INDIAN INSTITUTE OF MANAGEMENT
AHMEDABAD • INDIA

Nonlinear 0-1 knapsack problem with capacity selection

Sachin Jayaswal

W.P. No. 2016-03-10

March 2016

INDIAN INSTITUTE OF MANAGEMENT
AHMEDABAD-380 015
INDIA

Nonlinear 0-1 knapsack problem with capacity selection

Sachin Jayaswal

Production & Quantitative Methods, Indian Institute of Management Ahmedabad
sachin@iima.ac.in

Abstract

We study a nonlinear 0-1 knapsack problem with capacity selection decision, as it arises as a part of facility location/service system design problems with congestion. The capacity selection decision gives rise to a non-convex objective function. We present two cutting plane based solution approaches: one based on Generalized Benders decomposition based, and the other based on a reformulation of the problem using additional auxiliary variables, followed by outer linearization of a resulting simple concave function in the constraint.

Keywords: knapsack, integer, non-convex, generalized Benders, cutting plane

Nonlinear 0-1 knapsack problem with capacity selection

1 Introduction

Consider a set of customer regions (nodes), indexed by $i \in I$, each generating demand arrivals with a Poisson distribution with rate λ_i to a service facility that takes a random amount of time with a mean $1/\mu$ and standard deviation σ (hence, coefficient of variation $cv = \sigma\mu$) to serve them. The service facility thus behaves like an $M/G/1$ queue with arrival rate $\Lambda = \sum_{i \in I} \lambda_i x_i$, where $x_i = 1$ if customer region i is served by the service facility, 0 otherwise. Revenue generated from customer region i, if served by the facility, is π_i. However, the variability in the arrival and the service processes results in congestion at the service facility, resulting in waiting for the customers. Customers are paid, on average, a penalty c per unit time for the time they spend in the system. Although the service facility may want to serve as many customer regions with its service capacity, doing so may lead to excessive penalty for waiting customers. So, it may not be optimal for the server to serve all the customer regions even if its capacity allows it to do so. Hence, the server faces the decision as to which customer regions to serve, i.e., $\{i \in I : x_i = 1\}$.

The service facility can appropriately select its capacity (i.e., mean service rate) μ as at most one from the set $\{\mu_l : l \in L\}$ of available capacity levels (with the corresponding standard deviation σ_l) such that $\mu = \sum_{l \in L} \mu_l y_l$, where $y_l = 1$ if capacity level l is selected, 0 otherwise. Setting up a capacity μ_l calls for an investment (amortized over a period) of $\$k_l \ \forall l \in L$. The server, therefore, also needs to decide its optimal service capacity μ, in addition to which customer regions to serve, such that its total profit (revenue minus penalty minus capacity cost) rate is maximized. We refer to this problem as Nonlinear 0-1 Knapsack Problem with Capacity Selection (NKPSC) for reasons described later in this section. Such a problem appears naturally as a part of facility location/service system design problems with congestion (Berman and Krass, 2002; Amiri, 1997, 1998). To state the problem mathematically, we first summarize below the notation used.

Indices:

i : node; $i \in I$

l : capacity level; $l \in L$

Parameters:

λ_i : Rate of Poisson demand arrival from node i

μ_l : Service rate of the facility at capacity level l

σ_l : Standard deviation of service times at capacity level l

cv_l : Coefficient of variation of service times at capacity level l; $cv_l = \mu_l \sigma_l$

k_l : Fixed (amortized over the time period) cost for capacity level l

π_i : Potential profit from node i

c : Average penalty per unit time per customer in the system

Variables:

x_i : 1, if the server serves customer region i, 0 otherwise

y_l : 1, if capacity level l for the server is selected, 0 otherwise

Using the above notation, the optimal decision problem of the server can be mathematically stated as:

$[NKPCS]$:

$$\max_{\mathbf{x},\mathbf{y}} \sum_{i \in I} \pi_i x_i - \sum_{l \in L} k_l y_l - cE[N] \tag{1}$$

$$\text{s.t.} \sum_{i \in I} \lambda_i x_i \leq \sum_{l \in L} \mu_l y_l \tag{2}$$

$$\sum_{l \in L} y_l \leq 1 \tag{3}$$

$$x_i \in \{0,1\} \qquad \forall i \in I \tag{4}$$

$$y_l \in \{0,1\} \qquad \forall l \in L \tag{5}$$

where $E[N]$ in (1) is the expected number of customers in the system in steady state in the

M/G/1 queuing system at the service facility, the expression for which is given as:

$$E[N] = \left(\frac{1+cv^2}{2}\right)\frac{\Lambda^2}{\mu(\mu-\Lambda)} + \frac{\Lambda}{\mu}$$
$$= \left(\frac{1+\sum_{l\in} cv_l^2 y_l}{2}\right)\frac{\left(\sum_{i\in I}\lambda_i x_i\right)^2}{\sum_{l\in L}\mu_l y_l(\sum_{l\in L}\mu_l y_l - \sum_{i\in I}\lambda_i x_i)} + \frac{\sum_{i\in I}\lambda_i x_i}{\sum_{l\in L}\mu_l y_l} \quad (6)$$

(2) is the steady state condition for the service facility. (3) enforces the selection of at most one capacity level. (4) and (5) are the binary constraints on the decision variables.

For a given capacity decision (i.e., fixed $y_l\ \forall l \in L$) and no waiting time penalty (i.e., $c = 0$), (1) - (5) reduces to the classical 0-1 knapsack problem (KP), which is known to be NP-hard (Martello et al., 2000). For a given capacity decision and a non-zero waiting time penalty (i.e., $c > 0$), the problem becomes a nonlinear (convex, non-separable) 0-1 knapsack problem (NKP), as studied by Elhedhli (2005). Within the 0-1 (or general integer) NKP class, quadratic, non-separable KPs have been studied by Gallo et al. (1980), Dussault et al. (1986), Klastorin (1990), Caprara et al. (1999). Readers are referred to an in-depth review of the various classes of NKPs by Bretthauer and Shetty (2002).

With capacity as a decision variable and a non-zero waiting time penalty (i.e., $c > 0$), the last term in the objective function of (1) - (5), as given by (6), becomes non-convex, non-separable. Hence, (1) - (5) falls in the class of non-convex, non-separable 0-1 NKPs, which is much more difficult to solve since locally optimal solutions may fail to be globally optimal. There has been little work on non-convex 0-1 NKPs. Moré and Vavasis (1990) study concave knapsack problems (with minimization as the objective), but deal with a continuous version. Moreover, they limit their focus primarily on finding local optimal, as opposed to global optimal, solutions.

The objective of this study is to develop efficient solution approaches for the 0-1 non-convex, non-separable NKP as described by (1) - (6). For this, we present two exact solution approaches. The first is based on Generalized Benders decomposition, while the second is based on a reformulation of the problem using additional auxiliary variables, followed by outer approximation of a resulting simple concave function in the constraint. We refer to the second method as Alternate Outer Approximation (AOA) since our GBD can also be viewed as one form of outer approximation. The paper thus contributes to the very limited literature on non-convex, non-separable NKP by presenting its excat solution approaches. The rest of the paper is organized as follows. The two solution approaches, namely GBD and AOA, are presented in Sections 2 & 3, respectively. The paper concludes with a comparison of the two solution methods based on computational results reported in Section 4.

