Volume 24 (1), pp. 1–15
http://www.orssa.org.za

ORiON
ISSN 0529-191-X
©2008

On the efficient solvability of a simple class of nonlinear knapsack problems

SE Visagie*

Received: 18 February 2008; Revised: 13 May 2008; Accepted: 15 May 2008

Abstract

In this paper the efficient solvability of a class of nonlinear knapsack problems are investigated by means of the problem's necessary and sufficient conditions. It is shown that, from the general theory, it is impossible to determine sufficient conditions for a solution to be globally optimal. Furthermore, it is shown that even for the smallest possible instance of this problem it is, in general, possible to have an arbitrary large number of solutions for which the necessary conditions hold. These results are then generalised to larger instances of the problem. A possible solution approach is applied to sets of randomly generated problems that utilises the necessary conditions together with the branch-and-bound technique in an attempt to limit the search space. This approach solves mixed 0/1 knapsack problems in order to find all possible solutions satisfying the necessary conditions. Due to the large number of solutions satisfying the necessary conditions the proposed solution approach takes substantially longer than existing branch-and-bound algorithms together with linear enveloping when applied to the same set of problems. This result renders the proposed approach not very efficient.

Key words: Nonlinear knapsack problem, nonconvex optimisation, resource allocation problem.

1 Introduction

Knapsack problems have a wide range of applications in, for example, marketing, chemistry, information technology, portfolio optimisation, optimal search strategies, production planning, logistics and statistical sampling [1, 2, 7, 8]. The specific problem formulation considered in this paper arises in the optimal allocation of funding to different, independent projects within a fixed budget. Depending on the expected return on each project, this problem may be formulated as a separable nonlinear knapsack problem in which the objective function consists of the sum of increasing convex and increasing concave functions, subject to a single constraint, and with lower and upper bounds on the allocation

*Department of Logistics, University of Stellenbosch, Private Bag X1, Stellenbosch, 7602, South Africa, email: svisagie@sun.ac.za

to each project. Mathematically this problem may be formulated as

$$\left.\begin{array}{lrl}\text{maximise } w = \underbrace{\sum_{i=1}^{m} f_i(x_i)}_{f(\boldsymbol{x})} + \underbrace{\sum_{j=1}^{n} g_j(y_j)}_{g(\boldsymbol{y})} & & \\ \text{subject to} & \sum_{i=1}^{m} x_i + \sum_{j=1}^{n} y_j = B & \\ & l_i^{(x)} \leq x_i \leq u_i^{(x)}, & i = 1, 2, \ldots, m \\ & l_j^{(y)} \leq y_j \leq u_j^{(y)}, & j = 1, 2, \ldots, n,\end{array}\right\} \quad (1)$$

where $f_i(x_i)$ is an increasing convex function and $g_j(y_j)$ is an increasing concave function and B is the size of the budget. Furthermore $l_i^{(x)}$ and $u_i^{(x)}$ $(i = 1, 2, \ldots, m)$ are respectively lower and upper bounds on the decision variables $x_1, x_2, \ldots, x_m$. Similarly $l_j^{(y)}$ and $u_j^{(y)}$ $(j = 1, 2, \ldots, n)$ are respectively lower and upper bounds on the decision variables $y_1, y_2, \ldots, x_n$. We assume that

$$\sum_{i=1}^{m} l_i^{(x)} + \sum_{j=1}^{n} l_j^{(y)} < B < \sum_{i=1}^{m} u_i^{(y)} + \sum_{j=1}^{n} u_j^{(y)}$$

to exclude trivial solutions. No generality is lost by setting the lower bounds of all the decision variables to zero, *i.e.* by taking $l_i^{(x)} = 0$ and $l_j^{(x)} = 0$ for all $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$, because this may always be achieved by means of a simple linear transformation [2].

The remainder of this paper is structured as follows. In §2 the smallest possible instance of the problem with $m = n = 1$ is considered, and it is shown that only necessary conditions for local optimality can be determined by means of the general theory on nonlinear programming. A generalisation to $n > 1$ and $m > 1$ is considered in §3. Some thoughts and insights on possible solution approaches are presented in §4 and these are followed up with some computational results obtained via these approaches. Finally, some conclusions, together with ideas for future study, follow in §5.

2 One convex and one concave function

The special, and smallest possible, instance of (1), where the objective function consists of only one concave and one convex function, is considered in this section because it clearly demonstrates the computational complexities in solving this problem. Let the decision variables be $x \in [0, u_x]$ and $y \in [0, u_y]$. The objective in this case is then to

$$\left.\begin{array}{ll}\text{maximise } z = f(x) + g(y) & \\ \text{subject to} & x + y = B \\ & 0 \leq x \leq u_x \\ & 0 \leq y \leq u_y.\end{array}\right\} \quad (2)$$

For the formulation in (2) the second order necessary conditions [3] require that for a point (x^*, y^*) in solution space to be a local maximum there must exist a $\lambda \in \mathbb{R}$ and $\mu_1, \mu_2, \mu_3, \mu_4 \geq 0$, such that

$$\begin{aligned} x^* + y^* &= B & (3)\\ f'(x^*) + \lambda + \mu_1 - \mu_2 &= 0 & (4)\\ g'(y^*) + \lambda + \mu_3 - \mu_4 &= 0 & (5)\\ \mu_1(x^*) &= 0 & (6)\\ \mu_2(u_x - x^*) &= 0 & (7)\\ \mu_3(y^*) &= 0 & (8)\\ \mu_4(u_y - y^*) &= 0 & (9) \end{aligned}$$

and, with $F(x) = \frac{\partial^2 f(x)}{\partial x^2}$ and $G(y) = \frac{\partial^2 g(y)}{\partial y^2}$, the Lagrangian matrix

$$L(x^*, y^*) = \begin{bmatrix} F(x^*) & 0 \\ 0 & G(y^*) \end{bmatrix} \qquad (10)$$

must be negative semi-definite on the tangent plane determined by the binding (or active) constraints.

