Discrete Applied Mathematics 42 (1993) 279–290
North-Holland

A class of generalized greedy algorithms for the multi-knapsack problem

A.H.G. Rinnooy Kan*

Econometric Institute, Erasmus University, Rotterdam, Netherlands

L. Stougie

Instituut voor Actuariaat en Econometrie, University of Amsterdam, Amsterdam, Netherlands

C. Vercellis**

Dipartimento di Economia e Produzione, Politecnico di Milano, Milano, Italy

Received 15 June 1990
Revised 1 May 1991

Abstract

Rinnooy Kan, A.H.G., L. Stougie and C. Vercellis, A class of generalized greedy algorithms for the multi-knapsack problem, Discrete Applied Mathematics 42 (1993) 279–290.

A class of generalized greedy algorithms is proposed for the solution of the {0,1} multi-knapsack problem. Items are selected according to decreasing ratios of their profit and a weighted sum of their requirement coefficients. The solution obtained depends on the choice of the weights. A geometrical representation of the method is given and the relation to the dual of the linear programming relaxation of multi-knapsack is exploited. We investigate the complexity of computing a set of weights that gives the maximum greedy solution value. Finally, the heuristics are subjected to both a worst-case and a probabilistic performance analysis.

Keywords. Multi-knapsack problem, greedy heuristic, linear programming relaxation, computational complexity, worst-case analysis, probabilistic analysis.

* Partially supported by NSF Grant ECS-83-16224.
** Partially supported by MPI Research Project "Models and algorithms for Optimization".
Correspondence to: Professor C. Vercellis, Dipartimento di Economia e Produzione, Politecnico di Milano, Piazza Leonardo da Vinci, 32, 20133 Milano, Italy.

1. Introduction

We are given a set of n items and m knapsack constraints. Each item j $(j=1,2,\dots,n)$ requires a_{ij} units of space in the ith knapsack $(i=1,2,\dots,m)$ and yields c_j units of profit upon inclusion. The capacities of the knapsacks are given by $(b_1, b_2, \dots, b_m)$. Maximization of profit subject to the knapsack restrictions leads to the following $\{0,1\}$ integer linear programming problem:

$$\begin{aligned} \max \quad & \sum_{j=1} c_j x_j, \\ \text{s.t.} \quad & \sum_{j=1}^{n} a_{ij} x_j \le b_i \quad (i = 1,2,\dots,m), \\ & x_j \in \{0,1\} \qquad (j = 1,2,\dots,n). \end{aligned} \tag{1}$$

By its nature, all parameters of this multi-knapsack problem are nonnegative. It has applications in the area of scheduling and capital budgeting [7]. The problem is unary NP-hard [3]. A fully polynomial approximation scheme based on dynamic programming techniques has been proposed in [2]. A probabilistic characterization of its value function is described in [10].

In this paper we consider a class of generalized greedy heuristics that yield approximate solutions. Given a set of weights $(w_1, w_2, \dots, w_m)$ the items are ordered according to decreasing ratio

$$\varrho_j = \frac{c_j}{\sum_{i=1}^{m} w_i a_{ij}}.$$

Items are considered for inclusion in the knapsack in this order. An item is omitted if inclusion would lead to a violation of any of the constraints. The procedure terminates after having considered the last item in the list. The solution obtained is denoted by $x^G(w_1, w_2, \dots, w_m) = (x_1^G(w_1, w_2, \dots, w_m), x_2^G(w_1, w_2, \dots, w_m), \dots, x_n^G(w_1, w_2, \dots, w_m))$ and its value by $z^G(w_1, w_2, \dots, w_m)$. Obviously, the quality of the approximate solution is affected by the choice of the weights. The above generalizes the greedy heuristic for the multi-knapsack problem. In [1] a weighted version of the greedy heuristic is proposed for covering problems, restricting the choice of the weights to $w_i = 1/b_i$.

We will analyze the performance of the generalized greedy heuristic as a function of the weights. First, in Section 2, we give a geometrical representation of its behaviour and relate this to the dual of the linear programming relaxation of the multi-knapsack problem. In particular, this shows that a choice of weights can be made such that the greedy solution is at least as good as the integer round-down of the LP-relaxation. In Section 3 the computational complexity of determining an optimal set of weights is investigated; i.e., a set of weights $(w_1^0, w_2^0, \dots, w_m^0)$ such that

$$z^G(w_1^0, w_2^0, \dots, w_m^0) = \max\{z^G(w_1, w_2, \dots, w_m):\ w_i \ge 0\ (i=1,2,\dots,m)\}. \tag{2}$$

The performance quality of the generalized greedy heuristic is analyzed from a worst-case point of view in Section 4 and from a probabilistic one in Section 5.