2 Generalized Benders decomposition

The last term in (1), as given by (6), is not jointly convex in (Λ, μ). However, for a fixed value of μ, it becomes strictly convex, which we exploit to solve the problem using Generalized Benders Decomposition. However, the term lacks separability in Λ and μ, something desirable for an implementable algorithm for GBD. With this in mind, we now present NKPCS a little differently. For this, we define $x_{il} = 1$ if region i is served by the server with capacity level $l \in L$. NKPCS can then be rewritten as:

$$\max_{\mathbf{x},\mathbf{y}} \sum_{i\in I}\sum_{l\in L} \pi_i x_{il} - \sum_{l\in L} k_l y_l - c\sum_{l\in L}\left\{\left(\frac{1+cv_l^2}{2}\right)\frac{\Lambda_l^2}{\mu_l(\mu_l-\Lambda_l)} + \frac{\Lambda_l}{\mu_l}\right\} \tag{7}$$

$$\text{s.t. } \Lambda_l = \sum_{i\in I} \lambda_i x_{il} \qquad \forall l \in L \tag{8}$$

$$\sum_{i\in I} \lambda_i x_{il} \le \mu_l y_l \qquad \forall l \in L \tag{9}$$

$$\sum_{l\in L} y_l \le 1 \tag{10}$$

$$x_{il} \in \{0,1\} \qquad \forall i \in I; l \in L \tag{11}$$

$$y_l \in \{0,1\} \qquad \forall l \in L \tag{12}$$

Projecting the problem onto the space of the $\mathbf{x}$ and $\mathbf{y}$ variables results in the following equivalent problem:

$$\max_{\mathbf{x},\mathbf{y}} \sum_{i\in I}\sum_{l\in L}\pi_i x_{il} - \sum_{l\in L} k_l y_l - c\sum_{l\in L}\phi_l(\mathbf{x}_l, y_l) \tag{13}$$
$$\text{s.t. } (9) - (12)$$

where, $\phi_l(\mathbf{x}_l, y_l)$ solves for the following Sub Problem:

$[SP_l]$:

$$\phi_l(\mathbf{x}_l, y_l) = \min_{\Lambda_l} \left\{ \left(\frac{1+cv_l^2}{2}\right)\frac{\Lambda_l^2}{\mu_l(\mu_l-\Lambda_l)} + \frac{\Lambda_l}{\mu_l} \right\} \tag{14}$$
$$\text{s.t. } (8)$$

Clearly, $[SP_l]$ is always feasible for any given choice of $\mathbf{x}_l, y_l$ variables. Further, its objective function is convex and differentiable, and its constraint is linear (in the variable Λ_l). Hence, its KKT conditions are necessary and sufficient for optimality. So, associating a dual variable $\alpha_l\ \forall l \in L$ to constraint (8), and because there is no duality gap, $\phi_l(\mathbf{x}_l, y_l)$ can be rewritten as:

$$\phi_l(\mathbf{x}_l, y_l) = \max_{\alpha\geq 0}\left\{ \min_{\Lambda_l\geq 0}\left\{ \left(\frac{1+cv_l^2}{2}\right)\frac{\Lambda_l^2}{\mu_l(\mu_l-\Lambda_l)} + \frac{\Lambda_l}{\mu_l} - \alpha_l\Lambda_l \right\} + \alpha_l\sum_{i\in I}\lambda_i x_{il} \right\}$$

Introducing a new variable $\theta_l \geq 0$, (7) - (12) can be rewritten as the following Master Problem:

$[MP]$:

$$Z = \max_{\mathbf{x},\mathbf{y}} \sum_{i\in I}\sum_{l\in L}\pi_i x_{il} - \sum_{l\in L} k_l y_l - c\sum_{l\in L}\theta_l \tag{15}$$
$$\text{s.t. } (9) - (12)$$
$$\theta_l \geq \min_{\Lambda_l\geq 0}\left\{ \left(\frac{1+cv_l^2}{2}\right)\frac{\Lambda_l^2}{\mu_l(\mu_l-\Lambda_l)} + \frac{\Lambda_l}{\mu_l} - \alpha_l\Lambda_l \right\} + \alpha_l\sum_{i\in I}\lambda_i x_{il} \quad \forall l \in L, \forall\alpha \tag{16}$$

$$\theta_l \geq 0 \qquad \forall l \in L \tag{17}$$

For a given iteration t, where $\mathbf{x}_l = \mathbf{x}_l^t$ and $y_l = y_l^t$ $\forall l \in L$, and after the solution of the associated $[SP_l]$ and the recovery of the optimal value of α_l (call it α_l^t) $\forall l \in L$, the optimal value of $\phi_l(\mathbf{x}_l^t, y_l^t)$ is given by:

$$\phi_l(\mathbf{x}_l^t, y_l^t) = \min_{\Lambda_l \geq 0} \left\{ \left(\frac{1+cv_l^2}{2}\right) \frac{\Lambda_l^2}{\mu_l(\mu_l - \Lambda_l)} + \frac{\Lambda_l}{\mu_l} - \alpha_l^t \Lambda_l \right\} + \alpha_l^t \sum_{i \in I} \lambda_i x_{il}^t \tag{18}$$

Further, constraint (16), for iteration t, can be rewritten as:

$$\theta_l \geq \min_{\Lambda_l \geq 0} \left\{ \left(\frac{1+cv_l^2}{2}\right) \frac{\Lambda_l^2}{\mu_l(\mu_l - \Lambda_l)} + \frac{\Lambda_l}{\mu_l} - \alpha_l^t \Lambda_l \right\} + \alpha_l^t \sum_{i \in I} \lambda_i x_{il} \quad \forall l \in L \tag{19}$$

Therefore, by eliminating the minimum in (19) using (17), [MP] can be restated as:

$$(15)$$

$$\text{s.t.} (9) - (12), (17)$$

$$\theta_l \geq \phi_l(\mathbf{x}_l^t, y_l^t) + \alpha_l^t \sum_{i \in I} (\lambda_i x_{il} - \lambda_i x_{il}^t) \quad \forall l \in L; \forall t = 1, ..., T \tag{20}$$

