

Approximating the solution for the
multiparametric 0-1-mixed integer linear
programming problem with interval data

Alejandro Crema, Edgar Hugo Peraza,
Fernando Crema

August 2012

Alejandro Crema

Escuela de Computación, Facultad de Ciencias, Universidad Central de Venezuela,
Caracas, Venezuela.
email: alejandro.crema@ciens.ucv.ve

Edgar Hugo Peraza

Ingeniera Agro-Industrial, Universidad Central Occidental Lisandro Alvarado,
Barquisimeto, Venezuela.
email: eperaza@ucla.edu.ve

Fernando Crema

Escuela de Computación, Facultad de Ciencias, Universidad Central de Venezuela,
Caracas, Venezuela.
email: fcremarm@hotmail.com

Abstract: In this paper we present algorithms to approximate the solution for the multiparametric 0-1-mixed integer linear programming problem relative to the objective function. We consider the uncertainty for the parameters that define the cost vector corresponding to a subset of 0-1-variables by assuming that each parameter belongs to a known interval. We suppose that we have enough time to obtain an ϵ -optimal multiparametric solution. Then, when the true cost vector becomes known we can obtain an ϵ -optimal solution quickly. Our algorithms work by solving an appropriate finite sequence of non-parametric problems in such a manner that the solutions of the problems in the sequence provide us with an ϵ -optimal multiparametric solution.

Keywords: Integer programming, multiparametric programming, real time.

1 Introduction

Let $\Omega = \{f : f \in \mathbb{R}^n, l \leq f \leq u\}$. Let $f \in \Omega$. Let $P(f)$ be a problem in (x, y) defined as:

$$\max c^t x + f^t y \quad s.t. \quad Ax + By \leq b, \quad x \in X, \quad y \in \{0, 1\}^n$$

where $c \in \mathbb{R}^p$, $b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times p}$, $B \in \mathbb{R}^{m \times n}$ and $X \subseteq \mathbb{R}^p$. We consider two alternatives in our computational experience: either $X = \{x : x \in \{0, 1\}^p\}$ or $X = \{x : x \in \mathbb{R}^p, x \geq 0\}$.

We use the following standard notation: if T is a maximization problem then $F(T)$ denotes its set of feasible solutions and $v(T)$ denotes its optimal value (if $F(T) = \emptyset$ then $v(T) = -\infty$). Since $F(P(f^1)) = F(P(f^2)) \quad \forall f^1, f^2 \in \Omega$ we will use $F(P)$ instead of $F(P(f))$. We suppose that $F(P)$ is nonempty. Note that $P(f)$ may be either a 0-1-mixed integer linear programming (0-1-MIP) problem or a 0-1-integer linear programming (0-1-IP) problem.

The multiparametric analysis may be considered in the presence of uncertainty in the data. We consider the uncertainty relative to the cost vector corresponding to the y -variables in such a manner that each parameter belongs to a known interval. In this paper the multiparametric problem relative to the objective function is a family of problems in which a member is $P(f)$ where $f \in \Omega$.

Let $\epsilon \geq 0$. We say that $(x^{(1)}, y^{(1)}), \dots, (x^{(l)}, y^{(l)})$ is an ϵ -optimal multiparametric solution if: $(x^{(i)}, y^{(i)}) \in F(P)$ for all $i = 1, \dots, l$ and

$$v(P(f)) \geq \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, l\} \geq v(P(f)) - \epsilon \quad \forall f \in \Omega$$

Let $\lambda \in (0, 1)$. Note that if $v(P(l)) > 0$ and $\epsilon = \lambda v(P(l))$ then

$$v(P(f)) \geq \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, l\} \geq v(P(f))(1 - \lambda) \quad \forall f \in \Omega$$

Thus, ϵ and λ may be interpreted as the *absolute* and *relative* error respectively.

We suppose that we have enough time to obtain an ϵ -optimal multiparametric solution. Then, when the true cost vector f becomes known we can obtain an ϵ -optimal solution for $P(f)$ quickly, by using $v(P(f)) \approx \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, l\}$, which is very useful in real time optimization. Also, the ϵ -optimal solution for $P(f)$ may be used to guide a reoptimization. Even if the structure of $P(f)$ enables us to solve it quickly, the multiparametric analysis may be useful for systematically analyzing the effect of uncertainty.

In [1] an algorithm to approximate the stability region of an optimal solution for $P(f)$ for some f fixed is developed. In [2] the parametric case is considered

when a single parameter affects the objective function, the constraints matrix and the right hand side vector simultaneously. In [3] an algorithm is presented, and illustrated with small examples, for the general multiparametric case, with uncertain parameters in the objective function, the constraints matrix and the right hand side vector simultaneously. In [4] the multiparametric case is considered with the parameters in the right-hand-side vector.

Our algorithms work by solving an appropriate finite sequence of non-parametric problems in such a manner that the solutions of the problems in the sequence provide us with an ϵ -optimal multiparametric solution. This kind of approach was introduced in [5] for the single parametric case and was used in [6] for the multiparametric 0-1-IP problem relative to the objective function.

The rest of the paper is organized as follows. Section 2 presents the theoretical results and algorithms. Section 3 gives computational results for the Fixed-Charge Knapsack Problem (FCHKP), the Fixed-Charge Multiple Knapsack Problem (FCHMKP) and the Simple PLant Location Problem (SPLP). Section 4 has a summary and some ideas for future research.

2 Theoretical results and algorithms

Suppose that $(x^{(1)}, y^{(1)}), \dots, (x^{(r)}, y^{(r)})$ are given such that: $(x^{(i)}, y^{(i)}) \in F(P)$ for all $i = 1, \dots, r$. If $(x^{(1)}, y^{(1)}), \dots, (x^{(r)}, y^{(r)})$ is not an ϵ -optimal multiparametric solution then we must be able to find $f \in \Omega$ such that:

$$v(P(f)) - \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, r\} > \epsilon.$$

Let $Q^{(r)}$ be a problem in $((x, y), f)$ defined as:

$$\begin{aligned} \max \quad & c^t x + f^t y - \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, r\} \quad s.t. \\ & f \in \Omega, \quad (x, y) \in F(P) \end{aligned}$$

Note that with $Q^{(r)}$ we are looking for the maximal difference between $v(P(f))$ and $\max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, r\}$.

