
$2 \leq s \leq t$	$(\pi_1, \dots, \pi_{s-2}, \pi_s, \pi_{s-1}, \pi_{s+1}, \dots, \pi_t)$	a
$s = t + 1$	$(\pi_t, \pi_1, \pi_2, \dots, \pi_{t-1})$	$a - e(\pi_t)$

Table 3.1 Given $\sigma(a, \pi)$ with parameters $a = (a(1), \dots, a(n+1))$ and $\pi = (\pi_1, \dots, \pi_t)$, $\underline{\pi}$ and $\underline{a}$ are the parameters of the t -simplex $\sigma(\underline{a}, \underline{\pi})$ sharing the facet τ opposite the vertex y^s with $\sigma(a, \pi)$.

In Situation 1b, the algorithm distinguishes three possibilities in view of Definition 2.4:

- 1) $s = 1, \pi_1 = 1, a(\pi_1) = m - 1,$
 - 2) $2 \leq s \leq t, a(\pi_{s-1}) = a(\pi_s), \pi_{s-1} = k - 1$ and $\pi_s = k$ for a certain $k, 2 \leq k \leq t,$
 - 3) $s = t + 1, \pi_t = t, a(\pi_t) = 0.$
- 1) The facet τ lies in $\text{bd } S^n$. Therefore, for some $i \in T^-$ it holds that $y_i = 0$ for every $y \in \tau$. This implies $y_i^j = 0$ and hence $z_i(y^j) \geq 0$ for all $j = 2, \dots, t + 1$. Consequently, $Z_i(p) \geq 0$ where $p = \sum_{j=1}^{t+1} \lambda_j y^j$. On the other hand, since $i \in T^-$, we must have that $Z_i(p) = \alpha < 0$. This yields a contradiction, so this situation cannot occur.
 - 2) The facet τ also lies in a different subset of $A(T)$, say $A(\underline{T}^0, \underline{T}^1)$, and is therefore a facet of a unique t -simplex $\sigma(\underline{a}, \underline{\pi})$ in this subset. The necessary adaptations to characterize $\sigma(\underline{a}, \underline{\pi})$ and $A(\underline{T}^0, \underline{T}^1)$ may be found in Table 3.2. The vertex $\underline{y}^s$ of $\sigma(\underline{a}, \underline{\pi})$ opposite τ replaces the vertex y^s of $\sigma(a, \pi)$ in (2). Next a linear programming pivot step is performed with $(z^T(\underline{y}^s), 1)^T$ by increasing λ_s from zero. Thus a line segment is followed in $\sigma(\underline{a}, \underline{\pi})$ until the algorithm finds another basic solution.

Case	$\underline{T}^0, \underline{T}^1, \underline{\Gamma}(\underline{T}^1), \underline{a}, \underline{\pi}$
$k > 2$	$\underline{T}^0 = T^0, \underline{T}^1 = T^1,$ $\underline{\Gamma}(\underline{T}^1) = (\Gamma_1, \dots, \Gamma_{k-3}, \Gamma_{k-1}, \Gamma_{k-2}, \Gamma_k, \dots, \Gamma_{t-1})$ $\underline{a} = a, \underline{\pi} = \pi$
$k = 2,$ $T^0 = \{+h, -j\}$	if $\Gamma_1 > 0$ then $\underline{T}^0 = \{\Gamma_1, -j\}, \underline{T}^1 = T \setminus \underline{T}^0,$ $\underline{\Gamma}(\underline{T}^1) = (+h, \Gamma_2, \dots, \Gamma_{t-1})$ $\underline{a} = a, \underline{\pi} = \pi$ if $\Gamma_1 < 0$ then $\underline{T}^0 = \{+h, \Gamma_1\}, \underline{T}^1 = T \setminus \underline{T}^0,$ $\underline{\Gamma}(\underline{T}^1) = (-j, \Gamma_2, \dots, \Gamma_{t-1})$ $\underline{a} = a, \underline{\pi} = \pi$

Table 3.2 Adaptations to be made when the facet τ opposite the vertex y^s of $\sigma(a, \pi)$ in $A(T^0, \Gamma(T^1))$ lies in $\text{bd } A(T^0, \Gamma(T^1))$ but not in $\text{bd } A(T)$.