2. Graphical interpretation and LP-relaxation

For a graphical interpretation of the performance of the generalized greedy algorithm we let each decision variable x_j $(j=1,2,\dots,n)$ correspond to a point in $\mathbb{R}_+^m$, given by the coordinates $(a_{1j}/c_j, a_{2j}/c_j, \dots, a_{mj}/c_j)$ and rewrite the ratio ϱ_j as

$$\varrho_j = \frac{1}{\sum_{i=1}^m w_i(a_{ij}/c_j)}.$$

For each choice of the weights, the selection procedure of the generalized greedy heuristic can be regarded as moving upwards an $(m-1)$-dimensional hyperplane with normal vector $(w_1, w_2, \dots, w_m)$, starting from the origin, and considering items for inclusion in the knapsack in the order in which the hyperplane passes their corresponding points. In Fig. 1 this is visualized for $m=2$.

We will show now how the dual of the LP-relaxation of the multi-knapsack problem fits into this picture. The LP-relaxation is obtained from the integer programming formulation (1) by substituting the constraints $x_j \in \{0,1\}$ by $0 \le x_j \le 1$ $(j=1,2,\dots,n)$. Its dual is given by

$$\begin{aligned}
\min \quad & \sum_{i=1}^{m} \lambda_i b_i + \sum_{j=1}^{n} v_j, \\
\text{s.t.} \quad & v_j + \sum_{i=1}^{m} \lambda_i a_{ij} \ge c_j \quad (j=1,2,\dots,n), \\
& v_j \ge 0 \ (j=1,2,\dots,n), \ \lambda_i \ge 0 \ (i=1,2,\dots,m).
\end{aligned} \tag{3}$$

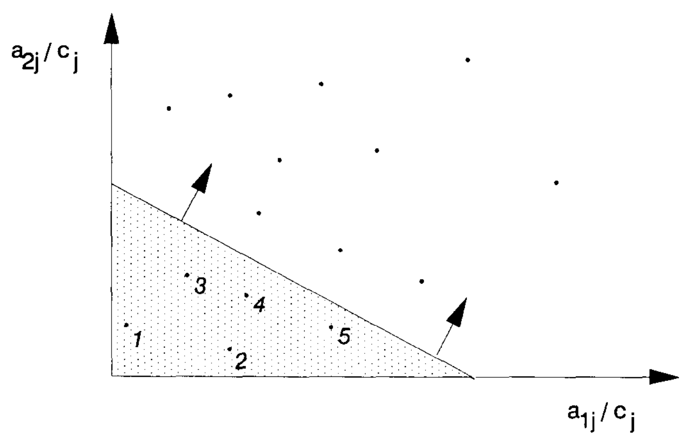

Fig. 1. The greedy algorithms picking up items.

Consider m points $\{j_1, j_2, \ldots, j_m\}$, among the n represented in Fig. 1, such that the hyperplane through these points has positive normal vector $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$. The vector λ is determined by solving the system of linear equations

$$\sum_{i=1}^{m} \lambda_i a_{ij_k} = c_{jk} \quad (k = 1, 2, \ldots, m).$$

It is easy to see that λ is part of a basic feasible solution (λ, v) of the dual problem (3), where the values of $v = (v_1, v_2, \ldots, v_n)$ are given by

$$v_j = \max\{0, c_j - \sum_{i=1}^{m} \lambda_i a_{ij}\} \quad (j = 1, 2, \ldots, n).$$

To this dual basic feasible solution corresponds a primal solution $(x_1, x_2, \ldots, x_n)$, which is in general not feasible. We have that $x_j = 1$ if $v_j \geq 0$, i.e., if the corresponding point $(a_{1j}/c_j, a_{2j}/c_j, \ldots, a_{mj}/c_j)$ lies below the hyperplane through the points $\{j_1, j_2, \ldots, j_m\}$. We have instead $x_j = 0$ if $v_j = 0$ and $\sum_{i=1}^{m} \lambda_i a_{ij} > c_j$, or, stated otherwise, if the corresponding point lies not below the given hyperplane. The values $(x_{j_1}, x_{j_2}, \ldots, x_{j_m})$ are obtained by solving the system of equations

$$b_i - \sum_{j \notin \{j_1, j_2, \ldots, j_m\}} a_{ij} x_j = \sum_{k=1}^{m} a_{ij_k} x_{j_k} \quad (i = 1, 2, \ldots, m).$$