$Q^{(r)}$ was considered previously in [6] for the multiparametric 0-1-IP problem. The results presented in [6] may be generalized easily for the case presented in this paper and we present a summary without a proof:

Remark 1 *By construction of $Q^{(r)}$ we have:*

$$(i) \quad F(Q^{(r)}) \neq \emptyset.$$

(ii) *There exists an optimal solution.*

(iii) *If $((x, y), f)$ is an optimal solution for $Q^{(r)}$ then (x, y) is an optimal solution for $P(f)$.*

(iv) $v(Q^{(r)}) \geq 0$.

(v) $v(Q^{(i)}) \geq v(Q^{(i+1)}) \quad i = 1, \dots, r-1$

(vi) $v(Q^{(r)}) \leq \epsilon$ if and only if $(x^{(1)}, y^{(1)}), \dots, (x^{(r)}, y^{(r)})$ is an ϵ -optimal multiparametric solution.

The presence of the nonlinear term $f^t y$ in the objective function implies that in order to rewrite the problem as a 0-1-MIP we need to use w , an additional vector of variables, to write $\sum_{j=1}^n w_j$ instead of $f^t y$ and $4n$ additional constraints, by using an standard linearization method, in order to ensure that $w_j = f_j y_j \quad \forall j$.

$Q^{(r)}$ may be rewritten as a 0-1-MIP in $((x, y), z, w, f)$ as follows:

$$\max c^t x + \sum_{j=1}^n w_j - z \quad s.t.$$

$$f \in \Omega, \quad (x, y) \in F(P)$$

$$z \geq c^t x^{(i)} + f^t y^{(i)} \quad (i = 1, \dots, r)$$

$$l_j y_j \leq w_j \leq u_j y_j \quad (j = 1, \dots, n)$$

$$l_j(1 - y_j) \leq f_j - w_j \leq u_j(1 - y_j) \quad (j = 1, \dots, n)$$

Let $y \in [0, 1]^n$. Let $f^+(y)$ be defined as follows: $f^+(y)_j = u_j y_j + l_j(1 - y_j)$. Note that $f^+(y) \in \Omega$. If $y \in \{0, 1\}^n$ then $f^+(y)$ is known as *the most favorable scenario for y* . If $y, \hat{y} \in \{0, 1\}$ let $J_{0,1} = \{j \in \{1, \dots, n\} : y_j = 0, \hat{y}_j = 1\}$ and $J_{1,0} = \{j \in \{1, \dots, n\} : y_j = 1, \hat{y}_j = 0\}$.

The following lemma ensures that we can consider a discrete subset of Ω .

Lemma 1 *Let $(x, y) \in F(P)$. If (x, y) is not an optimal solution for $P(f^+(y))$ then (x, y) is not an optimal solution for $P(f)$ for all $f \in \Omega$.*

Proof:

Let us suppose that (x, y) is not an optimal solution for $P(f^+(y))$. Let (\hat{x}, \hat{y}) be an optimal solution for $P(f^+(y))$. Since $c^t \hat{x} + f^+(y)^t \hat{y} > c^t x + f^+(y)^t y$ then $c^t \hat{x} + \sum_{J_{0,1}} l_j > c^t x + \sum_{J_{1,0}} u_j$, therefore if $f \in \Omega$ we have that

$$c^t \hat{x} + f^t \hat{y} - c^t x - f^t y = c^t \hat{x} + \sum_{J_{0,1}} f_j - c^t x - \sum_{J_{1,0}} f_j \geq$$

$$c^t \hat{x} + \sum_{J_{0,1}} l_j - c^t x - \sum_{J_{1,0}} u_j > 0$$

It follows that (x, y) is not an optimal solution for $P(f)$.•

Corollary 1 *(x, y) is an optimal solution for $P(f)$ for some $f \in \Omega$ if and only if (x, y) is an optimal solution for $P(f^+(y))$.*

Therefore, we can define a new problem by using only scenarios that belong to Ω^+ where $\Omega^+ = \{f : f = f^+(y) \text{ for some } y \in \{0, 1\}^n\}$.

Let $Q^{+(r)}$ be a problem in (x, y) defined as:

$$\max c^t x + f^+(y)^t y - \max\{c^t x^{(i)} + f^+(y)^t y^{(i)} : i = 1, \dots, r\} \quad s.t.$$

$$(x, y) \in F(P)$$

Later we will prove the equivalence between $Q^{+(r)}$ and $Q^{(r)}$.

Note that $Q^{+(r)}$ may be rewritten as a 0-1-MIP problem in $((x, y), z)$ as follows:

$$\max \quad c^t x + f^+(y)^t y - z \quad s.t.$$

$$z \geq c^t x^{(i)} + f^+(y)^t y^{(i)} \quad (i = 1, \dots, r)$$

$$(x, y) \in F(P)$$

with $f^+(y)^t y = u^t y$ and $f^+(y)^t y^{(i)} = \sum_{j=1}^n ((u_j y_j + l_j(1 - y_j)) y_j^{(i)}) \quad \forall i$.