The second order sufficient conditions for a strong local maximum is that (3)–(9) must hold and that the Lagrangian in (10) must be negative definite on the tangent plane determined by the binding constraints.

To determine necessary and sufficient conditions it is therefore important to consider all possibilities in which the constraints may be binding. Exactly one of the following three cases may arise:

1. $x + y = B$ is the only binding constraint,
2. $x + y = B$ and $0 \leq x \leq u_x$ (*i.e.* either $x = 0$ or $x = u_x$) are the only binding constraints, or
3. $x + y = B$ and $0 \leq y \leq u_y$ (*i.e.* either $y = 0$ or $y = u_y$) are the only binding constraints.

In all three cases additional conditions are required for a feasible solution to be an optimal solution candidate. The following theorem summarises all the cases with their respective additionally required conditions. However, in the theorem we require the notion of curvature. If C is a smooth curve of the graph $y = f(x)$, then the *curvature* of C at the point (x, y) is given by [9]

$$K_f(x) = \frac{|f''(x)|}{[1 + (f'(x))^2]^{3/2}}.$$

Theorem 1
Let (x^*, y^*) be a feasible solution to (2) for which $x^* + y^* = B$ is a binding constraint.

A *If additionally $x^* = 0$ (or alternatively $y^* = u_y$) is binding, and if $f'(x^*) < g'(y^*)$, then the second order necessary conditions for the point (x^*, y^*) to be a relative maximum point hold.*

B If no other constraints are binding, then $f'(x^*) = g'(y^*)$. Furthermore, if $|K_f(x^*)| < |K_g(y^*)|$, then the second order sufficient conditions for the point (x^*, y^*) to be a strong relative maximum point hold.

C If additionally $x^* = u_x$ (or alternatively $y^* = 0$) is binding, and if $f'(x^*) > g'(y^*)$, then the second order necessary conditions for the point (x^*, y^*) to be a relative maximum point hold.

The proofs of Parts A and B are supplied. The proof of Part C is omitted, because it is similar to the proof of Part A. The cases of Theorem 1 are summarised schematically in Figure 1.

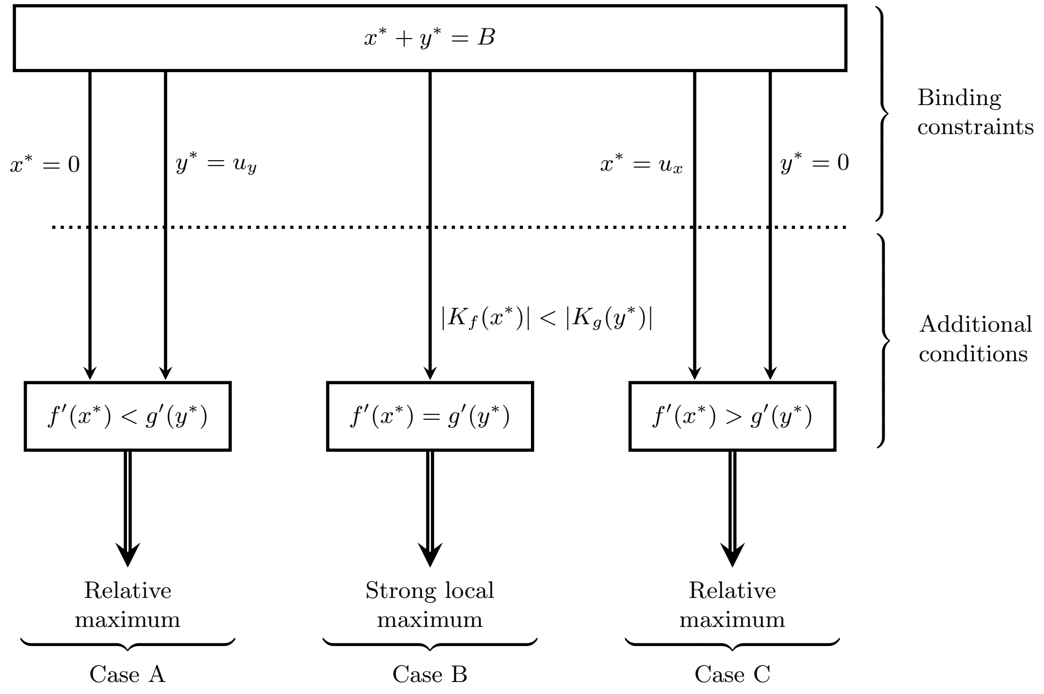

Figure 1: *Schematic representation of the cases in Theorem 1.*

Proof of A: From the conditions that $x + y = B$ and $x = 0$ are binding constraints in (2), as well as from (7)–(9), it follows that $\mu_2 = \mu_3 = \mu_4 = 0$. Because $x = 0$, μ_1 may assume any non-negative value. If this result is substituted back into (4)–(5), it follows that