In nondegenerate situations the latter values are fractional. Let us compare the primal solution with the generalized greedy solution when weights are chosen as $w_1 = \lambda_1$, $w_2 = \lambda_2, \ldots, w_m = \lambda_m$. The greedy solution is then obtained by moving up the hyperplane with normal vector λ starting in the origin. The primal solution may be unfeasible because $x_{j_k} < 0$ for some k. This occurs when at least one item whose corresponding point lies not above the hyperplane through $\{j_1, j_2, \ldots, j_m\}$ cannot be included in the knapsack. In this case, the greedy algorithm will not select all items with $x_j = 1$, i.e., not all items with $\varrho_j > 1$. The other possible infeasibility is due to condition $x_{j_k} > 1$ for some k, the situation when the items below the hyperplane leave enough space to include at least item j_k as well. In this case the greedy algorithm will select next to all items with $x_j = 1$ ($\varrho_j > 1$) other items with $\varrho_j \leq 1$.

There exists a set of m points that corresponds to the optimal dual solution $(\lambda, v) = (\lambda^*, v^*)$ and therefore with the optimal primal solution $x^* = (x_1^*, x_2^*, \ldots, x_n^*)$, assuming uniqueness and nondegeneracy. The generalized greedy algorithm with weights $w_1 = \lambda_1^*, w_2 = \lambda_2^*, \ldots, w_m = \lambda_m^*$ will select all items with $x_j^* = 1$. It may also select extra items with $\varrho_j \leq 1$ if they come from a combination of low requirement coefficients and low profits. Thus, a comparison of this solution with the integer round-down of the optimal LP-relaxation solution, given by $x_j^{\lfloor \mathrm{LP} \rfloor} = 1$ if $x_j^* = 1$ and $x_j^{\lfloor \mathrm{LP} \rfloor} = 0$ otherwise, leads to the following lemma.

Lemma 2.1. *If x^* is the unique and nondegenerate optimal solution of the* LP-*relaxation of* (1), *then*

(i) $x_j^{\lfloor \mathrm{LP} \rfloor} = 1 \Rightarrow x_j^G(\lambda_1, \lambda_2, \ldots, \lambda_m) = 1$,

(ii) $z^G(\lambda_1, \lambda_2, \ldots, \lambda_m) \geq z^{\lfloor \mathrm{LP} \rfloor}$.

The following example illustrates that $(\lambda_1^*, \lambda_2^*, \ldots, \lambda_m^*)$ is not the optimal set of weights for the generalized greedy algorithm.

Example 2.2. Consider the problem

$$\begin{aligned} \max \quad & 3x_1 + 6x_2 + 6x_3, \\ \text{s.t.} \quad & 6x_1 + 6x_2 + 24x_3 \leq 26, \\ & 6x_1 + 24x_2 + 6x_3 \leq 26, \\ & x_1, x_2, x_3 \in \{0, 1\}. \end{aligned}$$

The optimal solution of its linear programming relaxation is given by $x_1^* = 1$, $x_2^* = x_3^* = 2/3$. This solution is nondegenerate and unique. The optimal dual multipliers are $\lambda_1^* = \lambda_2^* = 1/5$. We have $x_1^G(\lambda_1^*, \lambda_2^*) = 1$, $x_2^G(\lambda_1^*, \lambda_2^*) = x_3^G(\lambda_1^*, \lambda_2^*) = 0$, with objective value $z^G(\lambda_1^*, \lambda_2^*) = 3$. Obviously, it is better to set either $x_2 = 1$ or $x_3 = 1$. This is accomplished by the greedy algorithm with weights $w_1 = 1$, $w_2 = 0$. For this choice we have $\varrho_1 = 1/2$, $\varrho_2 = 1$, $\varrho_3 = 1/4$. This leads to the greedy solution $x_1^G(1,0) = 0$, $x_2^G(1,0) = 1$, $x_3^G(1,0) = 0$, with value $z^G(1,0) = 6$. In this specific example the latter solution is optimal, but we will see in the sequel that the best greedy solution is not necessarily optimal.

3. Computational complexity of finding optimal weights

To study the computational complexity of determining the optimal set of weights $(w_1^0, w_2^0, \ldots, w_m^0)$, in the sense of (2), we distinguish between two situations. In the first situation we consider the multi-knapsack problem with a fixed number m of constraints. The following theorem states that the problem is well solved in this case. Its proof, based on the graphical insight from the previous section, is constructive.

Theorem 3.1. *If the multi-knapsack problem has a fixed number m of constraints, then there exists a polynomial time algorithm to determine an optimal set of weights $(w_1^0, w_2^0, \ldots, w_m^0)$ for the generalized greedy algorithm.*

Proof. Recall the graphical interpretation of the generalized greedy algorithm. Each set of weights induces an ordering of the points (items), according to which they are considered for inclusion in the knapsack. To facilitate the exposition, let us consider the case $m = 2$. If we set $w_1 = 1$ and let w_2 increase continuously starting from 0, then at a certain value of w_2 two consecutive points in the ordering induced by the starting value exchange their places. This occurs when the slope $-1/w_2$ passes the slope of the line through these two points (see Fig. 2).