Remark 2 *By construction of $Q^{+(r)}$ we have:*

- (i) $F(Q^{+(r)}) \neq \emptyset$.
- (ii) *There exists an optimal solution.*
- (iii) $v(Q^{+(r)}) \geq 0$.
- (iv) $v(Q^{+(i)}) \geq v(Q^{+(i+1)}) \quad i = 1, \dots, r - 1$

In order to prove the core of our work we need a well known auxiliary lemma:

Lemma 2 *Let $f \in \Omega$ and let $y, \hat{y} \in \{0, 1\}^n$ then*

$$f^t \hat{y} - f^t y \leq f^+(\hat{y})^t \hat{y} - f^+(\hat{y})^t y$$

Proof:

$$f^t \hat{y} - f^t y = \sum_{J_{0,1}} f_j - \sum_{J_{1,0}} f_j \leq \sum_{J_{0,1}} u_j - \sum_{J_{1,0}} l_j = f^+(\hat{y})^t \hat{y} - f^+(\hat{y})^t y \quad \bullet$$

The following lemma is the core of our work:

Lemma 3 *$v(Q^{+(r)}) \leq \epsilon$ if and only if $(x^{(1)}, y^{(1)}), \dots, (x^{(r)}, y^{(r)})$ is an ϵ -optimal multiparametric solution.*

Proof:

If $(x^{(1)}, y^{(1)}), \dots, (x^{(r)}, y^{(r)})$ is an ϵ -optimal multiparametric solution then

$$c^t x + f^t y - \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, r\} \leq \epsilon$$

for all $f \in \Omega$ and for all $(x, y) \in F(P)$. Then

$$c^t x + f^+(y)^t y - \max\{c^t x^{(i)} + f^+(y)^t y^{(i)} : i = 1, \dots, r\} \leq \epsilon$$

for all $(x, y) \in F(P)$. Therefore $v(Q^{+(r)}) \leq \epsilon$.

Let us suppose that $v(Q^{+(r)}) \leq \epsilon$. Let $f \in \Omega$ and let (x, y) be an optimal solution for $P(f)$. By using lemma 2 and some algebraic manipulations we have that:

$$\begin{aligned} c^t x + f^t y - \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, r\} &= \\ \min\{c^t x + f^t y - c^t x^{(i)} - f^t y^{(i)} : i = 1, \dots, r\} &\leq \\ \min\{c^t x + f^+(y)^t y - c^t x^{(i)} - f^+(y)^t y^{(i)} : i = 1, \dots, r\} &= \\ c^t x + f^+(y)^t y - \max\{c^t x^{(i)} + f^+(y)^t y^{(i)} : i = 1, \dots, r\} &\leq v(Q^{+(r)}) \leq \epsilon \end{aligned}$$

Therefore,

$$v(P(f)) \geq \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, r\} \geq v(Q^{+(r)}) - \epsilon \quad \bullet$$

For the sake of completeness some relations between $Q^{(r)}$ and $Q^{+(r)}$ are presented in the following lemma.

Lemma 4

(i) $v(Q^{(r)}) = v(Q^{+(r)})$.

(ii) If $((x, y), f)$ is an optimal solution for $Q^{(r)}$ then (x, y) is an optimal solution for $Q^{+(r)}$ and for $P(f^+(y))$.

(iii) If (x, y) is an optimal solution for $Q^{+(r)}$ then $((x, y), f^+(y))$ is an optimal solution for $Q^{(r)}$ and (x, y) is an optimal solution for $P(f^+(y))$.

(iv) Let us suppose that $Q^{(r)}$ and $Q^{+(r)}$ are written as 0-1-MIP problems and let $\bar{Q}^{(r)}$ and $\bar{Q}^{+(r)}$ be the linear relaxations then $v(\bar{Q}^{+(r)}) = v(\bar{Q}^{(r)})$.

Proof:

(i) Since $\Omega^+ \subseteq \Omega$ we have that $v(Q^{(r)}) \geq v(Q^{+(r)})$. Let $((x, y), f)$ be an optimal solution for $Q^{(r)}$. By using the definition of the problems, the optimality of $((x, y), f)$, lemma 2 and some algebraic manipulations we have:

$$\begin{aligned} v(Q^{(r)}) &= c^t x + f^t y - \max\{c^t x^{(i)} + f^t y^{(i)} : i = 1, \dots, r\} = \\ &\quad \min\{c^t x + f^t y - c^t x^{(i)} - f^t y^{(i)} : i = 1, \dots, r\} \leq \\ &\quad \min\{c^t x + f^+(y)^t y - c^t x^{(i)} - f^+(y)^t y^{(i)} : i = 1, \dots, r\} = \\ &\quad c^t x + f^+(y)^t y - \max\{c^t x^{(i)} + f^+(y)^t y^{(i)} : i = 1, \dots, r\} \leq v(Q^{+(r)}) \end{aligned}$$

(ii) Let $((x, y), f)$ be an optimal solution for $Q^{(r)}$. From the proof of (i) we have that (x, y) is an optimal solution for $Q^{+(r)}$. From remark 1 (x, y) is an optimal solution for $P(f)$ and from lemma 1 (x, y) is an optimal solution for $P(f^+(y))$.

(iii) Let (x, y) be an optimal solution for $Q^{+(r)}$. From the proof of (i) and from remark 1 we have that $((x, y), f^+(y))$ is an optimal solution for $Q^{(r)}$ and (x, y) is an optimal solution for $P(f^+(y))$.

(iv) Let $((\bar{x}, \bar{y}), \bar{z})$ be an optimal solution for $\bar{Q}^{+(r)}$ then $((\bar{x}, \bar{y}), \bar{z}, \bar{w}, f^+(\bar{y})) \in F(\bar{Q}^{(r)})$ where $\bar{w}_j = u_j \bar{y}_j \forall j$. Therefore:

$$v(\bar{Q}^{+(r)}) = c^t \bar{x} + u^t \bar{y} - \bar{z} = c^t \bar{x} + \sum_{j=1}^n \bar{w}_j - \bar{z} \leq v(\bar{Q}^{(r)})$$

Let $((\bar{x}, \bar{y}), \bar{z}, \bar{w}, \bar{f})$ be an optimal solution for $\bar{Q}^{(r)}$ then $\bar{w}_j \leq u_j \bar{y}_j$ and $\bar{w}_j - \bar{f}_j \leq -l_j(1 - \bar{y}_j) = u_j \bar{y}_j - f^+(\bar{y})_j$. Therefore

$$\bar{w}_j - \bar{f}_j y_j^{(i)} \leq u_j \bar{y}_j - f^+(\bar{y})_j y_j^{(i)} \text{ and } \sum_{j=1}^n \bar{w}_j - \bar{f}^t y^{(i)} \leq u^t \bar{y} - f^+(\bar{y})^t y^{(i)}.$$