- 3) The facet τ lies in $\text{bd } A(T)$ but not in $\text{bd } S^n$. More precisely, τ is a $(t-1)$ -simplex in $A(T \setminus \{\Gamma_{t-1}\})$. Then the element Γ_{t-1} is removed from T and from T^1 , and the algorithm continues in $A(\underline{T})$ with $\underline{T} = T \setminus \{\Gamma_{t-1}\}$. The simplex τ is the $(t-1)$ -simplex $\sigma(\underline{a}, \underline{\pi})$ in $A(\underline{T}^0, \underline{\Gamma}(\underline{T}^1))$ as described in Table 3.3. A linear programming pivot step is performed in (2) with $(e^T(j), 0)^T$, where $j = \Gamma_{t-1}$ if $\Gamma_{t-1} > 0$ and $j = -\Gamma_{t-1}$ if $\Gamma_{t-1} < 0$, by decreasing μ_j from β if $\Gamma_{t-1} > 0$, and by increasing μ_j from α if $\Gamma_{t-1} < 0$. Thus, p_j is kept equal to v_j and a line segment is followed in the simplex $\sigma(\underline{a}, \underline{\pi})$ of one dimension lower until the algorithm finds a new basic solution in this simplex.

$\underline{T}^0, \underline{T}^1, \underline{\Gamma}(\underline{T}^1), \underline{a}, \underline{\pi}$
$\underline{T}^0 = T^0, \underline{T}^1 = T^1 \setminus \{\Gamma_{t-1}\},$ $\underline{\Gamma}(\underline{T}^1) = (\Gamma_1, \dots, \Gamma_{t-2})$ $\underline{a} = a, \underline{\pi} = (\pi_1, \dots, \pi_{t-1})$

Table 3.3 Adaptations to be made when the facet τ opposite vertex y^s of $\sigma(a, \pi)$ in $A(T^0, \Gamma(T^1))$ lies in $\text{bd } A(T)$ but not in $\text{bd } S^n$.

If at the basic solution corresponding with p^2 one of the variables μ_h , $h \in T(c)$ becomes equal to α or β as in Situation 2a, then let $j = k$ if $\mu_k = \alpha$, and let $j = -k$ if $\mu_k = \beta$. The simplex $\sigma(a, \pi)$ is the facet of exactly one $(t+1)$ -simplex $\sigma(\underline{a}, \underline{\pi})$ in the subset $A(\underline{T}^0, \underline{\Gamma}(\underline{T}^1))$ of $A(\underline{T})$ where $\underline{T} = T \cup \{j\}$, as described in Table 3.4. A linear programming pivot step is performed with $(z^T(\underline{y}^{t+2}), 1)^T$ by increasing λ_{t+2} from zero, where $\underline{y}^{t+2}$ is

the vertex of $\sigma(\underline{a}, \underline{\pi})$ opposite $\sigma(a, \pi)$. The algorithm enters a subset of the unit simplex of one dimension higher, such that p_k is decreased from v_k if $\mu_k = \alpha$ and increased from v_k if $\mu_k = \beta$. Thus a line segment in a simplex of one dimension higher is followed until a new basic solution is found.

$\underline{T}^0, \underline{T}^1, \underline{\Gamma}(\underline{T}^1), \underline{a}, \underline{\pi}$
$\underline{T}^0 = T^0, \underline{T}^1 = T^1 \cup \{j\},$
$\underline{\Gamma}(\underline{T}^1) = (\Gamma_1, \dots, \Gamma_{t-1}, j),$
$\underline{a} = a, \underline{\pi} = (\pi_1, \dots, \pi_t, t+1)$

Table 3.4 Adaptations to be made if μ_h is equal to α or β ; $j = h$ if $\mu_h = \alpha$, and $j = -h$ if $\mu_h = \beta$.

If at a basic solution either α or β becomes equal to zero as in Situation 2b, then the algorithm terminates with solution, say, $(\lambda^*, \mu^*, \alpha^*, \beta^*)$. We call the point $p^* = \sum_{i=1}^{t+1} \lambda_i^* y^i$ in S^n an approximating solution of the zero point problem of z . Since at p^* all elements of $Z(p^*)$ are either positive or negative and Z is an approximation of z , this gives rise to the conclusion that p^* is close to a solution or at least to a point where all components of z are small in absolute value. How close depends on the grid size m^{-1} .

If the accuracy of an approximating solution is not sufficient, the algorithm may be restarted in p^* with a smaller grid size in order to improve the accuracy of the solution.