T in (20) is the total number of possible values of $(\mathbf{x}_l, y_l)$. Including constraints corresponding to all possible values of t in (20) will result in a very large [MP]. Hence, constraints of (20) are only added one at a time at each iteration, and not all of them are needed to solve the problem to optimality. So, at any iteration, a relaxed Master Problem [RMP] with only a subset of constraints corresponding to $t = 1, ...T'$, where $T' < T$, in (20) is solved. For ($\mathbf{x}_l^t$, y_l^t) fixed by [RMP] at iteration t, the Sub Problem $\phi_l(\mathbf{x}_l^t, y_l^t)$ is given by:

$$(14)$$

$$\text{s.t. } \Lambda_l = \sum_{i \in I} \lambda_i x_{il}^t \tag{21}$$

With the dual variable α associated with constraint (21), [SP] at iteration t can be solved using KKT conditions, as given below:

$$\frac{\partial}{\partial \alpha_l}\left\{\left(\frac{1+cv_l^2}{2}\right)\frac{\Lambda_l^2}{\mu_l(\mu_l-\Lambda_l)}+\frac{\Lambda_l}{\mu_l}-\alpha_l\Lambda_l+\alpha_l\sum_{i\in I}\lambda_i x_{il}^t\right\}=0$$

$$\Rightarrow \Lambda_l^t=\sum_{i\in I}\lambda_i x_{il}^t \tag{22}$$

$$\frac{\partial}{\partial \Lambda_l}\left\{\left(\frac{1+cv_l^2}{2}\right)\frac{\Lambda_l^2}{\mu_l(\mu_l-\Lambda_l)}+\frac{\Lambda_l}{\mu_l}-\alpha_l\Lambda_l+\alpha_l\sum_{i\in I}\lambda_i x_{il}^t\right\}=0$$

$$\Rightarrow \alpha_l^t=\left(\frac{1+cv_l^2}{2}\right)\frac{\Lambda_l^t(2\mu_l-\Lambda_l^t)}{\mu_l(\mu_l-\Lambda_l^t)^2}+\frac{1}{\mu_l} \tag{23}$$

Using the above expressions for Λ_l^t and α_l^t, [RMP] at iteration T' can be restated as:

$[RMP]$:

(15)

s.t.(9) − (12), (17)

$$\theta_l \geq \left\{\left(\frac{1+cv_l^2}{2}\right)\frac{\sum_{i\in I}\lambda_i x_{il}^{t\,2}}{\mu_l(\mu_l-\sum_{i\in I}\lambda_i x_{il}^t)}+\frac{\sum_{i\in I}\lambda_i x_{il}^t}{\mu_l}\right\}+$$
$$\left\{\left(\frac{1+cv_l^2}{2}\right)\frac{\sum_{i\in I}\lambda_i x_{il}^t(2\mu_l-\sum_{i\in I}\lambda_i x_{il}^t)}{\mu_l(\mu_l-\sum_{i\in I}\lambda_i x_{il}^t)^2}+\frac{1}{\mu_l}\right\}\sum_{i\in I}(\lambda_i x_{il}-\lambda_i x_{il}^t) \quad \forall l\in L; \forall t=1,...,T' \tag{24}$$

Since [RMP] at any given iteration is a relaxation to NKPCS, its optimal objective function value, $\nu(RMP)$, provides an upper bound (UB) to NKPCS. At the same time, since $\mathbf{x}^t$ at iteration t is a feasible solution to NKPCS, a lower bound (LB) to NKPCS is provided by:

$$LB^t = Z(\mathbf{x}^t,\mathbf{y}^t)=\sum_{i\in I}\sum_{l\in L}\pi_i x_{il}^t-\sum_{l\in L}k_l y_l^t - c\sum_{l\in L}\phi_l(\mathbf{x}_l^t, y_l^t) \tag{25}$$

The complete solution algorithm to solve NKPCS using GBD can be summarized using Algorithm 1. The form of Benders constraint (24) merits further discussion. Seen differently,

Algorithm 1 Generalized Benders Decomposition Based Algorithm to Solve NKPCS

1: $t \leftarrow 1$; $UB^{t-1} \leftarrow +\infty$; $LB^{t-1} \leftarrow 0$.
2: **repeat**
3: Solve [RMP], given by (15), (9)-(12), (17), (24), and obtain its optimal solution $(\mathbf{x}^t, \mathbf{y}^t)$.
4: Update UB: $UB^t \leftarrow \nu(RMP)$ using (15).
5: Update LB: $LB^t \leftarrow max\{LB^{t-1}, Z(\mathbf{x}^t, \mathbf{y}^t)\}$ using (25).
6: Add to [RMP] a new constraint using (24).
7: $t \leftarrow t + 1$
8: **until** $(UB^{t-1} - LB^{t-1})/UB^{t-1} \leq \epsilon$

the RHS of (24) is a linear outer approximation of the nonlinear part of the objective function in (1). Hence, the above GBD algorithm can also be viewed as an Outer Approximation (OA) algorithm (Floudas, 1995). GBD, as described above, has the drawback that it uses an expanded variable set x_{il} defined for each i and l pair, as opposed to the variable set x_i defined only for each i in the original model. In the next section, we present an alternate solution method that overcomes this drawback by working with the original variable set x_i.

3 Alternate Outer Approximation

We now present an alternate reformulation of (1) - (7), followed by a piecewise linear outer approximation of the resulting nonlinear function of an auxiliary variable. For this, we rewrite the nonlinear term in (1), given by (6), as:

$$E[N] = \left(\frac{1+cv^2}{2}\right)\frac{\Lambda^2}{\mu(\mu-\Lambda)} + \frac{\Lambda}{\mu} = \left(\frac{1+cv^2}{2}\right)\frac{\rho^2}{(1-\rho)} + \rho$$

where $\rho = \Lambda/\mu$. Upon rearranging the terms in the RHS, the above relation can be rewritten as:

$$E[N] = \frac{1}{2}\left\{(1+cv^2)\frac{\rho}{1-\rho} + (1-cv^2)\rho\right\} \tag{26}$$

To linearize the RHS in (26), we define a new variable U, such that:

$$U = \frac{\rho}{1-\rho} = \frac{\Lambda}{\mu} = \frac{\sum_{i \in I} \lambda_i x_i}{\sum_{l \in L} \mu_l y_l} \tag{27}$$

which implies:

$$\rho = \frac{U}{1+U} \tag{28}$$

Using $\rho = \Lambda/\mu$, Λ can be expressed as:

$$\Lambda = \sum_{i \in I} \lambda_i x_i = \rho\mu = \rho \sum_{l \in L} \mu_l y_l = \sum_{l \in L} \mu_l z_l \tag{29}$$

$$\text{where } z_l = \begin{cases} \rho y_l \text{ if } y_l = 1 \\ 0, \text{ otherwise} \end{cases}$$

Given that at most one capacity level can be selected, the above relation between z_l and y_l can be restated using the following sets of constraints:

$$\begin{aligned} z_l \leq y_l \quad \forall l \in L \\ \sum_l z_l = \rho \\ z_l \geq 0 \quad \forall l \in L \end{aligned}$$

With the above substitutions, the expression for E[N] in (1) reduces to:

$$\begin{aligned} E[N] &= \frac{1}{2}\left\{(1 + \sum_{l \in L} cv_l^2 y_l)U + (1 - \sum_{l \in L} cv_l^2 y_l)\rho\right\} \\ &= \frac{1}{2}\left\{U + \sum_{l \in L} cv_l^2 w_l + \rho - \sum_{l \in L} cv_l^2 z_l\right\} \end{aligned}$$

$$\text{where } w_l = \begin{cases} U \text{ if } y_l = 1 \\ 0, \text{ otherwise} \end{cases}$$

Again, exploiting the fact that $y_l = 1$ for at most one l, the above relation between w_l and y_l can be restated as:

$$w_l \leq My_l \quad \forall l \in L$$
$$\sum_l w_l = U$$
$$w_l \geq 0 \quad \forall l \in L$$

where M is a large number (called Big M).