It follows that:

$$v(\bar{Q}^{(r)}) = c^t \bar{x} + \sum_{j=1}^n \bar{w}_j - \bar{z} = c^t \bar{x} + \sum_{j=1}^n \bar{w}_j - \max\{c^t x^{(i)} + \bar{f}^t y^{(i)} : i = 1, \dots, r\} =$$

$$\min\{c^t \bar{x} + \sum_{j=1}^n \bar{w}_j - c^t x^{(i)} - \bar{f}^t y^{(i)} : i = 1, \dots, r\} \leq$$

$$\min\{c^t \bar{x} + u^t \bar{y} - c^t x^{(i)} - f^+(\bar{y})^t y^{(i)} : i = 1, \dots, r\} =$$

$$c^t \bar{x} + u^t \bar{y} - \max\{c^t x^{(i)} + f^+(\bar{y})^t y^{(i)} : i = 1, \dots, r\} \leq v(\bar{Q}^{+(r)}) \bullet$$

Note that lemma 3 may be seen as a consequence of lemma 4.

Since $\{0, 1\}^n$ is a finite set, remark 2 and lemma 3 prove that the next algorithm provide us with an ϵ -optimal multiparametric solution in a finite number of steps:

The *new* multiparametric algorithm

Find $(x^{(1)}, y^{(1)}) \in F(P)$, let $r = 1$ and let $\delta = 2\epsilon$.

while $\delta > \epsilon$

Solve $Q^{+(r)}$ to obtain (x, y) .

Let $\delta = v(Q^{+(r)})$.

If $\delta > \epsilon$ let $(x^{(r+1)}, y^{(r+1)}) = (x, y)$ and let $r = r + 1$.

endwhile

Since $v(\bar{Q}^{+(r)}) = v(\bar{Q}^{(r)})$ and because of $Q^{+(r)}$ does not have the n additional variables and the $4n$ additional constraints that we use to rewrite $Q^{(r)}$ as a MIP-0-1 problem then we can expect that the *new* multiparametric algorithm has a very much better performance than the *old* algorithm defined by using $Q^{(r)}$. Computational experience confirms this presumption: now we can solve

problems with higher dimensions to those used before (see [6]).

$Q^{+(r)}$ may be a hard problem, therefore, alternative algorithms based on relaxations may be useful. Next we present two of them that we use in our computational experience. Others alternatives will be sketched in section 4.

2.1 The *relax and fix* multiparametric algorithm

The *relax and fix* multiparametric algorithm is based on an heuristic for MIP problems with two sets of 0-1-variables ([11]). Let Q be a 0-1-MIP problem in (x, y) . First we solve the relaxation in which the integrality of the x -variables is dropped. Let (\hat{x}, \hat{y}) be the corresponding solution. Next we fix the y -variables at their values in \hat{y} and the restriction problem is solved. We use a similar idea to obtain an ϵ -optimal multiparametric solution.

Let us suppose that $x \in X = \{0, 1\}^p$. Let $\hat{Q}^{+(r)}$ be a relaxation of $Q^{+(r)}$ in which the integrality of the x variables is dropped. Let $P(u)/\hat{y}$ be the restriction of $P(u)$ by adding the constraints $y = \hat{y}$.

We define the *relax and fix* multiparametric algorithm as follows:

Find $(x^{(1)}, y^{(1)}) \in F(P)$, let $r = 1$ and let $\delta = 2\epsilon$.

while $\delta > \epsilon$

Solve $\hat{Q}^{+(r)}$ and let $\delta = v(\hat{Q}^{+(r)})$.

If $\delta > \epsilon$

Let (\hat{x}, \hat{y}) be an optimal solution for $\hat{Q}^{+(r)}$.

Let $F(\hat{Q}^{+(r)}) = F(\hat{Q}^{+(r)}) - \{\hat{y}\}$.

Solve $P(u)/\hat{y}$.

If $F(P(u)/\hat{y}) \neq \emptyset$ let (x, \hat{y}) be an optimal solution.

Let $(x^{(r+1)}, y^{(r+1)}) = (x, \hat{y})$ and let $r = r + 1$

endif

endif

endwhile

Some remarks are necessary: (i) $P(u)/\hat{y}$ may be an infeasible problem, (ii) if (\hat{x}, \hat{y}) is an optimal solution for $\hat{Q}^{+(r)}$ then we may have $\hat{y} \in \{y^{(1)}, \dots, y^{(r)}\}$ with $\delta > \epsilon$. Therefore we delete \hat{y} from $F(\hat{Q}^{+(r)})$. Now $\hat{Q}^{+(r)}$ may be an infeasible problem ($v(\hat{Q}^{+(r)}) = -\infty$) but in this case an ϵ -optimal multiparametric solution has been found and the algorithm is well defined. With the new constraints ($y \notin \{y^{(1)}, \dots, y^{(r)}\}$) added to $\hat{Q}^{+(r)}$ the algorithm generates an ϵ -optimal multiparametric solution in a finite number of steps.

2.2 The *Branch and Bound* multiparametric algorithm

Let us suppose that $X = \{x : x \in \mathbb{R}^p, x \geq 0\}$. A problem in the search tree will be defined by $(K0, K1, K2)$ where $K0 = \{j : y_j = 0\}$, $K1 = \{j : y_j = 1\}$ and $K2 = \{1, \dots, n\} - (K0 \cup K1)$. Let $Q_{(K0, K1, K2)}^{+(r)}$ be the restriction of $Q^{+(r)}$ defined by $(K0, K1, K2)$ and let $\bar{Q}_{(K0, K1, K2)}^{+(r)}$ be its linear relaxation.