There are at most $\binom{n}{2}$ such lines in the graph induced by n points, and consequently at most $\binom{n}{2}$ of these changes in the ordering (we need only consider lines with negative slopes). For each order which can be obtained, the generalized greedy algorithm requires $O(n)$ time to compute the corresponding solution. An algorithm

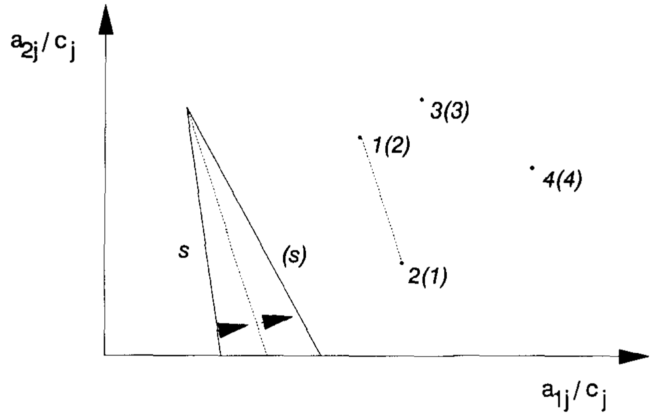

Fig. 2. A change of slope implies a change of order among points to be picked up.

devised in [6] can be used to determine all the possible orderings in $O(n^2 \log n)$ time. Selecting the solution with the maximum profit takes $O(\log n)$ time and yields an optimal set of weights. The total amount of time required is therefore $O(n^3 \log n)$.

For the m-dimensional problem we have to shift with $(m-1)$-dimensional hyperplanes. The number of possible exchanges is here $O(n^m)$. They can be determined in $O(n^m \log n)$ time and hence the above algorithm has a running time of $O(n^{m+1} \log n)$, which is polynomial in n for fixed m. □

In the second situation the number of constraints depends on the number of available items. When $m \geq n+1$ the problem of determining an optimal set of weights is essentially more difficult to solve than in the case of a fixed number of constraints.

Theorem 3.2. *If the multi-knapsack problem has a number m of constraints with $m \geq n+1$, then the problem of determining an optimal set of weights for the generalized greedy algorithm is* NP*-hard.*

Proof. Consider the recognition version of our problem:

WEIGHT SELECTION. Given an n-dimensional $\{0,1\}$ vector of variables x, an $m \times n$ matrix A of nonnegative integers, a vector $b \in \mathbb{Z}_+^n$, a vector $c \in \mathbb{Z}_+^n$ and a constant $K>0$, is there a set of weights $w_1, w_2, \ldots, w_m$ such that the generalized greedy algorithm yields a value assignment to x with $cx \geq K$ and $Ax \leq b$?

Obviously this problem belongs to NP. To show that it is NP-complete, we prove that the knapsack problem [3] can be reduced to it.

KNAPSACK. Given a set of $\{0,1\}$ variables $x_1, x_2, \ldots, x_n$, a set of positive integers $a_1, a_2, \ldots, a_n$, b and a constant $K>b$, is there a value assignment to the variables such that $K \leq \sum_{j=1}^n a_j x_j \leq b$?

Given any instance of KNAPSACK construct an instance of WEIGHT SELECTION by defining $c_j = a_j$ $(j = 1, 2, \dots, n)$, and adding a set of n redundant restrictions of the form $a_j x_j \le h_j$, with $h_j > a_j$ $(j = 1, 2, \dots, n)$. With the restriction $\sum_{j=1}^{n} a_j x_j \le b$ we associate a weight w_0. With a redundant restriction $a_j x_j \le h_j$ we associate a weight w_j $(j = 1, 2, \dots, n)$. Then, the ratios ϱ_j are given by $\varrho_j = 1/(w_0 + w_j)$ $(j = 1, 2, \dots, n)$. It is easy to see that KNAPSACK yields a positive answer if and only if weights $w_0, w_1, \dots, w_m$ can be chosen such that the corresponding generalized greedy solution has a value $\sum_{j=1}^{n} a_j x_j \ge K$ that satisfies also $\sum_{j=1}^{n} a_j x_j \le b$. □

It is still an open question to which complexity class the WEIGHT SELECTION problem belongs when m is a function of n growing slower than $n+1$.

4. Worst-case error analysis

In Section 2 we showed that the generalized greedy algorithm with an appropriately chosen set of weights produces a solution which is at least as good as the solution obtained by rounding down the optimal solution of the LP-relaxation. Now, we will show that in the worst case the two solutions are of equal quality, even if an optimal set of weights for the greedy algorithm is used.