Clearly, ρ, as given by (28), is a concave function in U. Hence, $\rho(U)$ can be approximated arbitrarily closely by piecewise linear functions that are tangent to ρ at points $\{U^h\}_{h \in H}$, such that:

$$\rho = \min_{h \in H} \left\{ \frac{U}{(1+U^h)^2} + \left(\frac{U^h}{1+U^h} \right)^2 \right\} \tag{30}$$

(30) is equivalent to:

$$\rho \leq \left\{ \frac{U}{(1+U^h)^2} + \left(\frac{U^h}{1+U^h} \right)^2 \right\} \quad \forall h \in H \tag{31}$$

Using the new variables ρ, U, $\mathbf{z}$ and $\mathbf{w}$, as introduced above, NKPCS can be reformulated as: $[NKPCS(H)]$:

$$\max_{\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{w},\rho,U} \sum_{i \in I} \pi_i x_i - \sum_{l \in L} k_l y_l - \frac{c}{2} \left\{ U + \rho + \sum_{l \in L} c v_l^2 (w_l - z_l) \right\} \tag{32}$$

$$s.t. \ (3) - (5), (31)$$

$$\sum_{i \in I} \lambda_i x_i = \sum_{l \in L} \mu_l z_l \tag{33}$$

$$z_l \leq y_l \quad \forall l \in L \tag{34}$$

$$\sum_{l \in L} z_l = \rho \tag{35}$$

$$w_l \leq M y_l \quad \forall l \in L \tag{36}$$

$$\sum_{l \in L} w_l = U \tag{37}$$

$$0 \leq \rho, z_l \leq 1; \quad U, w_l \geq 0 \quad \forall l \in L \tag{38}$$

Consider a relaxation $P(H^q) : H^q \subset H$. An upper bound to $P(H)$, and hence to NKPSC, is provided by the objective function value of $P(H^q)$, given by:

$$UB^q = \nu(NKPCS(H^q)) = \sum_{i \in I} \pi_i x_i^q - \sum_{l \in L} k_l y_l^q - \frac{c}{2} \left\{ U^q + \rho^q + \sum_{l \in L} cv_l (w_l^q - z_l^q) \right\} \tag{39}$$

where, $(\mathbf{x^q}, \mathbf{y^q}, \mathbf{z^q}, \mathbf{w^q}, \rho^q, U^q)$ is the optimal solution vector to $P(H^q)$. Further, since the optimal solution $(\mathbf{x^q}, \mathbf{y^q})$ to $P(H^q)$ is also a feasible solution to NKPCS, a lower bound to NKPCS is provided by the objective function of NKPCS evaluated at $(\mathbf{x^q}, \mathbf{y^q})$, given by:

$$\begin{aligned} LB^q =& Z(\mathbf{x^q}, \mathbf{y^q}) = \sum_{i \in I} \pi_i x_i^q - \sum_{l \in L} k_l y_l^q \\ & - c \left(\frac{1 + \sum_{l \in} cv_l^2 y_l^q}{2} \right) \frac{\left(\sum_{i \in I} \lambda_i x_i^q\right)^2}{\sum_{l \in L} \mu_l y_l^q (\sum_{l \in L} \mu_l y_l^q - \sum_{i \in I} \lambda_i x_i^q)} + \frac{\sum_{i \in I} \lambda_i x_i^q}{\sum_{l \in L} \mu_l y_l^q} \quad \text{if } \sum_{l \in L} y_l^q \geq 0 \end{aligned} \tag{40}$$

However, if $\sum_{l \in L} y_l^q = 0$, then $LB^q = 0$. The complete AOA based solution algorithm to solve NKPCS can be summarized as Algorithm 2.

AOA, as described in Algorithm 2, offers the possibility of arbitrarily selecting the initial set of points $\{U_h\}_{h \in H^1}$ to approximate the function $f(U) = U/(1+U)$. However, a careful selection of the initial set of points should help the algorithm converge faster (Elhedhli, 2005). Hence, we report our computational results in Section 4 for two different variants of AOA that differ in the choice of $\{U_h\}_{h \in H^1}$. In the first variant of AOA (referred to as AOA1), $H^1 \leftarrow \Phi$. In the second variant (referred to as AOA2), the initial set of points $\{U_h\}_{h \in H^1}$ is carefully chosen such that the outer approximation of $f(U)$, as given by (31), overestimates $f(U)$ by at most $\epsilon = 0.001$. The exact algorithm to select $\{U_h\}_{h \in H^1}$ for any arbitrary value

Algorithm 2 Alternate Outer Approximation Based Algorithm to Solve NKPCS

$q \leftarrow 1$; $UB^{q-1} \leftarrow +\infty$; $LB^{q-1} \leftarrow -\infty$.
Choose an initial set of points $\{U^h\}_{h \in H^q}$ to approximate the function $f(U) = U/(1+U)$ using (31).
repeat
    Solve $NKPCS(H^q)$, and obtain its optimal solution $(\mathbf{x^q}, \mathbf{y^q}, \mathbf{z^q}, \mathbf{w^q}, \rho^q, U^q)$.
    Update UB: $UB^q \leftarrow \nu(NKPSC(H^q))$ using (39).
    Update LB: $LB^q \leftarrow max\{LB^{q-1}, 0\}$ if $\sum_{l \in L} y_l^q = 0$. Else, $LB^q \leftarrow max\{LB^{q-1}, Z(\mathbf{x}^q, \mathbf{y}^q)\}$ using (40).
    Generate a new point $U^h = \frac{\sum_{i \in I} \lambda_i x_i^q}{\sum_{l \in L} \mu_l y_l^q - \sum_{i \in I} \lambda_i x_i^q}$.
    Add to $NKPCS(H^q)$ a new constraint of the form (31).
    $H^{q+1} \leftarrow H^q \cup \{h\}$.
    $q \leftarrow q + 1$
until $(UB^{q-1} - LB^{q-1})/UB^{q-1} \leq \epsilon$

of ϵ is presented by Elhedhli (2005).

4 Computational experience

We now present our computational experience with the two alternate solution approaches as described in sections 2 & 3. Both the solution algorithms (Algorithm 1 and Algorithm 2) are coded in C++ (using Visual Studio 2010), while RMP in Algorithm 1 and NKPCS(H) at each iteration in Algorithm 1 and Algorithm 2 respectively are solved using IBM ILOG CPLEX 21.5 on a PC with Intel Core i5-3470 CPU @ 3.20 GHz, 8 GB RAM, and Windows 64-bit operating system.

4.1 Data generation

The difficulty level of 0-1 NKPs is known to vary with the level of correlation between the weights of the objects and their profits (Martello et al., 2000; Elhedhli, 2005). As such, we test the efficacy of Algorithm 1 and Algorithm 2 for the following two different classes of problems instances based on the level of correlation between the object weights and profits.