We have the following rules in our *multiparametric Branch and Bound* algorithm :

- (1) If $F(\bar{Q}_{(K0, K1, K2)}^{+(r)}) = \emptyset$ then $(K0, K1, K2)$ will be pruned.
- (2) If $v(\bar{Q}_{(K0, K1, K2)}^{+(r)}) \leq \epsilon$ then $(K0, K1, K2)$ will be pruned.
- (3) Let (\bar{x}, \bar{y}) be an optimal solution for $\bar{Q}_{(K0, K1, K2)}^{+(r)}$ with $v(\bar{Q}_{(K0, K1, K2)}^{+(r)}) > \epsilon$ and $\bar{y} \in \{0, 1\}^n$ then: **we do not prune** $(K0, K1, K2)$ and $(x^{(r+1)}, y^{(r+1)}) = (\bar{x}, \bar{y})$. Note that with this rule the problem to be solved changes from $Q^{+(r)}$ to $Q^{+(r+1)}$ and **we keep the same search tree**.

Some remarks are enough to justify our rules:

Remark 3 Since $F(Q_{(K0, K1, K2)}^{+(r+j)}) \subseteq F(Q_{(K0, K1, K2)}^{+(r)})$ for all $j \geq 1$ then we have:

(i) If $Q_{(K0, K1, K2)}^{+(r)}$ is pruned by infeasibility then $Q_{(K0, K1, K2)}^{+(r+j)}$ must be pruned by infeasibility for all $j \geq 1$ justifying rule 1.

(ii) If $v(\bar{Q}_{(K0, K1, K2)}^{+(r)}) \leq \epsilon$ then $v(\bar{Q}_{(K0, K1, K2)}^{+(r+j)}) \leq \epsilon$ for all $j \geq 1$ justifying rule 2.

(iii) Let (\bar{x}, \bar{y}) be an optimal solution for $\bar{Q}_{(K0, K1, K2)}^{+(r)}$ with

$v(\bar{Q}_{(K0, K1, K2)}^{+(r)}) > \epsilon$ and $\bar{y} \in \{0, 1\}^n$ then:

$v(Q_{(K0,K1,K2)}^{+(r+j)})$ may be greater than ϵ and therefore $(K0, K1, K2)$ may have descendants with feasible solutions that we need to obtain an ϵ -optimal multi-parametric solution justifying rule 3.

The **Branch and Bound** multiparametric algorithm is defined as follows:

Find $(x^{(1)}, y^{(1)}) \in F(P)$, let $r = 1$ and let $list = \{(\emptyset, \emptyset, \{1, \dots, n\})\}$.

while $list \neq \emptyset$.

 Choose the next candidate problem defined by $(K0, K1, K2)$.

 Solve $\bar{Q}_{(K0,K1,K2)}^{+(r)}$.

If $F(\bar{Q}_{(K0,K1,K2)}^{+(r)}) \neq \emptyset$ and $v(\bar{Q}_{(K0,K1,K2)}^{+(r)}) > \epsilon$.

 Let (\bar{x}, \bar{y}) be an optimal solution for $\bar{Q}_{(K0,K1,K2)}^{+(r)}$.

If $\bar{y} \in \{0, 1\}^n$ then:

 Let $(x^{(r+1)}, y^{(r+1)}) = (\bar{x}, \bar{y})$ and let $r = r + 1$.

else

 Choose $j : 0 < \bar{y}_j < 1$ and let

$list = list \cup \{(K0+j, K1, K2-j), (K0, K1+j, K2-j)\}$

endif

else

 Let $list = list - \{(K0, K1, K2)\}$.

endif

endwhile.

3 Computational results

All experiments were carried out on a system with two 2.2 GHz Intel processors and 4.00 GB RAM and using CPLEX 12.2 as the optimization engine. We select some problems from the literature in order to evaluate our algorithms. We

do not claim that our algorithms are the best approach for selected problems under uncertainty. Our approach is general and the experiments are designed in order to try to show that it works reasonably well for several problems. The problems considered were the FCHKP ([8]), the FCHMKP([9]) and the SPLP ([10]).

The *new* multiparametric algorithm, the *relax and fix* multiparametric algorithm and the *Branch and Bound* multiparametric algorithm will be denoted **NEW**, **REFIX** and **BB** respectively.

We present computational experience with the multiparametric FCHKP solved with **NEW**, the multiparametric FCHMKP solved with **REFIX** and the multiparametric SPLP solved with **NEW** and **BB**. **NEW** had a very poor performance to solve the multiparametric FCHMKP and the corresponding results are not included. The results by using **NEW** and **REFIX** to solve the multiparametric FCHKP were similar and then we present only the corresponding to **NEW**.

If we use either a Branch and Cut or a Branch and Bound algorithm to solve any problem then a *relative tolerance* is used to prune the tree search (the default value is 10^{-4} by using CPLEX). Let α be the *relative tolerance*. Note that our theoretical results are valid with $\alpha = 0$. If we use $\alpha = 0$ the computational effort to reach an ϵ -optimal multiparametric solution could be huge and impractical for many problems. In order to ensure that we generate an ϵ -optimal multiparametric solution by using $\alpha > 0$ we use $\epsilon/(1 + \alpha)$ instead of ϵ to stop the algorithms. Also, when we use **REFIX**, ϵ is used like an *absolute tolerance* to solve $P(u)/\hat{y}$.

Let λ be the *relative error* to be used. Let $z(l)$ be the value obtained by solving $P(l)$ with the relative tolerance α . We have $z(l) > 0$ for all the experiments and then we use $\epsilon = \lambda z(l)$. Let $tmax$ and $rmax$ be bounds for the CPU-time and number of generated solutions respectively. We use the following rules to stop the algorithms even if $v(Q^{+(r)}) > \epsilon$: either CPU-time $\geq tmax$ after solving $Q^{+(r)}$ (or a relaxation of $Q^{+(r)}$) or $r \geq rmax$. When the algorithm is stopped with $v(Q^{+(r)}) > \epsilon$ then we compute the corresponding value of the *relative error* ($\lambda = (1 + \alpha)(v^{+(r)})/z(l)$) with $v^{+(r)}$ the value obtained by solving $Q^{+(r)}$ (or a relaxation of $Q^{+(r)}$) with the relative tolerance α . Let $(x^{(1)}, y^{(1)})$ be the suboptimal solution for $P(l)$ by using α .