$$f'(x^*) + \lambda + \mu_1 = 0 = g'(y^*) + \lambda,$$

or

$$f'(x^*) + \mu_1 = g'(y^*).$$

In the alternative situation, *i.e.* when $x + y = B$ and $y = u_y$ are binding, then by the same argument it follows that $\mu_4 \geq 0$ and

$$f'(x^*) = g'(y^*) - \mu_4.$$

The functions $f(x)$ and $g(y)$ are both increasing and therefore $f'(x) \geq 0$ and $g'(y) \geq 0$. The second order necessary conditions [3] for optimality of a point (x^*, y^*) are that the point should be a regular point and that the Lagrangian should be negative semi-definite on the tangent plane determined by the active constraints. For (x^*, y^*) to be a regular point, the vectors

$$\left[\frac{\partial(x+y-B)}{\partial x}, \frac{\partial(x+y-B)}{\partial y}\right] = [1,1] \quad \text{and} \quad \left[\frac{\partial(x-0)}{\partial x}, \frac{\partial(x-0)}{\partial y}\right] = [1,0]$$

should be linearly independent at the point (x^*, y^*), which they are.

At the point (x^*, y^*) on the tangent plane it holds that

$$\begin{aligned} \frac{\partial(x+y-B)}{\partial x} z_1 + \frac{\partial(x+y-B)}{\partial y} z_2 = 0 \\ \frac{\partial(x-0)}{\partial x} z_1 + \frac{\partial(x-0)}{\partial y} = 0, \end{aligned}$$

which means that $z_1 + z_2 = 0$ and $z_1 = 0$. For the second order necessary conditions it should hold that $\boldsymbol{z}^T L(x^*, y^*) \boldsymbol{z} \leq 0$, which is satisfied since $z_1 = z_2 = 0$. ■

In the situation described in Case A of Theorem 1, no sufficient conditions for optimality can be found form the general theory on nonlinear programming. We thus have to conclude that if the point (x^*, y^*) is a feasible solution for (2) such that $x^* + y^* = B$ and $x^* = 0$ (or $y^* = u_y$) are the only binding constraints, then general second order sufficient conditions for the feasible solution (x^*, y^*) to be a strong relative maximum point for (2) cannot be found.

Proof of B: If $x + y = B$ is the only binding constraint in (2), it follows from (6)–(9) that $\mu_1 = \mu_2 = \mu_3 = \mu_4 = 0$. These values, together with (4) and (5), gives the first result, namely that $f'(x^*) = g'(y^*)$, implying that the first order necessary conditions hold if $f'(x^*) = g'(y^*)$.

Sufficient conditions for local optimality at the point (x^*, y^*) require that, in addition to the requirement that the first order necessary conditions must hold, the Lagrangian $L(x^*, y^*)$ must be negative definite on the tangent plane M defined by

$$\begin{aligned} M &= \left\{ \boldsymbol{z} \,\middle|\, \frac{\partial(x+y-B)}{\partial x} z_1 + \frac{\partial(x+y-B)}{\partial y} z_2 = 0 \right\} \\ &= \{\boldsymbol{z} \mid z_1 + z_2 = 0\} \quad \text{for all } z_1 \text{ and } z_2, \end{aligned}$$

at the point (x^*, y^*). Thus, for any point $\boldsymbol{z} = (z_1, z_2)$ on the tangent plane, $z_1 = -z_2$ holds. For $L(x^*, y^*)$ to be negative definite on M it is required that

$$[z_1 \quad -z_1] \begin{bmatrix} F(x^*) & 0 \\ 0 & G(y^*) \end{bmatrix} \begin{bmatrix} z_1 \\ -z_1 \end{bmatrix} < 0 \quad \text{for all } z_1, z_2 \neq 0.$$

From the properties of $f(x)$ and $g(y)$ it follows that $F(x^*) \geq 0$ and $G(y^*) \leq 0$. Hence $z_1^2 \left[F(x^*) + G(y^*)\right] < 0$ and $F(x^*) + G(y^*) < 0$ if $F(x^*) < |G(y^*)|$. If the inequality $F(x^*) < |G(y^*)|$ holds, then $|K_g(y^*)| > |K_f(x^*)|$, which completes the proof. ■

The same conclusion, namely that in general no sufficient conditions for optimality can be found from the general theory, follows from the proof of Part C.

In order for the necessary conditions to hold, it follows from Theorem 1, that one of the following must hold for any feasible solution (x^*, y^*) to (2) so that the feasible solution may be considered a candidate for a globally optimal point:

1. $x^* = 0$, $0 < y^* < u_y$ and $f'(x^*) \leq g'(y^*)$
   In this situation no sufficient condition for global optimality can be determined from the general theory.
2. $x^* = u_x$, $0 < y^* < u_y$ and $f'(x^*) \geq g'(y^*)$
   In this situation no sufficient condition for global optimality can be determined from the general theory.
3. $y^* = 0$, $0 < x^* < u_x$ and $f'(x^*) \geq g'(y^*)$
   In this situation no sufficient condition for global optimality can be determined from the general theory.
4. $y^* = u_y$, $0 < x^* < u_x$ and $f'(x^*) \leq g'(y^*)$
   In this situation no sufficient condition for global optimality can be determined from the general theory.
5. $0 < x^* < u_x$, $0 < y^* < u_y$ and $f'(x^*) = g'(y^*)$
   In this situation the necessary conditions can be changed to sufficient conditions for a locally optimal point if $|K_g(y^*)| > |K_f(x^*)|$. Practically it means that the concave function should curve "more" (*i.e.* have a larger curvature) for the sufficient conditions to hold at the point (x^*, y^*).