- Weakly correlated instances: customer arrival rates are randomly generated as $\lambda_i \sim$

$U[1, R]$, and revenues are randomly generated as $\pi_i \sim U[\lambda_i - R/10, \lambda_i + R/10]$

- Strongly correlated instances: customer arrival rates are randomly generated as $\lambda_i \sim U[1, R]$, and revenues are set as $\pi_i = \lambda_i + R/10$

where R takes values from the set {1,000, 10,000}. The above two classes of data instances are generated for the following 4 number of demand nodes {500, 1,000, 5,000, 10,000}. The waiting penalty c is set as $a * R/10$, where a takes values from the set {0.1, 0.5, 1.0, 2.0}. The number of capacity levels is set as $L = 5$. The coefficient of variation of service times (cv_l) at capacity level l takes values from the set {0, 0.5, 1.0, 2.0}. The capacity (service rate) corresponding to capacity level $l = 3$ is set as $\mu_3 = (2 * b * \max_i \lambda_i)/10.1$, where b takes values from the set {1, 2, 3, ..., 10}. The service rates corresponding to the remaining 4 levels are set as $\mu_1 = 0.5\mu_3$, $\mu_2 = 0.75\mu_3$, $\mu_4 = 1.25\mu_3$, $\mu_5 = 1.5\mu_3$. The amortized capacity cost for capacity level $l = 3$ is set as $k_3 = \mu_3/10$. The capacity costs for the remaining 4 capacity levels are set as $k_1 = 0.60k_3$, $k_2 = 0.85k_3$, $k_4 = 0.15k_3$, $k_5 = 1.35k_3$. The capacity costs at different capacity levels are so chosen to exhibit both economies as well as diseconomies of scale in capacity addition.

4.2 Discussion of results

For each class of problem instances (weakly correlated and strongly correlated), we run the experiments for all combinations of the problem parameters, namely the number of nodes, R, a, cv and b. This gives us, for each class, a total of 1,280 (= 4x2x4x4x10) problem instances. Tables 1-4 present, for each combination of problem size, R, a and cv, the average over the 10 different values, of b of CPU times (in seconds) taken by GBD, AOA1 and AOA2. We use a run time limit of 2 hours (7,200 seconds) for each instance. For cases where some instances cannot be solved to optimality within the 2 hour time limit, we report the corresponding average optimality gap (GAP%) over the 10 different values of b.

Tables 1 & 3 clearly show that for problems instances of the size 500 and 1,000 nodes, both GBD and AOA (whether AOA1 or AOA2) are very efficient, solving all instances to

Table 1: Comparison between GBD and AOA with initial cuts (AOA1) and without initial cuts (AOA2) for weakly correlated data and small problem instances

				GBD			AOA1			AOA2		
Nodes	R	a	cv	Gap(%)	CPU(s)	Iter	Gap(%)	CPU	Iter	Gap(%)	CPU	Iter
500	1000	0.1	0	0.00	1.5	5	0.00	0.4	5	0.00	0.5	5
			0.5	0.00	1.6	6	0.00	0.5	6	0.00	0.5	6
			1	0.00	1.9	6	0.00	0.6	6	0.00	1.0	6
			2	0.00	2.3	7	0.00	0.6	7	0.00	0.6	7
		0.5	0	0.00	2.0	7	0.00	0.6	7	0.00	0.6	7
			0.5	0.00	1.8	6	0.00	0.5	6	0.00	0.5	6
			1	0.00	2.0	7	0.00	0.6	7	0.00	0.6	7
			2	0.00	2.6	8	0.00	0.7	8	0.00	0.6	8
		1.0	0	0.00	2.4	7	0.00	0.7	7	0.00	0.6	7
			0.5	0.00	2.2	7	0.00	0.6	7	0.00	0.6	7
			1	0.00	2.2	7	0.00	0.6	7	0.00	0.7	7
			2	0.00	3.8	12	0.00	0.8	9	0.00	0.8	9
		2.0	0	0.00	2.3	7	0.00	0.6	7	0.00	0.6	7
			0.5	0.00	2.6	8	0.00	0.7	8	0.00	0.7	8
			1	0.00	2.7	9	0.00	0.7	8	0.00	0.7	8
			2	0.00	5.1	14	0.00	0.8	9	0.00	0.8	9
	10000	0.1	0	0.00	2.5	7	0.00	0.6	7	0.00	0.6	7
			0.5	0.00	2.0	6	0.00	0.5	6	0.00	0.5	6
			1	0.00	2.4	7	0.00	0.6	7	0.00	0.6	7
			2	0.00	2.4	7	0.00	0.6	7	0.00	0.6	7
		0.5	0	0.00	2.2	6	0.00	0.5	6	0.00	0.6	6
			0.5	0.00	2.7	7	0.00	0.6	7	0.00	0.6	7
			1	0.00	2.5	7	0.00	0.7	7	0.00	0.7	7
			2	0.00	3.2	9	0.00	0.7	9	0.00	0.7	9
		1.0	0	0.00	2.7	7	0.00	0.6	7	0.00	0.6	7
			0.5	0.00	2.6	7	0.00	0.7	7	0.00	0.6	7
			1	0.00	3.8	9	0.00	0.8	9	0.00	0.8	9
			2	0.00	4.3	12	0.00	0.8	9	0.00	0.8	9
		2.0	0	0.00	3.4	10	0.00	1.0	10	0.00	0.9	10
			0.5	0.00	3.0	9	0.00	0.8	9	0.00	0.7	9
			1	0.00	3.4	10	0.00	0.8	8	0.00	0.7	8
			2	0.00	6.5	15	0.00	0.9	10	0.00	0.8	10
1000	1000	0.1	0	0.00	5.8	6	0.00	0.7	6	0.00	0.7	6
			0.5	0.00	5.9	6	0.00	0.8	6	0.00	0.8	6
			1	0.00	6.5	7	0.00	0.9	7	0.00	0.8	7
			2	0.00	9.4	7	0.00	1.0	7	0.00	0.9	7
		0.5	0	0.00	8.3	8	0.00	1.1	8	0.00	1.0	8
			0.5	0.00	7.8	8	0.00	1.0	8	0.00	1.0	8
			1	0.00	8.4	8	0.00	1.1	8	0.00	1.0	8
			2	0.00	8.6	9	0.00	1.2	9	0.00	1.1	9
		1.0	0	0.00	8.7	8	0.00	1.0	8	0.00	1.0	8
			0.5	0.00	9.4	8	0.00	1.1	8	0.00	1.0	8
			1	0.00	8.1	8	0.00	1.1	8	0.00	1.0	8
			2	0.00	10.2	10	0.00	1.2	9	0.00	1.1	9
		2.0	0	0.00	9.8	8	0.00	1.1	8	0.00	1.0	8
			0.5	0.00	9.0	9	0.00	1.1	9	0.00	1.1	9
			1	0.00	9.2	10	0.00	1.2	9	0.00	1.1	9
			2	0.00	16.2	16	0.00	1.4	11	0.00	1.3	11
	10000	0.1	0	0.00	9.2	7	0.00	0.9	7	0.00	0.9	7
			0.5	0.00	8.4	7	0.00	0.9	7	0.00	0.8	7
			1	0.00	9.5	7	0.00	1.0	7	0.00	1.0	7
			2	0.00	9.4	8	0.00	1.1	8	0.00	1.0	8
		0.5	0	0.00	11.3	8	0.00	1.1	8	0.00	1.0	8
			0.5	0.00	9.1	9	0.00	1.2	9	0.00	1.1	9
			1	0.00	10.7	9	0.00	1.2	9	0.00	1.1	9
			2	0.00	12.7	10	0.00	1.2	10	0.00	1.2	10
		1.0	0	0.00	9.4	9	0.00	1.1	8	0.00	1.0	8
			0.5	0.00	8.9	9	0.00	1.0	9	0.00	1.0	9
			1	0.00	11.5	10	0.00	1.3	10	0.00	1.2	10
			2	0.00	12.9	12	0.00	1.4	11	0.00	1.4	11
		2.0	0	0.00	11.0	9	0.00	1.2	9	0.00	1.2	9
			0.5	0.00	13.1	10	0.00	1.3	11	0.00	1.3	11
			1	0.00	10.0	10	0.00	1.2	10	0.00	1.2	10
			2	0.00	19.9	16	0.00	1.3	11	0.00	1.3	11