Let $I = \{1, \dots, n\}$ and $J = \{1 \dots, m\}$.

The formulation used for the FCHKP in (x, y) is ([8]):

$$\begin{aligned}
max \quad & \sum_{i=1}^n \sum_{j=1}^m c_{ij} x_{ij} - \sum_{i=1}^n f_i^0 y_i \quad s.t. \\
& \sum_{i=1}^n \sum_{j=1}^m w_{ij} x_{ij} \leq W \\
& x_{ij} - y_i \leq 0, \quad x_{ij} \in \{0, 1\} \quad \forall i \in I \quad \forall j \in J \quad y \in \{0, 1\}^n
\end{aligned}$$

The data were generated as follows. We use a procedure presented in [8] with some minor changes: $c_{ij} \sim U(5, 100)$, $w_{ij} = \max(1, c_{ij} + \gamma_{ij})$, $\gamma_{ij} \sim U(-10, 10)$, $W = \delta \sum_{i=1}^n \sum_{j=1}^m w_{ij}$, $f_i^0 \sim U(fmin, fmax)$ with $fmin = \delta \sum_{i=1}^n \sum_{j=1}^m c_{ij} / n$ and $fmax = sfmin$. Next we randomly perturb f_j^0 as follows: $u_j = -(1 - \beta_j)f_j^0$ and $l_j = -(1 + \beta_j)f_j^0$ with $\beta_j \sim U(0, \beta)$. Finally all the data were rounded.

The first set of problems (set I) was generated with: $n \in \{100, 200\}$, $m \in \{20, 40\}$, $\delta \in \{0.05, 0.15, 0.25, 0.50\}$ and $s \in \{2, 4, 6, 10\}$. The level of perturbation was 5 percent ($\beta = 0.05$). We generate one problem for each combination to obtain 64 problems. We use **NEW** with $\alpha = 0.001$, $\lambda = 0.001$, $tmax = 600$ seconds and $rmax = 50$.

The relative errors average was 0.000856 with three values greater than 0.002 and two values greater than 0.004 (see figure 1). The generated solutions average was 7.23 with eight values greater than 20 and two values equal to 50 (see figure 2). The CPU time average was 76.07 seconds with 11 values greater than 100 seconds and 5 values greater than 600 seconds (see figure 3). Problem 62 had the worse relative error with 0.0056. Next we use **NEW** for problem 62 with $rmax = 300$ and $tmax = \infty$. The relative error reached was 0.0031 in 4087 seconds (see figure 4).

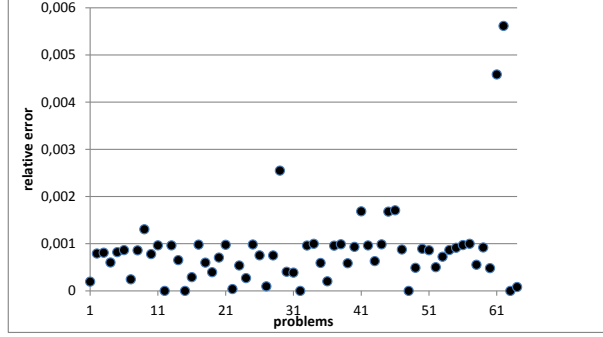


Figure 1: Fixed Charge KP-set I, relative errors by using **NEW**

The second set of problems (set II) was generated as before but $s \in \{2, 3, 4, 5\}$ because of problems with $s \geq 6$ were easy to solve in the set I. The level of perturbation was 7.5 percent ($\beta = 0.075$). Again we have 64 problems and **NEW** was used with $\alpha = 0.001$, $\lambda = 0.001$, $tmax = 2400$ seconds and $rmax = 50$.

The relative errors average was 0.0019 with seventeen values greater than 0.002, nine values greater than 0.004 and four values greater than 0.006 (see figure 5). The generated solutions average was 18.56 with fourteen values greater than 40 and twelve values equal to 50 (see figure 6). The CPU time average was 231.49 seconds with 11 values greater than 600 seconds and one value greater than 2000 sweconds (see figure 7). Problem 61 had the worse relative error with 0.0091. Next we use **NEW** for problem 61 with $rmax = 35$ and $tmax = \infty$. The relative error reached was 0.0082 in 19334 seconds (see figure 8). The relative error sequence was not monotone because of we use $\alpha = 0.075$ in order to save time.

The formulation used for the FCHMKP in (x, y) is ([9]):

$$\begin{aligned}
 \max \quad & \sum_{i=1}^n \sum_{j=1}^m c_j x_{ij} - \sum_{i=1}^n f_i^0 y_i \quad s.t. \\
 & \sum_{j=1}^m w_j x_{ij} \leq W_i y_i \quad \forall i \in I \\
 & \sum_{i=1}^n x_{ij} \leq 1 \quad \forall j \in J
 \end{aligned}$$