An upper bound on the worst-case absolute difference between the optimal solution value z^I of the multi-knapsack problem and the solution value $z^{\lfloor \mathrm{LP} \rfloor}$ of the integer round-down of its LP-relaxation is $mc_{\max}$, where $c_{\max} = \max_{j=1,2,\dots,n}\{c_j\}$. This is easily seen, since the optimal solution of the LP-relaxation contains at most m fractional values (see Section 2). Rounding down these values implies that the optimal value z^{LP} of the linear program is decreased by at most $mc_{\max}$, i.e., $z^{\mathrm{LP}} - z^{\lfloor \mathrm{LP} \rfloor} < mc_{\max}$. Since $z^I \le z^{\mathrm{LP}}$, we have that $z^I - z^{\lfloor \mathrm{LP} \rfloor} < mc_{\max}$. We know from Lemma 2.1 that $z^G(w_1^0, w_2^0, \dots, w_m^0) \ge z^{\lfloor \mathrm{LP} \rfloor}$. Hence, $z^I - z^G(w_1^0, w_2^0, \dots, w_m^0) < mc_{\max}$. The following example shows that this bound is sharp.

Example 4.1. For any ε with $0 < \varepsilon < c$ we choose $\delta < \varepsilon / c$ and consider the multi-knapsack instance:

$$\begin{aligned} \max \quad & cx_1 + cx_2 + \cdots + cx_m + \varepsilon x_{m+1} + \varepsilon x_{m+2} + \cdots + \varepsilon x_{2m}, \\ \text{s.t.} \quad & x_i + \delta x_{m+i} \le 1 \quad (i = 1, 2, \dots, m), \\ & x_j \in \{0, 1\} \qquad (j = 1, 2, \dots, n). \end{aligned}$$

Obviously, the optimal solution of this problem is given by $x_1 = x_2 = \cdots = x_m = 1$ and $x_{m+1} = x_{m+2} = \cdots = x_{2m} = 0$, and has value $z^I = mc$. For a given set of weights $(w_1, w_2, \dots, w_m)$ the ratios for the items are given by

$$\varrho_i = \frac{c}{w_i}, \quad \varrho_{m+i} = \frac{\varepsilon}{\delta w_i} > \frac{c}{w_i} \qquad (i = 1, 2, \dots, m).$$

Therefore, for any set of weights, the generalized greedy algorithm selects item $m+i$ $(i=1,2,\dots,m)$ which occupies too much space in the knapsack constraint i to include item i as well. Thus, also for an optimal set of weights, we have $z^G(w_1^0, w_2^0, \dots, w_m^0) = m\varepsilon$, and hence

$$z^I - z^G(w_1^0, w_2^0, \dots, w_m^0) = m(c-\varepsilon).$$

Since ε was chosen arbitrarily, $mc = mc_{\max}$ is a bound on the worst-case absolute error of the generalized greedy heuristic that can be approximated as much closely as desired.

5. Probabilistic error analysis

We will show in this section that there exists a choice of the weights $(w_1, w_2, \dots, w_m)$ for which the resulting generalized greedy algorithm is asymptotically optimal with probability one (wp1) as n grows to infinity, in terms of the relative error.

In order to develop our analysis, we have to specify a probabilistic model of the multi-knapsack problem. According to the assumptions made in [10], which are a generalization of those made by several authors dealing with stochastic knapsack problems [2,8,9], the profit coefficients $\boldsymbol{c}_j$ $(j=1,2,\dots,n)$ are nonnegative independent identically distributed (i.i.d.) random variables with common continuous distribution function F, defined and strictly increasing over a nonsingular support $[\mu, \nu]$, with a bounded density function f and a finite expectation $E[\boldsymbol{c}_1]$. Furthermore, letting $\boldsymbol{a}_j = (a_{1j}, a_{2j}, \dots, a_{mj})$ $(j=1,2,\dots,n)$, we assume that the $\boldsymbol{a}_j$ $(j=1,2,\dots,n)$ are i.i.d. random vectors, with a density function with positive density over a convex and open set, a finite mean $E[\boldsymbol{a}_1]$, satisfying also $E[\boldsymbol{a}_1\boldsymbol{a}_1^{\mathrm{T}}] < \infty$. Finally, we assume that the capacities b_i $(i=1,2,\dots,m)$ grow proportionally with the number of items, i.e., $b_i = n\beta_i$ $(i=1,2,\dots,m)$ for a set of parameters $\beta_i \in V = \{\beta\colon 0<\beta<E[\boldsymbol{a}_1]\}$, in order to have a proper fraction of the n items included into the knapsack (cf. [8,10]). Notice, for example, that the uniform distribution of the $\boldsymbol{c}_j$ over $[0,1]$ and of the $\boldsymbol{a}_j$ over the m-dimensional unit hypercube do satisfy the previous assumptions.