4 The steps of the algorithm

The steps of the $n(n+1)$ -ray algorithm to approximate a zero point of a continuous excess demand function z on S^n , starting at $v \in \text{int } S^n$ and using the QV -triangulation with grid size m^{-1} as underlying simplicial subdivision of S^n , are as follows.

Step 0. Let $z_k(v) = \min_h z_h(v)$ and $z_l(v) = \max_h z_h(v)$. Set $y^1 = v$, $T^0 = \{l, -k\}$, $T = T^0$, $T^1 = \emptyset$, $t = 1$, $\pi = (1)$, $a = (0, \dots, 0)^T$, $w = 2$, $\lambda_1 = 1$, $\alpha = z_k(v)$, $\beta = z_l(v)$, $\mu_h = z_h(v)$, $h \in T(c)$.

Step 1. Perform a linear programming pivot step with $(z^T(y^w), 1)^T$ by increasing λ_w from zero in the following system of linear equations:

$$\sum_{i=1}^{t+1} \lambda_i \begin{bmatrix} z(y^i) \\ 1 \end{bmatrix} - \sum_{h \in T(c)} \mu_h \begin{bmatrix} e(h) \\ 0 \end{bmatrix} - \alpha \begin{bmatrix} e(T^-) \\ 0 \end{bmatrix} - \beta \begin{bmatrix} e(T^+) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (4)$$

where $\lambda_i \geq 0$ for $i \in I^{t+1}$, $\alpha \leq 0$, $\beta \geq 0$, $\alpha \leq \mu_h \leq \beta$ for $h \in T(c)$.
 When μ_h becomes equal to α or β for some $h \in T(c)$, or when α or β becomes equal to zero, go to Step 3.
 Else λ_s becomes equal to zero for some $s \in I^{t+1} \setminus \{w\}$.

Step 2. If $2 \leq s \leq t$, $a(\pi_{s-1}) = a(\pi_s)$, $\pi_{s-1} = k-1$ and $\pi_s = k$ for a certain k , $2 \leq k \leq t$, then we can find the necessary adjustments in Table 3.2 to obtain $\sigma(\underline{a}, \underline{\pi})$. Let s' be the index of the new vertex of $\sigma(\underline{a}, \underline{\pi})$, set $w = s'$, $\sigma(a, \pi) = \sigma(\underline{a}, \underline{\pi})$, and return to Step 1.
 If $s = t+1$, $\pi_t = t$ and $a(\pi_t) = 0$, go to Step 4.
 Else replace y^s according to Table 3.1 to obtain $\sigma(\underline{a}, \underline{\pi})$. Let s' be the index of the new vertex of $\sigma(\underline{a}, \underline{\pi})$, set $w = s'$, $\sigma(a, \pi) = \sigma(\underline{a}, \underline{\pi})$, and return to Step 1.

Step 3. If $\alpha = 0$ or $\beta = 0$, stop with solution $(\lambda^*, \mu^*, \alpha^*, \beta^*)$ and $p^* = \sum_{i=1}^{t+1} \lambda_i^* y^i$ is an approximating zero point of z .
 Else if $\mu_h = \alpha$, then $-h$ is added to T , if $\mu_h = \beta$, then h is added to T .
 The relevant adaptations may be found in Table 3.4 to obtain $\sigma(\underline{a}, \underline{\pi})$, set $t = t+1$, $w = t+1$, $\sigma(a, \pi) = \sigma(\underline{a}, \underline{\pi})$, and return to Step 1.

Step 4. The set T becomes $T \setminus \{\Gamma_{t-1}\}$. The other adjustments may be found in Table 3.3 to find $\sigma(\underline{a}, \underline{\pi})$. Set $\gamma = \Gamma_{t-1}$, $t = t-1$, $\sigma(a, \pi) = \sigma(\underline{a}, \underline{\pi})$ and perform a linear programming pivot step in (4) with $(e^T(j), 0)^T$ where $j = |\gamma|$, by increasing the variable μ_j from α if $\gamma < 0$, and by decreasing the variable μ_j from β if $\gamma > 0$. If either μ_h becomes equal to α or β for a certain $h \in T(c)$, or if α or β becomes equal to zero, then return to Step 3.
 Else λ_s becomes zero for some $s \in I^{t+1}$, and return to Step 2.