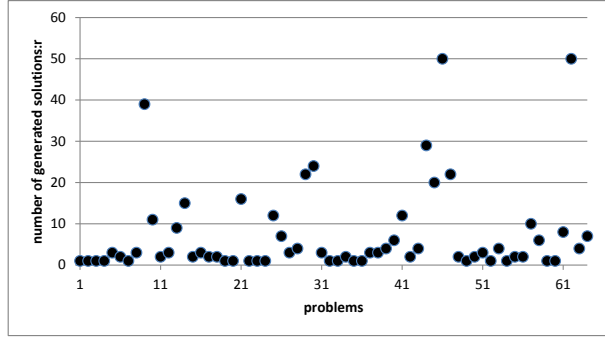


Figure 2: Fixed Charge KP-set I, number of generated solutions by using NEW

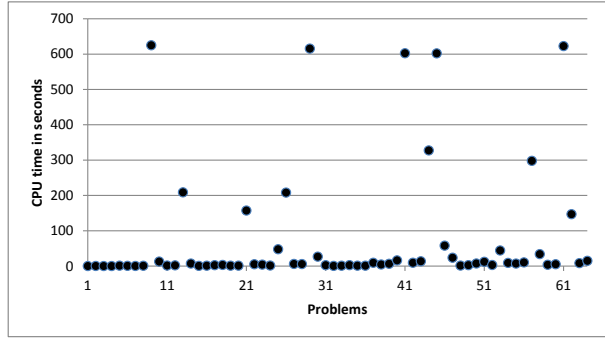


Figure 3: Fixed Charge KP-set I, CPU time in seconds by using NEW

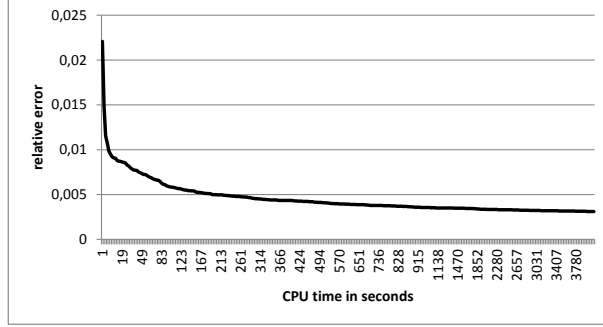


Figure 4: Fixed Charge KP-set I, relative error vs CPU time by using NEW, problem 62

$$x_{ij} \in \{0, 1\} \quad \forall i \in I \quad \forall j \in J, \quad y \in \{0, 1\}^n$$

The data were generated as follows ([9]): $w_j \sim U(1, 1000)$, $c_j \sim U(1, 1000)$ and $f_j^0 = \rho_j W_j$ with $W_j = 500n\delta(\gamma_j / \sum_{i=1}^n \gamma_i)$, $\gamma_i \sim U(0, 1)$ and $\rho_j \sim U(0.5, 1.5)$. Next we randomly perturb f_j^0 as follows: $u_j = -(1 - \beta_j)f_j^0$ and $l_j = -(1 + \beta_j)f_j^0$ with $\beta_j \sim U(0, \beta)$. Finally all the data were rounded.

The third set of problems (set III) was generated with $n \in \{25, 35\}$, $m \in \{500, 750, 1000\}$ and $\delta \in \{0.25, 0.35, 0.50\}$. The level of perturbation was 5 percent ($\beta = 0.05$). We generate four problems for each combination to obtain 72 problems. We use **REFIX** with $\alpha = 0.005$, $\lambda = 0.005$, $tmax = 1200$ seconds and $rmax = 50$.

The relative errors average was 0.0044 with only one greater than 0.005 (see figure 9). The generated solutions average was 2.62 with only two values greater than 10, one value greater than 20 and no one equal to 50 (see figure 10). The CPU time average was 98 seconds with three values greater than 400 seconds, two values greater than 800 seconds and only one value greater than 1600 seconds (see figure 11).

Next we use **REFIX** for the first 18 problems of set I with $\alpha = 0.005$, $\lambda = 0.0025$, $tmax = 1200$ seconds and $rmax = 50$. The relative errors average was 0.0020 with only one value greater than 0.0025. The worse value was 0.0036. The generated solutions average was 3.88 with only one greater than 10 and no one equal to 50. The CPU time average was 450.72 seconds with three values greater than 1000 seconds and only one value greater than 1200 seconds.

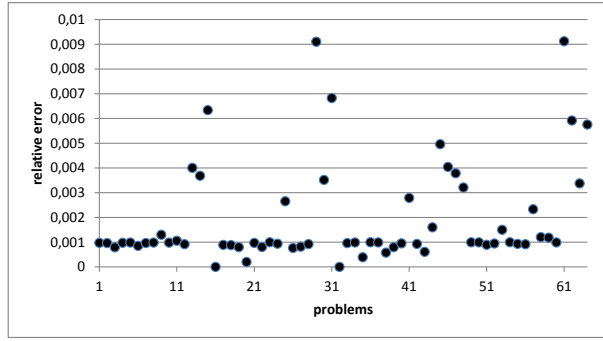


Figure 5: Fixed Charge KP-set II, relative errors by using NEW

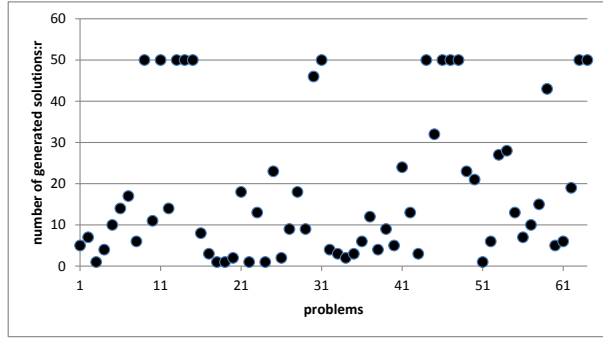


Figure 6: Fixed Charge KP-set II, number of generated solutions by using NEW

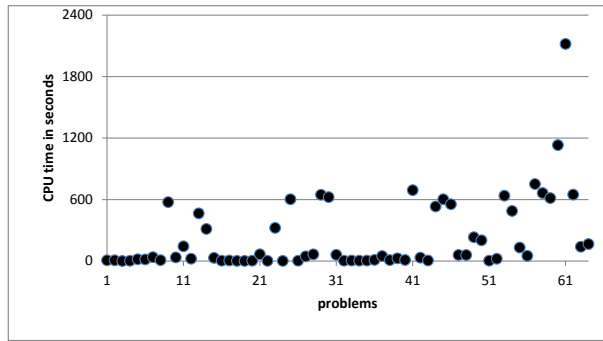


Figure 7: Fixed Charge KP-set II, CPU time in seconds by using NEW

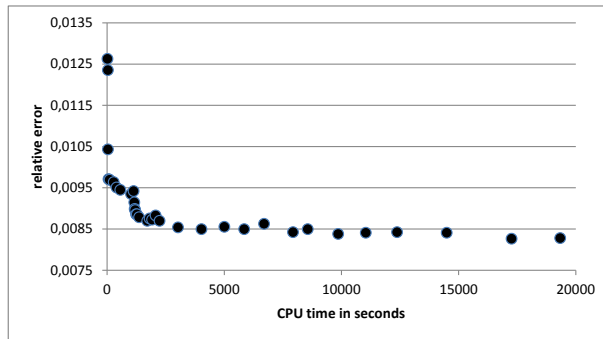


Figure 8: Fixed Charge KP-set II, relative error vs CPU time by using NEW, problem 61

The fourth set of problems (set IV) was generated as the set III but with the level or perturbation equal to 7.5 percent ($\beta = 0.075$). We use **REFIX** with $\alpha = 0.005$, $\lambda = 0.005$, $tmax = 2400$ seconds and $rmax = 50$.