In [10] the optimum value of the multi-knapsack problem as a function of the right-hand side vector b has been characterized in terms of almost sure convergence as n tends to infinity, under the previous stochastic assumptions. To this end, for any given m-dimensional vector $\lambda \geq 0$, define the following set of random variables

$$\boldsymbol{x}_j^L(\lambda) = \begin{cases} 1, & \text{if } c_j - \sum_{i=1}^m \lambda_i a_{ij} \geq 0 \\ 0, & \text{otherwise} \end{cases} \qquad (j=1,2,\dots,n)$$

and the function

$$L(\lambda) = \sum_{i=1}^m \lambda_i \beta_i + E[\boldsymbol{c}_1 \boldsymbol{x}_1^L(\lambda)] - \lambda^{\mathrm{T}} E[\boldsymbol{a}_1 \boldsymbol{x}_1^L(\lambda)].$$

Let also the random vector λ_n^* denote the optimal dual multipliers, as defined in Section 2, for an instance of the multi-knapsack problem whose coefficients are distributed according to the previous stochastic assumptions. Here the subscript n is introduced to express the dependence of the vector of optimal dual multipliers λ^* from the number of items in the instance of multi-knapsack. Let also z_n^I be the random variable denoting the optimum solution value of the multi-knapsack problem with n items.

The subsequent Lemma 5.1 summarizes the results contained in [10]:

Lemma 5.1. *Let the size and profit coefficients of the multi-knapsack problem be distributed according to the previous stochastic assumptions. Then*

(i) *the function $L(\lambda)$ has a unique minimum, denoted as $\lambda^* \neq 0$;*

(ii) *λ_i^* satisfies the following conditions for $i = 1, 2, \ldots, m$:*

$$\lambda_i^*(\beta_i - E[a_{i1} x_1^L(\lambda^*)]) = 0,$$

$$E[a_{i1} x_1^L(\lambda^*)] \leq \beta_i;$$

(iii) *the sequence of random vectors $\{\lambda_n^*\}$ converges with probability one to λ^*;*

(iv) *the sequence of normalized optimum values $\{n^{-1} z_n^I\}$ converges with probability one to $L(\lambda)$.*

The following lemma, due to Hoeffding [4], will be needed in the sequel:

Lemma 5.2. *Let $X_1, X_2, \ldots, X_n$ be independent random variables taking values in the interval* $[0, 1]$. *Then, for* $0 < t < 1 - n^{-1} E[\sum_{j=1}^n X_j]$, *we have*:

$$\Pr\left\{\frac{1}{n}\left(\sum_{j=1}^n X_j - E\left[\sum_{j=1}^n X_j\right]\right) \geq t\right\} \leq e^{-2nt^2},$$

$$\Pr\left\{\frac{1}{n}\left(\sum_{j=1}^n X_j - E\left[\sum_{j=1}^n X_j\right]\right) \leq -t\right\} \leq e^{-2nt^2},$$

$$\Pr\left\{\frac{1}{n}\left|\sum_{j=1}^n X_j - E\left[\sum_{j=1}^n X_j\right]\right| \geq t\right\} \leq 2e^{-2nt^2}.$$

We know from Section 2 that the generalized greedy algorithm which corresponds to the choice $w = \lambda_n^*$ is asymptotically optimal in a deterministic sense, if m is fixed. Unfortunately, the vector λ_n^* depends on the particular problem instance, requiring to solve the dual of the relaxed multi-knapsack problem. However, in light of Lemma 5.1, one should expect that the choice of the weights $w = \lambda^*$, where λ^* is the unique minimum of $L(\lambda)$ and therefore is independent from the particular problem instance, might lead to an asymptotically optimal generalized greedy algorithm, at least in a probabilistic sense. In fact, we will show that the generalized greedy algorithm corresponding to the choice $w = \lambda^*$ is asymptotically optimal with probability one. Actually, we will prove the result for a simplified version of the generalized greedy algorithm, in which the inclusion of any further item in the knapsack is disallowed as soon as the first item is met which violates any of the capacity constraints. Obviously, for any given choice of the weights, this simplified version of the algorithm produces solution values which are not better than the corresponding ones generated by means of the generalized greedy algorithm in its original form. Define

$$\xi_n = \inf\{\xi \geq 0: \sum_{j=1}^{n} a_{ij} x_j^L(\xi \lambda_n^*) \leq b_i \ (i=1,2,\ldots,m)\}.$$

The sequence ξ_n can be interpreted as the value of the ratio ϱ_j corresponding to the last item included in the knapsack by the simplified greedy algorithm with weights λ_n^*.