If all the situations above are considered, it would seem that the following fairly simple algorithm may be used to solve (2):

1. find, if they exist, solutions for which $x + y = B$ and $0 \leq x \leq u_x$ are binding, *i.e.* $\boldsymbol{x}_1 = (0, B)$ and $\boldsymbol{x}_2 = (u_x, B - u_x)$;
2. find, if they exist, solutions for which $x + y = B$ and $0 \leq y \leq u_y$ are binding, *i.e.* $\boldsymbol{x}_3 = (B, 0)$ and $\boldsymbol{x}_4 = (B - u_y, u_y)$;
3. find, if it exists, a best solution for which only $x + y = B$ is binding, *i.e.* find $\boldsymbol{x}_5 = (x_5, y_5)$, by performing a search for point(s) satisfying $0 < x < u_x$, $0 < y < u_y$, $x + y = B$, $f'(x) = g'(y)$ and $f''(x) \leq g''(y)$; and, finally
4. select the best solution from the five obtained above.

Steps 1 and 2 above are computationally inexpensive. To select an optimal solution (in Step 4) does not bring about an additional computational burden either. Furthermore, it is computationally simple to find one point for which the conditions in Step 3 hold numerically. A problem would, however, arise if there are a number of these points, in which case each of the points would have to be enumerated. To determine the number of points for which the conditions in Step 3 hold is equivalent to answering the following question: how many times can the difference between two increasing, convex functions be zero? Nahay [6] showed, by means of an example, that it is possible that this difference may intersect any value (and hence zero) an arbitrarily large number of times on a finite interval! In general, there is thus no upper bound on the number of points that have to be enumerated in Step 3. An algorithm based on the four steps above would therefore not be suitable as an algorithm to find an optimal solution to (2). It could, however, work as an outline for a heuristic approach towards solving this problem.

We note that the problem with only two functions considered in this section may be transformed to a problem containing only one function in one variable. This renders an unconstrained optimisation problem in one variable. Although it is a simpler formulation, it does not shed any new light on solving for an optimal solution to (2).

3 Generalisation to more functions

In the generalisation to more functions the increase in the number of concave functions poses no real problem. If the concave part of the formulation is seen as a subproblem (when $m = 0$) it is a well researched problem for which efficient algorithms have been established [8]. The same does not apply for the convex functions. An increase in the number of convex functions increases the complexity of the problem considerably.

Consider problem (1). In general the second order necessary conditions state that for a feasible solution $\boldsymbol{x}^* = (x_1^*, \ldots, x_m^*, y_1^*, \ldots, y_n^*)$ to be optimal, there must exist a $\lambda \in \mathbb{R}$ and $\mu_1, \ldots, \mu_m$, $\nu_1, \ldots, \nu_m$, $\eta_1, \ldots, \eta_n$, $\omega_1, \ldots, \omega_n \geq 0$ for which

$$\begin{aligned}
\sum_{i=1}^{m} x_i^* + \sum_{j=1}^{n} y_j^* &= B && && (11)\\
f'(x_i^*) + \lambda + \mu_i - \nu_i &= 0, && i = 1, \ldots, m && (12)\\
g'(y_j^*) + \lambda + \eta_j - \omega_j &= 0, && j = 1, \ldots, n && (13)\\
\mu_i(x_i^*) &= 0, && i = 1, \ldots, m && (14)\\
\eta_j(y_j^*) &= 0, && j = 1, \ldots, n && (15)\\
\nu_i(u_i - x_i^*) &= 0, && i = 1, \ldots, m && (16)\\
\omega_j(u_j - y_j^*) &= 0, && j = 1, \ldots, n && (17)
\end{aligned}$$

and for which

$$L(\boldsymbol{x}^*) = \begin{bmatrix} F_1(x_1^*) & \ldots & 0 & 0 & \ldots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & F_m(x_m^*) & 0 & \ldots & 0 \\ 0 & \ldots & 0 & G_1(y_1^*) & \ldots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & 0 & \ldots & G_n(y_n^*) \end{bmatrix} \qquad (18)$$

is negative semi-definite on the tangent plane M determined by the active constraints, where

$$F_i(x_i) = \frac{\partial^2 f_i(x_i)}{\partial x_i^2} \quad \text{and} \quad G_j(y_j) = \frac{\partial^2 g_j(y_j)}{\partial y_j^2}.$$

This implies that for $z_1, \dots, z_m, w_1, \dots, w_n \in M$,

$$[z_1, \dots, z_m, w_1, \dots, w_n] \begin{bmatrix} F_1(x_1^*) & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & F_m(x_m^*) & 0 & \dots & 0 \\ 0 & \dots & 0 & G_1(y_1^*) & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & \dots & G_n(y_n^*) \end{bmatrix} \begin{bmatrix} z_1 \\ \vdots \\ z_m \\ w_1 \\ \vdots \\ w_n \end{bmatrix} \leq 0. \quad (19)$$

From (19) it follows that

$$z_1^2 F_1(x_1^*) + \dots + z_m^2 F_m(x_m^*) + w_1^2 G_1(y_1^*) + \dots + w_n^2 G_n(y_n^*) \leq 0.$$