Every round gives an approximating solution p^* , satisfying either $Z_j(p^*) \geq 0$ for all $j \in I^{n+1}$ or $Z_j(p^*) \leq 0$ for all $j \in I^{n+1}$. Whenever this approximation of a zero point of z is not satisfactory, the algorithm may be restarted with v equal to p^* and a larger m . Thus within a finite number of restarts any a priori chosen accuracy can be reached. We remark that the algorithm could be continued in Step 3 when α or β becomes 0. In that case the algorithm should be terminated when in Step 2 it holds that $s = 1$, $\pi_1 = 1$ and $a(\pi_1) = m-1$, or when α becomes equal to β . In the former case it holds at p^* that $Z_j(p^*) \geq 0$ for all $j \in I^{n+1}$, and in the latter case that $Z_j(p^*) = \alpha$ for all $j \in I^{n+1}$ for some $\alpha \in \mathbb{R}$. In both cases it follows, from the property that $p^T z(p) = 0$ for all $p \in S^n$, that p^* is also an approximating solution.

5 The intersection theorem

Simplicial variable dimension algorithms can be used to find an intersection point of a certain collection of closed subsets covering S^n . The most well-known intersection theorem is probably the KKM lemma, see Knaster, Kuratowski and Mazurciewicz (1929). Another well-known intersection theorem is due to Freidenfelds and this result can be interpreted as a dual to the KKM lemma. For a survey on intersection theorems on the unit simplex, we refer to Talman (1991).

The $n(n+1)$ -ray algorithm developed in the previous sections also leads directly to an intersection theorem of closed subsets covering S^n . This theorem can be considered as a hybrid of both lemmas above and these lemmas immediately follow as a corollary. In the next theorem S^n is covered by $n(n+1)$ closed subsets. Some of these subsets might be empty. Each subset of the collection is indexed by a pair $(+i, -j)$ with $i, j \in I^{n+1}$.

Theorem 5.1 *Let $C(+i, -j)$, i and $j \in I^{n+1}$, be closed sets covering S^n such that for every $x \in \text{bd } S^n$ it holds that $x \in C(+i, -j)$ for some $j \in I^{n+1}$ satisfying $x_j > 0$ or $x \in C(+i, -j)$ for every $i \in I^{n+1}$ satisfying $x_i = 0$. Then there exist an $x^* \in S^n$ and a nonempty set $T^* \subset I^{n+1}$ satisfying that for every $i \in T^*$ there exists a $j \in T^*$ and a $k \in T^*$ such that $x^* \in C(+i, -j)$ and $x^* \in C(+k, -i)$.*

Proof. Without loss of generality we assume that $C(+i, -i) = \emptyset$, $i \in I^{n+1}$. For $x \in S^n$, let $z : S^n \rightarrow \mathbb{R}^{n+1}$ be defined by $z(x) = e(i) - e(j)$ for some i and $j \in I^{n+1}$ satisfying $x \in C(+i, -j)$ where $x_j > 0$ and if no such j exists then i is such that $x_i = 0$. Clearly, $\sum_{i=1}^{n+1} z_i(x) = 0$ for all $x \in S^n$. Given some fixed starting point $v \in \text{int } S^n$ let the $n(n+1)$ -algorithm be applied to z for a sequence of decreasing grid sizes $m_1^{-1}, m_2^{-1}, \dots$. Suppose the algorithm finds a vertex x^* in $\text{bd } S^n$ satisfying $x_j = 0$ for every j for which x lies in $C(+i, -j)$ for some $i \in I^{n+1}$. Then for every $i \in I^{n+1}$ satisfying $x_i^* = 0$ it holds that $x^* \in C(+i, -j_i)$ for some $j_i \in I^{n+1}$ with $x_{j_i}^* = 0$. Clearly, x^* satisfies the statement in the theorem with T^* equal to the set $\{j_i \in I^{n+1} | x^* \in C(+i, -j_i) \text{ and } x_i^* = 0\}$. Otherwise, in round k , $k = 1, 2, \dots$, the algorithm terminates with a, say, $t(k)$ -dimensional simplex σ^k with vertices $y^{k,1}, \dots, y^{k,t(k)+1}$ such that the system of linear equations

$$\sum_{h=1}^{t(k)+1} \lambda_h^k (e(i_h^k) - e(j_h^k)) = 0 \quad (5)$$

has a nonnegative solution $\lambda_1^k, \dots, \lambda_{t(k)+1}^k$ summing up to one and where i_h^k and $j_h^k \in I^{n+1}$ are such that the vertex $y^{k,h}$ of σ^k lies in $C(+i_h^k, -j_h^k)$, for $h = 1, \dots, t(k)+1$. Notice that due to the fact that $\sum_l z_l(y^{k,h}) = 0$ for every h , the algorithm can only terminate at Step 3 with both α and β equal to zero. The algorithm can not stop at the boundary of S^n because $z_l(x) \geq 0$ whenever $x_l = 0$.