Table 2: Comparison between GBD and Piecewise Linearization with initial cuts (AOA1) and without initial cuts (AOA2) for weakly correlated data and large problem instances

				GBD			AOA1			AOA2		
Nodes	R	a	cv	Gap(%)	CPU(s)	Iter	Gap(%)	CPU	Iter	Gap(%)	CPU	Iter
5000	1000	0.1	0	0.00	56.9	8	0.00	2.2	7	0.00	2.4	8
			0.5	0.00	56.0	7	0.00	2.3	8	0.00	2.0	7
			1	0.00	58.9	8	0.00	2.4	8	0.00	2.2	8
			2	0.00	73.8	9	0.00	2.8	9	0.00	2.6	9
		0.5	0	0.00	62.1	9	0.00	2.8	9	0.00	2.7	9
			0.5	0.00	68.7	9	0.00	2.8	9	0.00	2.6	9
			1	0.00	69.0	9	0.00	2.9	10	0.00	2.7	9
			2	0.00	76.8	10	0.00	3.1	11	0.00	2.9	10
		1.0	0	0.00	70.7	10	0.00	2.8	10	0.00	2.8	10
			0.5	0.00	72.8	10	0.00	3.2	11	0.00	2.9	10
			1	0.00	72.1	11	0.00	3.1	11	0.00	3.1	10
			2	0.00	81.0	11	0.00	3.4	12	0.00	3.2	11
		2.0	0	0.00	72.4	10	0.00	3.2	10	0.00	3.1	11
			0.5	0.00	73.9	11	0.00	3.1	10	0.00	3.0	11
			1	0.00	80.8	11	0.00	3.3	11	0.00	3.3	11
			2	0.00	85.6	13	0.00	3.6	12	0.00	3.6	12
	10000	0.1	0	0.00	77.0	9	0.00	2.6	9	0.00	2.5	9
			0.5	0.00	71.2	9	0.00	2.7	9	0.00	2.7	9
			1	0.00	64.4	9	0.00	2.7	9	0.00	2.5	9
			2	0.00	78.0	10	0.00	2.9	10	0.00	2.8	10
		0.5	0	0.00	80.4	10	0.00	2.9	10	0.00	2.7	10
			0.5	0.00	89.0	11	0.00	3.3	11	0.00	2.8	10
			1	0.00	84.6	11	0.00	3.1	11	0.00	2.9	11
			2	0.00	95.5	11	0.00	3.2	12	0.00	3.2	12
		1.0	0	0.00	93.1	11	0.00	3.2	11	0.00	3.1	11
			0.5	0.00	103.6	11	0.00	3.2	11	0.00	3.0	11
			1	0.00	86.6	12	0.00	3.3	12	0.00	3.2	12
			2	0.00	93.3	12	0.00	3.9	13	0.00	3.4	12
		2.0	0	0.00	100.4	11	0.00	3.6	12	0.00	3.3	12
			0.5	0.00	95.0	12	0.00	3.5	12	0.00	3.1	11
			1	0.00	88.9	12	0.00	3.5	12	0.00	3.2	12
			2	0.00	107.4	14	0.00	3.8	13	0.00	3.4	13
10000	1000	0.1	0	0.00	177.3	8	0.00	4.5	8	0.00	4.1	8
			0.5	0.00	222.3	8	0.00	5.0	8	0.00	4.6	8
			1	0.00	191.4	8	0.00	3.9	8	0.00	4.1	8
			2	0.00	222.3	9	0.00	5.4	9	0.00	5.1	9
		0.5	0	0.00	237.7	9	0.00	5.5	9	0.00	5.0	9
			0.5	0.00	234.3	10	0.00	5.6	9	0.00	5.2	10
			1	0.00	220.2	9	0.00	6.1	10	0.00	5.1	9
			2	0.00	253.4	11	0.00	6.6	11	0.00	6.0	11
		1.0	0	0.00	225.7	9	0.00	5.9	10	0.00	5.4	10
			0.5	0.00	241.0	10	0.00	6.2	10	0.00	5.4	10
			1	0.00	256.1	10	0.00	6.2	10	0.00	5.7	10
			2	0.00	253.6	11	0.00	6.5	11	0.00	6.2	11
		2.0	0	0.00	244.6	10	0.00	6.1	10	0.00	5.7	11
			0.5	0.00	254.6	11	0.00	6.8	11	0.00	6.2	11
			1	0.00	264.9	12	0.00	6.7	11	0.00	6.1	12
			2	0.00	292.7	12	0.00	6.6	12	0.00	5.9	12
	10000	0.1	0	0.00	196.6	9	0.00	4.1	9	0.00	4.4	9
			0.5	0.00	215.0	10	0.00	4.7	9	0.00	4.1	9
			1	0.00	206.3	9	0.00	4.7	9	0.00	4.2	10
			2	0.00	258.9	10	0.00	4.9	10	0.00	4.7	10
		0.5	0	0.00	249.0	10	0.00	5.6	11	0.00	4.9	11
			0.5	0.00	279.9	11	0.00	5.5	11	0.00	4.9	11
			1	0.00	274.0	11	0.00	5.6	11	0.00	5.6	11
			2	0.00	286.7	12	0.00	5.8	12	0.00	5.8	12
		1.0	0	0.00	244.3	11	0.00	5.6	11	0.00	5.8	11
			0.5	0.00	257.7	11	0.00	5.3	11	0.00	5.4	11
			1	0.00	288.9	12	0.00	6.3	12	0.00	5.6	12
			2	0.00	370.3	13	0.00	6.2	12	0.00	6.2	13
		2.0	0	0.00	301.5	12	0.00	6.4	12	0.00	5.8	12
			0.5	0.00	297.7	12	0.00	6.7	12	0.00	5.3	12
			1	0.00	345.7	12	0.00	6.2	13	0.00	6.0	13
			2	0.00	322.5	14	0.00	6.6	13	0.00	6.1	13

Table 3: Comparison between GBD and Piecewise Linearization with initial cuts (AOA1) and without initial cuts (AOA2) for strongly correlated data and small problem instances