Let M_1 be the tangent plane on the equality constraint, M_2 the tangent plane on the active lower bound constraints and M_3 be the tangent plane on the active upper bound constraints. Then $M = M_1 \cap M_2 \cap M_3$, and thus for $z_i, w_j \in M$ it follows that

$$z_1 + \dots + z_m + w_1 + \dots + w_n = 0 \qquad M_1 \qquad (20)$$

$$\left.\begin{aligned} z_i &= 0 \quad \text{if } x_i = 0 \\ w_j &= 0 \quad \text{if } y_j = 0 \end{aligned}\right\} \qquad M_2 \qquad (21)$$

$$\left.\begin{aligned} z_i &= 0 \quad \text{if } x_i = u_i \\ w_j &= 0 \quad \text{if } y_j = u_j \end{aligned}\right\} \qquad M_3. \qquad (22)$$

Theorem 1 may be generalised to more functions using this result. All the variables of a feasible solution $(x_1^*, \dots, x_m^*, y_1^*, \dots, y_n^*)$ may be partitioned into three sets, namely

$$\begin{aligned} \mathcal{L} &= \{i \mid x_i^* = 0\} \cup \{j \mid y_j^* = 0\}, \\ \mathcal{U} &= \{i \mid x_i^* = u_i^{(x)}\} \cup \{j \mid y_j^* = u_j^{(y)}\} \text{ and} \\ \mathcal{K} &= \{i \mid x_i^* \in (0, u_i^{(x)})\} \cup \{j \mid y_j^* \in (0, u_j^{(y)})\}. \end{aligned}$$

Each entry of a feasible solution will be in exactly one of these three sets, because the set $\mathcal{L}$ contains all the variables that are at their lower bounds, the set $\mathcal{U}$ contains all the variables that are at their upper bounds and the set $\mathcal{K}$ contains all the variables neither at their upper nor at their lower bounds. If a feasible solution is represented in this fashion it follows from (11)–(17) that

$$\left.\begin{array}{rcl} f_i'(x_i^*) & \leq & -\lambda \\ g_j'(y_j^*) & \leq & -\lambda \end{array}\right\} \text{for all } \ x_i^*, y_j^* \in \mathcal{L}$$

$$\left.\begin{array}{rcl} f_i'(x_i^*) & \geq & -\lambda \\ g_j'(y_j^*) & \geq & -\lambda \end{array}\right\} \text{for all } \ x_i^*, y_j^* \in \mathcal{U}$$

$$\left.\begin{array}{rcl} f_i'(x_i^*) & = & -\lambda \\ g_j'(y_j^*) & = & -\lambda \end{array}\right\} \text{for all } \ x_i^*, y_j^* \in \mathcal{K}.$$

All the functions with their variables in the set $\mathcal{K}$ have exactly the same slope at a feasible solution for which the first order necessary conditions hold. We call this slope the *consensus slope*, S_c. The result above may be summarised as

$$\begin{aligned} f_i'(x_i^*), g_j'(y_j^*) &\le S_c &\quad& \text{for all } x_i^*, y_j^* \in \mathcal{L} \\ f_i'(x_i^*) = g_j'(y_j^*) &= S_c && \text{for all } x_i^*, y_j^* \in \mathcal{K} \\ f_i'(x_i^*), g_j'(y_j^*) &\ge S_c && \text{for all } x_i^*, y_j^* \in \mathcal{U} \end{aligned}$$

in terms of the consensus slope. There is another result that may be used to limit the search space in order to facilitate development of a possible algorithm to solve the problem formulation in (1). Two different proofs of the following result were presented by Venter and Wolvaardt [11], and by Visagie [12].

Theorem 2
If a feasible solution $\boldsymbol{x}^* = (x_1, \ldots, x_i, \ldots, x_m)$ is an optimal solution to (1) with $n = 0$, then at most one element of $\boldsymbol{x}^*$ is in the set $\mathcal{K}$.

The following generalisation of the above result holds for the problem formulation in (1).

Theorem 3
An optimal solution $\boldsymbol{x}^* = (x_1^*, \ldots, x_m^*, y_1^*, \ldots, y_n^*)$ to (1) with $n > 0$ has at most one $x_i^* \in \mathcal{K}$.

Proof: The formulation in (1) may be divided into a master problem and two subproblems. The first subproblem (P_1) contains all the convex functions and the second subproblem (P_2) contains all the concave functions. The subproblem P_1 is given by

$$\left.\begin{aligned} &\text{maximise } Z_1 = \sum_{i=1}^{m} f_i(x_i) \\ &\text{subject to} \qquad \sum_{i=1}^{m} x_i = B_1 \\ &\qquad\qquad\qquad 0 \le x_i \le u_i \qquad i = 1, 2, \ldots, m \end{aligned}\right\} \tag{23}$$

and subproblem P_2 by

$$\left.\begin{aligned} &\text{maximise } Z_2 = \sum_{j=1}^{n} g_j(y_j) \\ &\text{subject to} \qquad \sum_{j=1}^{n} y_j = B_2 \\ &\qquad\qquad\qquad 0 \le y_j \le u_j \qquad j = 1, 2, \ldots, n. \end{aligned}\right\} \tag{24}$$

The master problem is then given by

$$\left.\begin{aligned} &\text{maximise } z = Z_1 + Z_2 \\ &\text{subject to} \qquad B_1 + B_2 = B \\ &\qquad\qquad\qquad B_1, B_2 \ge 0. \end{aligned}\right\} \tag{25}$$