Let $x^k = \sum_{h=1}^{t(k)+1} \lambda_h^k y^{k,h}$ be the approximating solution found in round $k, k = 1, 2, \dots$. When k goes to infinity there exists a subsequence $k(1), k(2), \dots$, such that $x^{k(m)}$ converges to some x^* in $S^n, t(k(m)) = t$ for all $m, \lambda_h^{k(m)}$ converges to some $\lambda_h^* \geq 0$, for all $h = 1, \dots, t+1$, with $\sum_{h=1}^{t+1} \lambda_h^* = 1$, and $i_h^{k(m)} = i_h$ and $j_h^{k(m)} = j_h$ for some i_h and $j_h \in I^{n+1}, h = 1, \dots, t+1$. Since the grid size m_k^{-1} converges to zero when k goes to infinity, we also have that $y^{k(m),h}$ converges to x^* for every $h = 1, \dots, t+1$. Since for every h we have that $y^{k(m),h} \in C(+i_h^{k(m)}, -j_h^{k(m)})$ and every $C(+i, -j)$ is closed, we obtain that $x^* \in C(+i_h, -j_h)$ for $h = 1, \dots, t+1$. Furthermore it follows from (5), when $k = k(m)$ and m goes to infinity

$$\sum_{h=1}^{t+1} \lambda_h^* (e(i_h) - e(j_h)) = 0. \quad (6)$$

Now, let $T^1 = \{i_h | \lambda_h^* > 0, h = 1, \dots, t+1\}$ and let $T^2 = \{j_h | \lambda_h^* > 0, h = 1, \dots, t+1\}$. We will show that $T^1 = T^2$. Take some $i \in T^1$ and suppose $i \notin T^2$. Then the i -th equation of (6) yields

$$0 = \sum_{h=1}^{t+1} \lambda_h^* e_i(i_h) \geq \lambda_k^* > 0$$

where k is such that $i_k = i$, yielding a contradiction.

Similarly, take some $j \in T^2$ and suppose $j \notin T^1$. Then the j -th equation of (6) yields

$$0 = - \sum_{h=1}^{t+1} \lambda_h^* e_j(j_h) \leq -\lambda_k^* < 0$$

where k is such that $j_k = j$, yielding also a contradiction.

Hence, $T^* = T^1 = T^2$ satisfies the statement of the theorem. Notice that $T^* \neq \emptyset$ since $\sum_{h=1}^{t+1} \lambda_h^* = 1$. $\square$

The intersection theorem is illustrated for $n = 2$ in the Figures 4 and 5. In Figure 4, the set T^* equals $\{1, 2, 3\}$ and x^* lies in the intersection of

$C(+1, -3)$, $C(+2, -1)$ and $C(+3, -2)$. In Figure 5, the set T^* equals $\{1, 3\}$ and $C(+1, -3) \cap C(+3, -1)$ consists of the point $e(2)$. From Theorem 5.1 both the KKM lemma and the intersection theorem of Freidenfelds on S^n follow as a corollary.