				GBD			AOA1			AOA2		
Nodes	R	a	cv	Gap(%)	CPU(s)	Iter	Gap(%)	CPU	Iter	Gap(%)	CPU	Iter
500	1000	0.1	0	0.00	5.4	10	0.00	2.0	10	0.00	1.8	10
			0.5	0.00	4.3	9	0.00	1.8	10	0.00	1.8	10
			1	0.00	4.6	10	0.00	1.9	10	0.00	1.8	10
			2	0.00	4.8	11	0.00	1.9	10	0.00	1.9	10
		0.5	0	0.00	4.6	11	0.00	1.9	10	0.00	1.9	10
			0.5	0.00	4.5	11	0.00	1.8	10	0.00	2.0	10
			1	0.00	9.1	17	0.00	2.3	11	0.00	1.9	11
			2	0.00	12.9	21	0.00	3.6	12	0.00	3.3	12
		1.0	0	0.00	8.7	17	0.00	2.1	11	0.00	1.9	11
			0.5	0.00	14.7	18	0.00	2.1	11	0.00	1.9	11
			1	0.00	12.3	19	0.00	3.1	12	0.00	3.5	12
			2	0.00	18.4	28	0.00	3.5	12	0.00	2.9	12
		2.0	0	0.00	11.5	19	0.00	3.3	12	0.00	2.7	12
			0.5	0.00	11.0	20	0.00	4.2	12	0.00	2.9	12
			1	0.00	14.0	24	0.00	3.7	12	0.00	3.0	12
			2	0.00	21.4	33	0.00	3.7	13	0.00	4.0	13
	10000	0.1	0	0.00	6.9	12	0.00	1.6	12	0.00	1.6	12
			0.5	0.00	7.6	13	0.00	1.9	13	0.00	1.8	13
			1	0.00	6.9	12	0.00	1.7	12	0.00	1.6	12
			2	0.00	8.3	15	0.00	1.8	13	0.00	1.7	13
		0.5	0	0.00	8.4	16	0.00	1.9	13	0.00	1.7	13
			0.5	0.00	9.2	17	0.00	1.8	13	0.00	1.8	13
			1	0.00	20.1	22	0.00	2.0	14	0.00	1.9	14
			2	0.00	19.4	27	0.00	2.1	15	0.00	1.9	15
		1.0	0	0.00	19.5	22	0.00	2.1	14	0.00	2.0	14
			0.5	0.00	19.8	23	0.00	2.0	14	0.00	1.9	14
			1	0.00	18.9	25	0.00	2.0	14	0.00	1.9	14
			2	0.00	28.8	35	0.00	2.1	16	0.00	2.0	16
		2.0	0	0.00	18.3	25	0.00	2.1	15	0.00	2.0	15
			0.5	0.00	19.6	26	0.00	2.1	15	0.00	2.0	15
			1	0.00	23.3	32	0.00	2.1	15	0.00	2.0	15
			2	0.00	34.2	41	0.00	2.1	16	0.00	2.1	16
1000	1000	0.1	0	0.00	7.2	9	0.00	4.0	9	0.00	3.8	9
			0.5	0.00	7.5	9	0.00	2.9	9	0.00	3.3	9
			1	0.00	10.0	9	0.00	4.1	9	0.00	3.1	9
			2	0.00	8.2	11	0.00	3.5	10	0.00	3.1	10
		0.5	0	0.00	7.8	11	0.00	3.8	11	0.00	3.6	11
			0.5	0.00	9.3	12	0.00	15.6	11	0.00	11.8	11
			1	0.00	12.6	15	0.00	3.6	11	0.00	3.6	11
			2	0.00	22.9	20	0.00	3.9	12	0.00	3.3	12
		1.0	0	0.00	27.0	14	0.00	3.7	11	0.00	4.3	11
			0.5	0.00	40.2	17	0.00	15.1	11	0.00	15.3	11
			1	0.00	21.6	19	0.00	3.7	12	0.00	3.8	12
			2	0.00	57.6	26	0.00	4.1	13	0.00	3.8	13
		2.0	0	0.00	18.0	20	0.00	4.2	12	0.00	4.2	12
			0.5	0.00	27.1	19	0.00	4.0	12	0.00	4.5	12
			1	0.00	24.5	23	0.00	6.0	13	0.00	5.6	13
			2	0.00	50.3	32	0.00	4.5	14	0.00	5.8	14
	10000	0.1	0	0.00	12.7	12	0.00	2.1	12	0.00	2.0	12
			0.5	0.00	12.9	12	0.00	2.1	12	0.00	2.1	12
			1	0.00	13.5	12	0.00	2.2	12	0.00	2.1	12
			2	0.00	16.4	14	0.00	2.4	14	0.00	2.3	14
		0.5	0	0.00	14.7	14	0.00	2.4	14	0.00	2.3	14
			0.5	0.00	14.2	15	0.00	2.4	14	0.00	2.3	14
			1	0.00	25.2	19	0.00	2.4	14	0.00	2.4	14
			2	0.00	37.7	27	0.00	2.5	15	0.00	2.5	15
		1.0	0	0.00	23.3	19	0.00	2.5	14	0.00	2.4	14
			0.5	0.00	34.1	22	0.00	2.5	15	0.00	2.4	15
			1	0.00	42.2	26	0.00	2.5	15	0.00	2.5	15
			2	0.00	59.9	34	0.00	2.7	16	0.00	2.7	16
		2.0	0	0.00	106.1	26	0.00	2.6	15	0.00	2.5	15
			0.5	0.00	41.1	26	0.00	2.6	15	0.00	2.5	15
			1	0.00	46.1	29	0.00	2.6	16	0.00	2.6	16
			2	0.00	97.7	41	0.00	2.8	17	0.00	2.8	17

Table 4: Comparison between GBD and Piecewise Linearization with initial cuts (AOA1) and without initial cuts (AOA2) for strongly correlated data and large problem instances