If an optimal solution $(x_1^*, \ldots, x_m^*, y_1^*, \ldots, y_n^*)$ to (25) exists, then $(x_1^*, \ldots, x_m^*)$ must be an optimal solution to (23) and $(y_1^*, \ldots, y_n^*)$ must be an optimal solution to (24). If this were not the case for one of the subproblems, then it would be possible to obtain a better objective function value for that subproblem, with the same amount of resources. However, then this improved solution may have been used as is in the master problem, without any effect on the other subproblem. This would then lead to a better solution for the master problem which is not possible, because the original solution was already optimal. Hence the solutions $(x_1^*, \ldots, x_m^*)$ and $(y_1^*, \ldots, y_n^*)$ to both subproblems must be optimal. If the solution $(x_1^*, \ldots, x_m^*)$ is an optimal solution for (23), then Theorem 2 must hold for (23) and thus at most one of the convex function variables may be in $\mathcal{K}$. ■

Theorem 1 may now also be generalised.

Theorem 4
Let $(x_1^*, \ldots, x_m^*, y_1^*, \ldots, y_n^*)$ be a feasible solution to (1), for which

A $f_i'(x_i^*) \leq S_c$ *and* $g_j'(y_j^*) \leq S_c$ *for all* $x_i^*, y_j^* \in \mathcal{L}$,

B $f_i'(x_i^*) = g_j'(y_j^*) = S_c$ *for at most one* $x_i^* \in \mathcal{K}$, *and for all* $y_j^* \in \mathcal{K}$ *and in addition, if there is one* $x_i \in \mathcal{K}$, *then* $|K_{f_i}(x_i)| < |\sum_j K_{g_j}(y_j)|$ *for all* $y_j^* \in \mathcal{K}$, *and*

C $f_i'(x_i^*), g_j'(y_j^*) \geq S_c$ *for all* $x_i^*, y_j^* \in \mathcal{U}$

hold. Then the first and second order necessary conditions for $(x_1^*, \ldots, x_m^*, y_1^*, \ldots, y_n^*)$ to be an optimal solution hold.

Thus all solutions $(x_1^*, \ldots, x_m^*, y_1^*, \ldots, y_n^*)$ having either one of the two properties described below, must be determined to find a globally optimal solution.

1. If there is exactly one $x_i^* \in \mathcal{K}$, say x_k^* (and thus $f_k'(x_k^*) = S_c$), then
   $S_c = g_j'(y_j^*)$ and $f_k''(x_k^*) < \sum_j g_j''(y_j^*)$ for all $y_j^* \in \mathcal{K}$,
   $f_i'(x_i^*) \leq S_c$ and $g_j'(y_j^*) \leq S_c$ for all $x_i^*, y_j^* \in \mathcal{L}$, and
   $f_i'(x_i^*) \geq S_c$ and $g_j'(y_j^*) \geq S_c$ for all $x_i^*, y_j^* \in \mathcal{U}$, or else
2. if there is no $x_i^* \in \mathcal{K}$, then
   $S_c = g_j'(y_j^*)$ for all $y_j^* \in \mathcal{K}$,
   $f_i'(x_i^*) \leq S_c$ and $g_j'(y_j^*) \leq S_c$ for all $x_i^*, y_j^* \in \mathcal{L}$, and
   $f_i'(x_i^*) \geq S_c$ and $g_j'(y_j^*) \geq S_c$ for all $x_i^*, y_j^* \in \mathcal{U}$.

With an increase in the number of convex functions the situation becomes worse than the situation with only one convex function, even though there may be at most one $x_i \in \mathcal{K}$, because the chances increase that there is more than one point for which $f_i'(x_i) = g_j'(y_j) = S_c$ holds for all $x_i, y_j \in \mathcal{K}$. For more than one such point to exist there must be at least two feasible solutions for which

$$\sum_{i=1}^{m} x_i + \sum_{j=1}^{n} y_j = B,$$

$$f_i'(x_i) = g_j'(y_j) \qquad \text{for all } x_i, y_j \in \mathcal{K}$$

and

$$\sum_{i\in\mathcal{K}} z_i^2 f_i''(x_i) \leq \sum_{j\in\mathcal{K}} w_j^2 g_j''(y_j) \qquad \text{for all } z_i, w_j \in M$$

holds. This implies that there should be at least one feasible solution for which

$$z_i^2 f_i''(x_i) = \sum_{j\in\mathcal{K}} w_j^2 g_j''(y_j), \tag{26}$$

with at most one $z_i \neq 0$. An increase in the number of convex functions translates to an increase in the number of choices on the left hand side of (26) to ensure that the equality holds.

4 Numerical results

For some types of problems (such as certain bin packing problems) standard sets of problem instances exist which may be used by researchers to measure the performance of new algorithms. Unfortunately, no such benchmark sets exist for the problem considered in this paper and therefore instances needed to be generated for the purpose of testing the algorithm described below. Eight families of increasing convex functions and increasing concave functions were used to create these instances as described in the following sections.