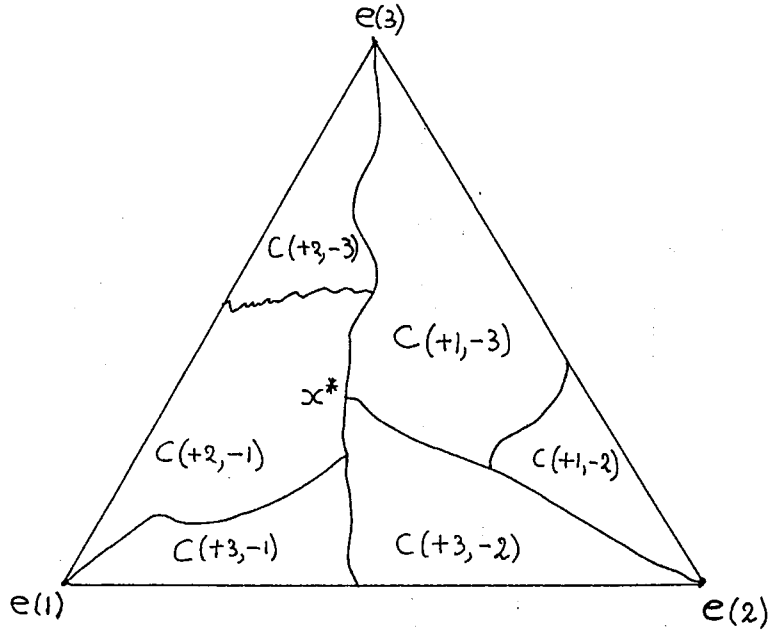


Figure 4: Illustration of the intersection theorem, $n = 2$, $T^* = \{1, 2, 3\}$.

Corollary 5.1 (KKM lemma)

Let $C_1, \dots, C_{n+1}$ be $n + 1$ closed subsets covering S^n such that for every $x \in S^n$ there exists a $j \in I^{n+1}$ satisfying $x_j > 0$ and $x \in C_j$. Then $\bigcap_{j=1}^{n+1} C_j \neq \emptyset$.

Proof. For $j = 1, \dots, n + 1$, define $C(+ (j + 1), -j) = C_j$, where $j + 1 = 1$ when $j = n + 1$. Moreover, $C(+i, -j) = \emptyset$ whenever $i \neq j + 1$. Then these sets satisfy the conditions of Theorem 5.1. Hence, there exist an $x^* \in S^n$ and a nonempty set $T^* \subset I^{n+1}$ such that $j \in T^*$ implies $j + 1 \in T^*$ and

$x^* \in C(+j+1, -j)$. Hence $T^* = I^{n+1}$ and so $x^* \in C_j$ for every $j \in I^{n+1}$. $\square$

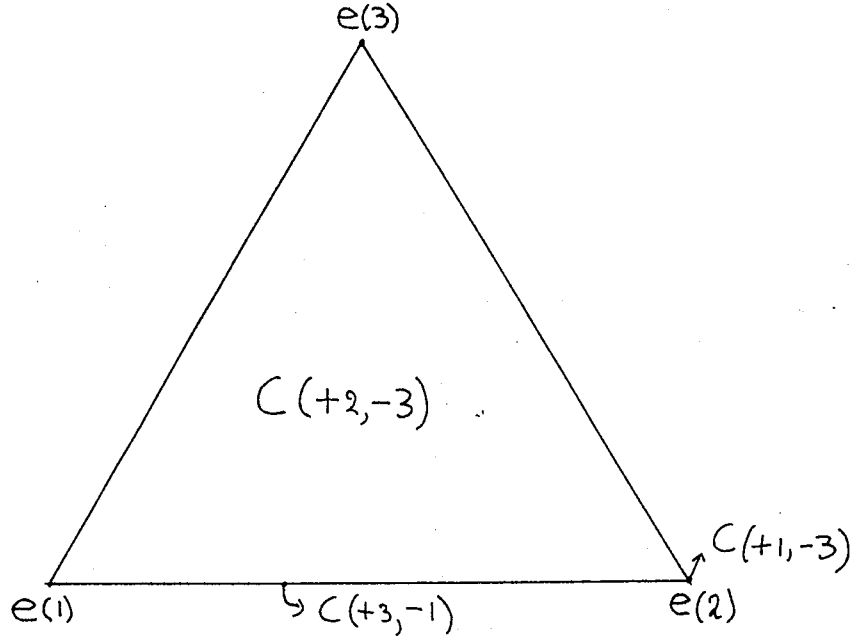


Figure 5: Illustration of the intersection theorem, $n = 2$, $T^* = \{1, 3\}$.

Similarly, the next intersection theorem follows.

Corollary 5.2 (*Freidenfelds' intersection theorem on S^n*)

Let $C_1, \dots, C_{n+1}$ be $n+1$ closed subsets covering S^n such that $x \in C_j$ whenever $x_j = 0$. Then $\bigcap_{j=1}^{n+1} C_j \neq \emptyset$.