				GBD			AOA1			AOA2		
Nodes	R	a	cv	Gap(%)	CPU(s)	Iter	Gap(%)	CPU	Iter	Gap(%)	CPU	Iter
5000	1000	0.1	0	0.00	7.3	8	0.00	1.2	9	0.00	1.1	9
			0.5	0.00	7.0	9	0.00	1.2	9	0.00	1.2	9
			1	0.00	7.7	9	0.00	1.3	9	0.00	1.2	9
			2	0.00	8.1	10	0.00	1.5	10	0.00	1.3	10
		0.5	0	0.00	8.3	10	0.00	1.5	10	0.00	1.3	10
			0.5	0.00	8.3	11	0.00	1.5	10	0.00	1.4	10
			1	0.00	9.8	12	0.00	1.6	11	0.00	1.5	11
			2	0.00	19.4	19	0.00	1.8	12	0.00	1.7	12
		1.0	0	0.00	9.1	12	0.00	1.5	11	0.00	1.5	11
			0.5	0.00	10.2	13	0.00	1.6	11	0.00	1.5	11
			1	0.00	17.0	17	0.00	1.6	12	0.00	1.5	12
			2	0.00	23.4	22	0.00	2.0	13	0.00	1.8	13
		2.0	0	0.00	16.9	17	0.00	1.7	12	0.00	1.5	12
			0.5	0.00	20.2	19	0.00	1.8	12	0.00	1.6	12
			1	0.00	22.5	21	0.00	1.8	13	0.00	1.7	13
			2	0.00	34.4	28	0.00	2.0	14	0.00	1.9	14
	10000	0.1	0	0.00	59.3	12	0.00	3.0	12	0.00	3.0	12
			0.5	0.00	62.1	12	0.00	3.1	12	0.00	3.1	12
			1	0.00	145.9	12	0.00	3.1	12	0.00	3.0	12
			2	0.00	64.8	13	0.00	3.4	13	0.00	3.3	13
		0.5	0	2.60*	849.93*	12.56*	0.00	3.4	13	0.00	3.4	13
			0.5	0.00	63.1	14	0.00	3.5	14	0.00	3.4	14
			1	0.00	658.2	15	0.00	3.5	14	0.00	3.4	14
			2	0.19*	1687.03*	23.67*	0.00	3.8	15	0.00	3.8	15
		1.0	0	0.00	67.6	15	0.00	3.6	14	0.00	3.5	14
			0.5	0.00	104.2	16	0.00	3.7	14	0.00	3.6	14
			1	0.10	1033.1	20	0.00	3.7	15	0.00	3.7	15
			2	0.84	1641.3	27	0.00	4.1	17	0.00	4.1	17
		2.0	0	10.25	2345.2	18	0.00	3.8	15	0.00	3.8	15
			0.5	0.74	3633.9	21	0.00	3.9	15	0.00	3.8	15
			1	1.50	2281.5	24	0.00	3.8	16	0.00	3.7	16
			2	2.06	1608.2	33	0.00	4.3	17	0.00	4.2	17
10000	1000	0.1	0	0.00	7.0	8	0.00	1.3	8	0.00	1.3	8
			0.5	0.00	7.4	9	0.00	1.4	9	0.00	1.4	9
			1	0.00	7.8	9	0.00	1.5	9	0.00	1.4	9
			2	0.00	8.1	10	0.00	1.6	10	0.00	1.6	10
		0.5	0	0.00	8.2	10	0.00	1.6	10	0.00	1.6	10
			0.5	0.00	8.8	10	0.00	1.7	11	0.00	1.7	11
			1	0.00	9.9	11	0.00	1.7	11	0.00	1.7	11
			2	0.00	19.4	17	0.00	1.9	12	0.00	1.9	12
		1.0	0	0.00	9.4	11	0.00	1.7	11	0.00	1.7	11
			0.5	0.00	10.1	12	0.00	1.8	11	0.00	1.8	11
			1	0.00	14.9	15	0.00	1.9	12	0.00	1.9	12
			2	0.00	27.7	21	0.00	2.1	13	0.00	2.0	13
		2.0	0	0.00	14.7	15	0.00	1.9	12	0.00	1.8	12
			0.5	0.00	20.0	17	0.00	1.9	12	0.00	1.9	12
			1	0.00	24.8	20	0.00	2.0	13	0.00	2.0	13
			2	0.00	35.9	26	0.00	2.2	14	0.00	2.1	14
	10000	0.1	0	0.00	286.0	11	0.00	4.1	11	0.00	4.1	11
			0.5	0.98	995.9	11	0.00	4.3	12	0.00	4.2	12
			1	0.00	158.5	12	0.00	4.5	12	0.00	4.5	12
			2	10.00	1028.2	13	0.00	4.9	13	0.00	4.7	13
		0.5	0	0.00	274.5	13	0.00	5.0	13	0.00	4.8	13
			0.5	4.90	824.4	13	0.00	5.0	13	0.00	4.8	13
			1	0.00	189.7	15	0.00	5.2	14	0.00	4.9	14
			2	10.60	3503.9	18	0.00	5.6	15	0.00	5.2	15
		1.0	0	0.00	154.3	15	0.00	5.4	14	0.00	5.1	14
			0.5	0.00	190.2	15	0.00	5.4	15	0.00	5.3	15
			1	0.15	1602.7	18	0.00	5.5	15	0.00	5.3	15
			2	12.91	5130.7	19	0.00	6.0	16	0.00	5.6	16
		2.0	0	10.21	2258.1	15	0.00	5.7	15	0.00	5.6	15
			0.5	10.54	3122.0	18	0.00	5.7	16	0.00	5.4	16
			1	11.47	3814.8	20	0.00	5.7	16	0.00	5.4	16
			2	7.52	4520.1	23	0.00	6.1	17	0.00	5.9	17

*average computed only over 9 instances since 1 instance terminated with the error message "integer solution contains unscaled infeasibilities" optimality in seconds. However, for larger problem instances of the size 5,000 and 10,000 nodes (see Tables 2 & 4), while AOA (both AOA1 and AOA2) is still very efficient, solving all instances in less than 10 seconds on average, GBD starts to struggle for many instances, leaving a substantial optimality gap even after 2 hours of CPU time. Further, GBD finds strongly correlated instances (see Tables 3 & 4) significantly more difficult (as reflected in larger CPU times or optimality gaps) than weakly correlated instances (see Tables 1 & 2). This is consistent with the general findings in the literature on NKPs (Martello et al., 2000). However, AOA (whether AOA1 and AOA2) exhibits little change in performance between weakly and strongly correlated instances. Further, for NKPCS, use of initial set of cuts of the form (31) has little impact on the performance of AOA. This is in contrast with the observation by Elhedhli (2005) for NKPs (without capacity selection decision).

References

Amiri, A. (1997). Solution procedures for the service system design problem. *Computers & operations research*, 24(1):49–60.

Amiri, A. (1998). The design of service systems with queueing time cost, workload capacities and backup service. *European Journal of Operational Research*, 104(1):201–217.

Berman, O. and Krass, D. (2002). 11 facility location problems with stochastic demands and congestion. *Facility location: applications and theory*, pages 329–371.

Bretthauer, K. M. and Shetty, B. (2002). The nonlinear knapsack problem–algorithms and applications. *European Journal of Operational Research*, 138(3):459–472.

Caprara, A., Pisinger, D., and Toth, P. (1999). Exact solution of the quadratic knapsack problem. *INFORMS Journal on Computing*, 11(2):125–137.

Dussault, J.-P., Ferland, J. A., and Lemaire, B. (1986). Convex quadratic programming with one constraint and bounded variables. *Mathematical Programming*, 36(1):90–104.

Elhedhli, S. (2005). Exact solution of a class of nonlinear knapsack problems. *Operations research letters*, 33(6):615–624.

Floudas, C. A. (1995). *Nonlinear and mixed-integer optimization: fundamentals and applications*. Oxford University Press on Demand.

Gallo, G., Hammer, P. L., and Simeone, B. (1980). Quadratic knapsack problems. In *Combinatorial Optimization*, pages 132–149. Springer.

Klastorin, T. (1990). On a discrete nonlinear and nonseparable knapsack problem. *Operations Research Letters*, 9(4):233–237.

Martello, S., Pisinger, D., and Toth, P. (2000). New trends in exact algorithms for the 0–1 knapsack problem. *European Journal of Operational Research*, 123(2):325–332.

Moré, J. J. and Vavasis, S. A. (1990). On the solution of concave knapsack problems. *Mathematical programming*, 49(1-3):397–411.