4.1 A possible solution approach

It would seem a logical approach to design an algorithm for solving (1) repeating itself n times. During each of these n repetitions another convex function is considered as the convex function of the decision variable in $\mathcal{K}$. As a first attempt one may approximate the remaining $n-1$ convex functions by enveloping them by means of straight lines. This ensures that these envelopes are indeed concave as well. Such an approach therefore has the advantage of considering all but one function as concave functions — and then using the available, efficient algorithms to solve the problem in the case where there are only concave functions. The problem with this approach is that the solution will almost always include an enveloped convex function (*i.e.* a straight line) that assumes a value between its bounds at an optimal solution candidate, which is, of course, disallowed. To avoid this, the algorithm may be altered to incorporate a binary branch and bound algorithm. In other words, as soon as the algorithm solves the problem and an enveloped function assumes a value between its bounds, the problem is branched into two subproblems. In the first subproblem this variable is set at its lower bound and in the second subproblem the variable is set at its upper bound. This implies solving a mixed 0/1 knapsack problem by a branch and bound technique. An algorithm based the logic of the Horowitz-Shani algorithm, see for example [5], may be used to solve this mixed 0/1 knapsack problem.

4.2 The functions utilised

The four families of convex functions

$$\begin{aligned}
f_i(x_i) &= -s_i^{(1)}\left(1-e^{-m_i^{(1)}x_i}\right),\\
f_i(x_i) &= -s_i^{(2)}x_i + m_i^{(2)}x_i^2,\\
f_i(x_i) &= -\frac{s_i^{(3)}(x_i+c_i^{(3)})}{(x_i+m_i^{(3)})}, \quad m_i^{(3)} > c_i^{(3)} \quad \text{and}\\
f_i(x_i) &= -s_i^{(4)}\ln(1+m_i^{(4)}x_i)
\end{aligned}$$

have been used in the literature [2, 10] to generate nonlinear knapsack problems. The first authors utilising these functions were Luss and Gupta [4]. Here the constants $s_i^{(k)}$, $m_i^{(k)}$ and $c_i^{(k)}$, with $k = 1, 2, 3, 4$ are positive parameters which may be used to alter the shape of the functions.

Unfortunately these families of functions cannot be used in this paper because they are not all increasing functions. Small adaptations were applied to these functions to ensure that they are nonnegative and increasing over the interval $[0, u_i^{(k)}]$. The adopted families of functions are

$$\begin{aligned}
f_i(x_i) &= -s_i^{(1)} + s_i^{(1)}e^{m_i^{(1)}x_i},\\
f_i(x_i) &= m_i^{(2)}x_i^2 + s_i^{(2)}x_i,\\
f_i(x_i) &= \frac{s_i^{(3)}(u_i^{(3)}+c_i^{(3)})}{u_i^{(3)}+m_i^{(3)}} - \frac{s_i^{(3)}[(u_i^{(3)}-x_i)+c_i^{(3)}]}{[(u_i^{(3)}-x_i)+m_i^{(3)}]}, \quad m_i^{(3)} > c_i^{(3)} \quad \text{and}\\
f_i(x_i) &= s_i^{(4)}\ln(1+m_i^{(4)}u_i^{(4)}) - s_i^{(4)}\ln(1+m_i^{(4)}(u_i^{(4)}-x_i)).
\end{aligned}$$

Note that all the above functions have already been transformed so that their lower bounds are zero and so that $f_i(0) = 0$ for all $i = 1, \ldots, m$.

Similarly, the families of concave functions

$$\begin{aligned}
g_j(y_j) &= s_j^{(5)}\left(1-e^{-m_j^{(5)}y_j}\right),\\
g_j(y_j) &= s_j^{(6)}y_j - m_j^{(6)}y_j^2,\\
g_j(y_j) &= \frac{s_j^{(7)}(y_j+c_j^{(7)})}{(y_j+m_j^{(7)})}, \quad m_j^{(7)} > c_j^{(7)} \quad \text{and}\\
g_j(y_j) &= s_j^{(8)}\ln(1+m_j^{(8)}y_j)
\end{aligned}$$

were used for benchmark generation purposes. These functions were adapted to

$$\begin{aligned}
g_j(y_j) &= s_j^{(5)}\left(1 - e^{-m_j^{(5)}y_j}\right),\\
g_j(y_j) &= s_j^{(6)}u_j^{(6)} + m_j^{(6)}(u_j^{(6)})^2 - s_j^{(6)}(u_j^{(6)} - y_j) - m_j^{(6)}(u_j^{(6)} - y_j)^2,\\
g_j(y_j) &= -\frac{s_j^{(7)}c_j^{(7)}}{m_j^{(7)}} + \frac{s_j^{(7)}(y_j + c_j^{(7)})}{(y_j + m_j^{(7)})}, \qquad m_j^{(7)} > c_j^{(7)} \text{ and}\\
g_j(y_j) &= s_j^{(8)}\ln(1 + m_j^{(8)}y_j),
\end{aligned}$$

in order to ensure that all of these functions are nonnegative, increasing concave functions on the interval $[0, u_j^{(k)}]$ for $j = 1, \ldots, m$ and $k = 5, 6, 7, 8$.

The constants $s_j^{(k)}$, $m_j^{(k)}$ and $c_j^{(k)}$, with $k = 5, 6, 7, 8$ are again positive parameters that may be used to alter the function shapes. Note that all the above functions have already been transformed so that their lower bounds are zero and so that $g_j(0) = 0$ for all $j = 1, \ldots, n$.