Proof. For $j = 1, \dots, n+1$, define $C(+j, -(j+1)) = C_j$, where $j+1 = 1$ when $j = n+1$. For $i \neq j+1$, let $C(+j, -i)$ be empty. We will first show that these sets satisfy the conditions of Theorem 5.1. Let $x \in \text{bd } S^n$. Then there exists a $j \in I^{n+1}$ such that $x_j = 0$ and $x_{j+1} > 0$. Since $x_j = 0$ we have that $x \in C_j$. Hence, $x \in C(+j, -(j+1))$ whereas $x_{j+1} > 0$. According to Theorem 5.1 there exist an $x^* \in S^n$ and a nonempty T^* such that $j \in T^*$ implies $j+1 \in T^*$ and $x^* \in C(+j, -(j+1))$. Hence $T^* = I^{n+1}$ and so

$x^* \in C_j$ for every $j \in I^{n+1}$. □

Now, let $z : S^n \rightarrow \mathbb{R}^{n+1}$ be a continuous excess demand function. For i and $j \in I^{n+1}$, let us define $C(+i, -j) = \{x \in S^n | z_i(x) = \max_h z_h(x) \text{ and } z_j(x) = \min_h z_h(x)\}$. Since z is continuous, $p^T z(p) = 0$, and $z_h(x) \geq 0$ whenever $x_h = 0$, these sets satisfy the conditions of Theorem 5.1. Hence there exist an x^* and a nonempty $T^* \subset I^{n+1}$, such that $i \in T^*$ implies there exist a j satisfying $x^* \in C(+i, -j)$ and a k satisfying $x^* \in C(+k, -i)$. Consequently, $z_i(x^*) = \max_h z_h(x^*)$ and $z_i(x^*) = \min_h z_h(x^*)$, hence $z(x^*) = 0$. Any intersection point obtained in this manner is therefore a zero point of z .

REFERENCES

- C. Dang and A.J.J. Talman (1990), "The D_1 -triangulation in simplicial variable dimension algorithms on the unit simplex for computing fixed points", Discussion paper 9050, Center for Economic Research, Tilburg University, The Netherlands.
- T.M. Doup (1988), *Simplicial Algorithms on The Simplotope*. Lecture Notes in Economics and Mathematical Systems 318, Springer-Verlag, Berlin.
- T.M. Doup, G. van der Laan and A.J.J. Talman (1987), "The $(2^{n+1} - 2)$ -ray algorithm: a new simplicial algorithm to compute economic equilibria", *Mathematical Programming* 39, pp. 241-252.
- T.M. Doup and A.J.J. Talman (1987a), "A new simplicial variable dimension algorithm to find equilibria on the product space of unit simplices", *Mathematical Programming* 37, pp. 319-355.
- T.M. Doup and A.J.J. Talman (1987b), "The 2-ray algorithm for solving equilibrium problems on the unit simplex", *Methods of Operations Research* 57, pp. 269-285.
- J. Freidenfelds (1974), "A set intersection theorem and applications", *Mathematical Programming* 7, pp. 199-211.
- B. Knaster, C. Kuratowski, and S. Mazurkiewicz (1929), "Ein Beweis des Fixpunktsatzes für n -dimensionale Simplexe", *Fundamenta Mathematicae* 14, pp. 132-137.

- G. van der Laan and A.J.J. Talman (1979), "A restart algorithm for computing fixed points without an extra dimension", *Mathematical Programming* 17, pp. 74-84.
- H. Scarf (1967), "The approximation of fixed points of a continuous mapping", *SIAM Journal of Applied Mathematics* 15, pp. 1328-1343.
- H. Scarf (1973) (with the collaboration of T. Hansen), *The Computation of Economic Equilibria*, Yale University Press, New Haven, CT, U.S.A.
- L.S. Shapley (1980), "On balanced games without side payments", in: *Mathematical Programming*, T.C. Hu and S.M. Robinson, eds., Academic Press, New York, N.Y., U.S.A., pp. 261-290.
- E. Sperner (1928), "Neuer Beweis für die Invarianz der Dimensionszahl und des Gebietes", *Abh. Math. Sem. Univ. Hamburg* 6, pp. 265-272.
- A.J.J. Talman (1991), "Intersection theorems on the unit simplex and the simplotope", Discussion Paper 9157, CentER for Economic Research, Tilburg University, Tilburg, The Netherlands.