We start by showing that

Lemma 5.3. *The sequence* $\{\xi_n\}$ *converges to* 1 *with probability one.*

Proof. We have, for any given $\varepsilon > 0$, that

$$\Pr\{|\xi_n - 1| > \varepsilon\} = \Pr\{\xi_n > 1+\varepsilon\} + \Pr\{\xi_n < 1-\varepsilon\}. \tag{4}$$

The first term of the sum in (4) can be bounded as follows:

$$\Pr\{\xi_n > 1+\varepsilon\} = \Pr\left\{\exists k: \sum_{j=1}^{n} a_{ij} x_j^L(\lambda^* + \varepsilon\lambda^*) > b_k\right\}$$
$$\leq \sum_{i=1}^{m} Pr\left\{\sum_{j=1}^{n} a_{ij} x_j^L(\lambda^* + \varepsilon\lambda^*) > b_i\right\}.$$

We know from Lemma 5.1 that $E[a_{i1} x_1^L(\lambda^*)] \leq \beta_i \ (i=1,2,\ldots,m)$, so that $E[a_{i1} x_1^L(\lambda^* + \varepsilon\lambda^*)] < \beta_i \ (i=1,2,\ldots,m)$, as the function $E[a_{i1} x_1^L(\lambda)]$ is strictly increasing in λ. Thus,

$$\Pr\{\xi_n > 1+\varepsilon\} = \sum_{i=1}^{m} Pr\left\{\sum_{j=1}^{n} a_{ij} x_j^L(\lambda^* + \varepsilon\lambda^*) - nE[a_{i1} x_1^L(\lambda^* + \varepsilon\lambda^*)]\right.$$
$$\left. > n(\beta_i - E[a_{i1} x_1^L(\lambda^* + \varepsilon\lambda^*)])\right\}$$
$$\leq \sum_{i=1}^{n} \mathrm{e}^{-2n(\beta_i - E[a_{i1} x_1^L(\lambda^* + \varepsilon\lambda^*)])^2}, \tag{5}$$

where the last inequality follows from application of Lemma 5.2.

To bound the second probability in (4) we proceed as follows. We know from Lemma 5.2 that $\lambda_i^*(\beta_i - E[a_{i1} x_1^L(\lambda^*)]) = 0$; since $\lambda^* \neq 0$, this implies that there exists at least one constraint, say the kth, for which $E[a_{k1} x_1^L(\lambda^*)] = \beta_k$. Furthermore, $E[a_{k1} x_1^L(\lambda^* - \varepsilon\lambda^*)] > \beta_k$. We have therefore

$$Pr\{\xi_n < 1-\varepsilon\} = \Pr\left\{\sum_{i=1}^{n} a_{ij} x_j^L(\lambda^* - \varepsilon\lambda^*) < b_i \ (i=1,2,\ldots,m)\right\}$$

$$\le \Pr\left\{\sum_{i=1}^{n} a_{kj} x_j^L(\lambda^* - \varepsilon\lambda^*) < b_k\right\}$$

$$= \Pr\left\{\sum_{i=1}^{n} a_{kj} x_j^L(\lambda^* - \varepsilon\lambda^*) - nE[a_{k1} x_1^L(\lambda^* - \varepsilon\lambda^*)] \right.$$
$$\left. < n(\beta_k - E[a_{k1} x_1^L(\lambda^* - \varepsilon\lambda^*)])\right\}$$
$$< \mathrm{e}^{-2n(\beta_k - E[a_{k1} x_1^L(\lambda^* - \varepsilon\lambda^*)])^2}, \tag{6}$$

where, again, the last inequality follows from Lemma 5.2.

Combining (5) and (6) shows that the two terms in the sum (4) are bounded by the general terms of two convergent series, thus leading to the required result. □

Let $z_n^H(\lambda^*)$ be the random variable denoting the solution value determined by the simplified version of the generalized greedy algorithm of an instance of the multi-knapsack problem with n items. We are now ready to prove the main result of this section.

Theorem 5.4. *The sequence* $\{z_n^H(\lambda^*)/z_n^I\}$ *converges to* 1 *with probability one.*

Proof. Since we know from Lemma 5.1 that $\{n^{-1}z_n^I\} \xrightarrow{\text{wp1}} L(\lambda^*)$, it is sufficient to prove that $\{n^{-1}z_n^H(\lambda^*)\} \xrightarrow{\text{wp1}} L(\lambda^*)$.