4.3 Methodology

All the algorithms tested in this paper were programmed in *Visual Basic for Applications* (VBA) in Microsoft Excel. Excel's ***rand*** and ***randbetween*** functions were utilised for random numbers generation purposes. Benchmark instances were generated as follows. All upper bounds on the decision variables were randomly generated (for each generated instance the minimum and maximum length of an interval were specified as 100 and 200, respectively). A budget amount B was then calculated as

$$\tfrac{1}{2}\left(\sum_{i=1}^{m} u_i^{(x)} + \sum_{j=1}^{n} u_j^{(y)}\right).$$

Thereafter an average slope, $f_i(u_i^{(x)})/u_i^{(x)}$ or $g_i(u_i^{(y)})/u_i^{(y)}$, was randomly generated within specified bounds (for this paper it was restricted to between 5 and 8). A random integer number between 1 and 4 was generated for both the convex and concave functions to determine which family of functions should be used in each case. Random values for $m_i^{(k)}$ or $m_j^{(k)}$, $c_i^{(k)}$ or $c_j^{(k)}$ were then generated. Finally, the average slope together with $m_i^{(k)}$, $m_j^{(k)}$, $c_i^{(k)}$ and $c_j^{(k)}$ were then used to calculate $s_i^{(k)}$ or $s_j^{(k)}$, thus determining the shape of the randomly generated function.

4.4 Results

The approach described in §4.1 yields optimal solutions, but the solution times are unacceptably long. The main reason for these long solution times is that considerably more branches are possible than in a pure 0/1 knapsack problem, because the concave part of the problem may "finance" additional branches by shifting resources between the binary and continuous parts of the problem.

The solution times are listed in Table 1. The table compares the solution times of piecewise linear enveloping together with the branch and bound technique, which also yields the optimal solutions, see for example [1]. These solution times were recorded on a Pentium(R) 4 computer with a 3.2 GHz central processing unit and 1 GB random access memory. A hundred random instances of the problem were generated and solved for each reported value of n.

n	NBB	MBB
5	0.0625	1.0155
10	0.0781	1.8750
15	0.1250	4.2195
20	0.1563	5.9380
25	0.3438	10.1575
30	0.3281	15.3060
35	0.5313	19.1415
40	0.7656	35.6240
45	0.9219	40.0770
50	1.3906	71.8750
55	2.2813	121.6875

Table 1: *Total solution times in seconds for 100 randomly generated instances of the problem for each reported value of n. The following abbreviations are used: NBB = normal branch and bound technique with piecewise linear enveloping and MBB = mixed 0/1 branch and bound algorithm.*

It is clear from the data in Table 1 that piecewise linear enveloping, with the branch and bound technique, outperforms the algorithm described above, when both are used to solve the same problem sets. Because of the poor solution times achieved by the proposed algorithm it is not presented in detail here.

Dynamic programming was used to validate the algorithmic results presented here.

5 Conclusions and ideas for further research

The main reason why the analytic, continuous approach to the problem formulation in (1) is not successful is that there are potentially many feasible solutions for which the necessary conditions hold. In fact, it may be shown that there may be arbitrarily many feasible solutions to (1) for which the conditions in Theorem 1 hold. This renders a brute force search for all these feasible solutions impractical. This conclusion remains the same if the results of Theorem 3 are used to supply additional necessary conditions in a bid to limit the search space for all feasible solutions for which the necessary conditions hold.

The possibility of constructing a heuristic that searches for only the first N points for which the necessary condition holds may be investigated further, for some specified value of N.

An analytical, continuous approach does not compete favourably with linear enveloping in conjunction with branch and bound algorithms. Research into improving the manner of performing piecewise linear enveloping, as well as on how the branching should be achieved, may lead to algorithms with shorter solution times than the normal branch and bound approach used in this paper.

Finally, there is a need to generate and document standard sets of benchmark instances for this problem formulation that have different properties regarding, for example, curvature of the functions, average slopes of the functions and lengths of the intervals $[0, u_i]$ of the decision variables. These standard sets may then be used to test algorithms against each other in future.

References

[1] Bretthauer KM & Shetty B, 1995, *The nonlinear resource allocation problem*, Operations Research, **43(4)**, pp. 670–683.

[2] Ibaraki T & Katoh N, 1988, *Resource allocation problems*, MIT Press, Cambridge (MA).

[3] Luenberger DG, 1973, *Introduction to linear and nonlinear programming*, Addison-Wesley Publishing Company Inc., Reading (MA).

[4] Luss H & Gupta TCA, 1975, *Allocation of effort resources among competing activities*, Operations Research, **23(2)**, pp. 360–366.

[5] Martello S & Toth P, 1990, *Knapsack problems: Algorithms and computer implementations*, John Wiley & Sons, Chichester.

[6] Nahay JM, 2004, *Differential resolvents are complete and useful*, [Online], [Cited May 2006], Available from `http://math.hunter.cuny.edu/ksda/papers/nahay.pdf`

[7] Patriksson M, 2005, *A survey on a classic core problem in operations research*, Report, Department of Mathematics, Chalmers University of Technology, Göteborg (Gothenburg).

[8] Patriksson M, 2008, *A survey on the continuous nonlinear resource allocation problem*, European Journal of Operational Research, **185(1)**, pp. 1–46.

[9] Swokovski EW, 1990, *Calculus*, 5th edition, PWS-Kent Publishing Company, Boston (MA).

[10] Venter G, 1994, *'n Doeltreffende algoritme vir die algehele minimum van die konkawe knapsakprobleem*, MSc Thesis, University of South Africa, Pretoria.

[11] Venter G & Wolfvaardt JS, 2005, *'n Algoritme vir die minimum van die konkawe knapsakprobleem*, ORiON, **21(1)**, pp. 13–31.

[12] Visagie SE, 2007, *Algoritmes vir die maksimering van konvekse en verwante knapsakprobleme*, PhD Dissertation, University of Stellenbosch, Stellenbosch.