By definition of ξ_n, we have

$$z_n^H(\lambda^*) = \sum_{j=1}^{n} c_j x_j^L(\xi_n\lambda^*).$$

Hence, for any $\varepsilon > 0$,

$$\Pr\left\{\left|\frac{1}{n} z_n^H(\lambda^*) - L(\lambda^*)\right| > \varepsilon\right\}$$

$$= Pr\left\{\left|\frac{1}{n}\sum_{j=1}^{n} c_j x_j^L(\xi_n\lambda^*) - L(\lambda^*)\right| > \varepsilon\right\}$$

$$= \Pr\left\{\left|\frac{1}{n}\sum_{j=1}^{n} c_j x_j^L(\xi_n\lambda^*) - E[c_1 x_1^L(\xi_n\lambda^*)]\right| > \frac{\varepsilon}{2}\right\}$$
$$+ \Pr\left\{|E[c_1 x_1^L(\xi_n\lambda^*)] - L(\lambda^*)| > \frac{\varepsilon}{2}\right\}. \tag{7}$$

By the continuity of $x_1^L(\xi\lambda^*)$ with respect to ξ, and by Lemma 5.3, it follows that $c_1 x_1^L(\xi_n\lambda^*) \xrightarrow{\text{wp1}} c_1 x_1^L(\lambda^*)$. This implies also, due to the uniform boundedness of the sequence $\{c_1 x_1^L(\xi_n\lambda^*)\}$, that convergence in expectation holds, i.e., $E[c_1 x_1^L(\xi_n\lambda^*)] \to L(\lambda^*)$. This means that, for every $n > n_0 = n_0(\varepsilon/2)$, the following inequality holds:

$$|E[c_1 x_1^L(\xi_n\lambda^*)] - L(\lambda^*)| < \frac{\varepsilon}{2}. \tag{8}$$

By substituting (8) in (7) and by denoting as F_{ξ_n} the distribution function of ξ_n, we obtain

$$\begin{aligned}
&\Pr\left\{\left|\frac{1}{n} z_n^H(\lambda^*) - L(\lambda^*)\right| < \varepsilon\right\} \\
&\le \Pr\left\{\left|\frac{1}{n}\sum_{j=1}^{n} c_j x_j^L(\xi_n\lambda^*) - E[c_1 x_1^L(\xi_n\lambda^*)]\right| < \frac{\varepsilon}{2}\right\} \\
&= \int \Pr\left\{\left|\frac{1}{n}\sum_{j=1}^{n} c_j x_j^L(\xi\lambda^*) - E[c_1 x_1^L(\xi\lambda^*)]\right| < \frac{\varepsilon}{2}\right\} \mathrm{d}F_{\xi_n},
\end{aligned} \tag{9}$$

the last inequality in (9) being derived from the application of Lemma 5.2. As the last term in (9) is the general term of a convergent series, the Borel–Cantelli lemma leads to the required result. □

Acknowledgement

The authors wish to express their thanks to J.K.L. Lenstra and M. Meanti for helpful suggestions on the early stages of this research.

References

[1] G. Dobson, Worst-case analysis of greedy heuristics for integer programming with nonnegative data, Math. Oper. Res. 7 (1982) 515–531.

[2] A.M. Frieze and M.R.B. Clarke, Approximation algorithms for the m-dimensional 0, 1 knapsack problem: worst-case and probabilistic analysis, European J. Oper. Res. 15 (1984) 100–109.

[3] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness (Freeman, San Francisco, CA, 1979).

[4] W. Hoeffding, Probability inequalities for sums of bounded random variables, Amer. Statist. Assoc. J. 3 (1963) 13–30.

[5] B. Korte and R. Schrader, On the existence of fast approximation schemes, Report 80163-OR, Institut Ökonometrie und Operations Research, Rheinische F. Wilhelms Universität, Bonn (1980).

[6] A.K. Lenstra, J.K. Lenstra, A.H.G. Rinnooy Kan and T.J. Wansbeek, Two lines least squares, in: Annals of Discrete Mathematics 16 (North-Holland, Amsterdam, 1982) 201–211.

[7] J. Lorie and L. Savage, Three problems in capital rationing, J. Business 28 (1955) 229–239.

[8] G.S. Lueker, On the average difference between the solution to linear and integer knapsack problems, in: Applied Probability-Computer Science: The Interface (Birkhauser, Boston, 1982) 489–504.

[9] A. Marchetti-Spaccamela and C. Vercellis, Stochastic on-line knapsack problems, Math. Programming, to appear.

[10] M. Meanti, A.H.G. Rinnooy Kan, L. Stougie and C. Vercellis, A probabilistic analysis of the multiknapsack value function, Math. Programming 46 (1990) 237–247.