The relative errors average was 0.0045 with no one one value grater than 0.005 (see figure 12). The generated solutions average was 5.11 with eight values greater than 10, three values greater than 20 and no one equal to 50 (see figure 13). The CPU time average was 182.16 seconds with five values greater than 500 seconds, two values greater than 1000 seconds, one value greater than 2300 seconds and no one greater than 2400 seconds (see figure 14).

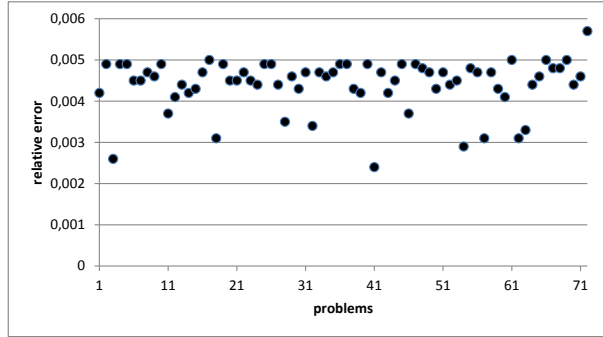


Figure 9: Fixed charge KP multiple-set III, relative errors bu using REFIX

The formulation used for the SPLP in (x, y) is:

$$\begin{aligned}
 \max \quad & \sum_{i=1}^n \sum_{j=1}^m c_{ij} x_{ij} - \sum_{i=1}^n f_i^0 y_i \quad s.t. \\
 & \sum_{i=1}^n x_{ij} = 1 \quad \forall j \in J \\
 & x_{ij} - y_i \leq 0, \quad x_{ij} \in \{0, 1\} \quad \forall i \in I \quad \forall j \in J, \quad y \in \{0, 1\}^n
 \end{aligned}$$

The data were generated as follows: n points were generated at random in the square unit, let d_{ij} be the distance from point i to point j , $D_j \sim U(Dl, Du)$, $c_{ij} = 3D_j/(1 + d_{ij})$, $f_i^0 = Fmin + s_i(Fmax - Fmin)$ with $s_i = (\sum_{j=1}^n c_{ij} - fmin)/(fmax - fmin)$, $fmin = \min\{\sum_{j=1}^n c_{ij} : i = 1, \dots, n\}$

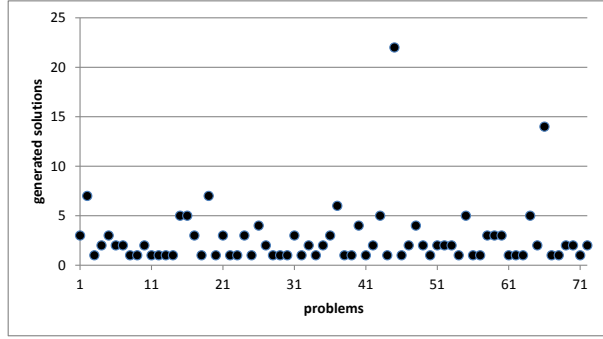


Figure 10: Fixed charge KP multiple-set III, number of generate solutions by using REFIX

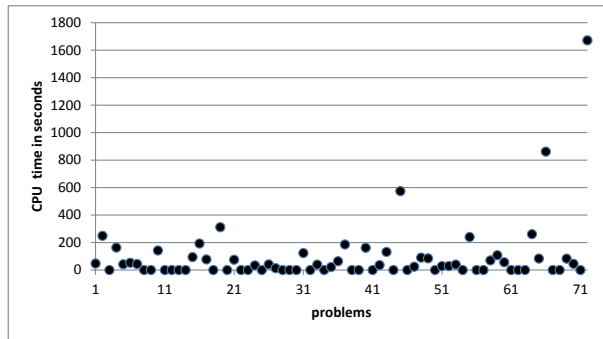


Figure 11: Fixed charge KP multiple-set III, CPU time in seconds by using REFIX

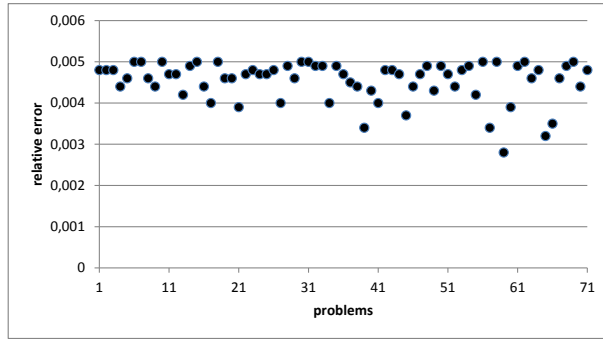


Figure 12: Fixed charge KP multiple-set IV, relative errors by using REFIX

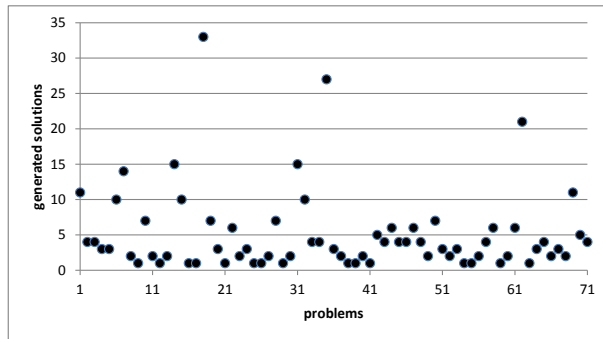


Figure 13: Fixed charge KP multiple-set IV, number of generate solutions by using REFIX

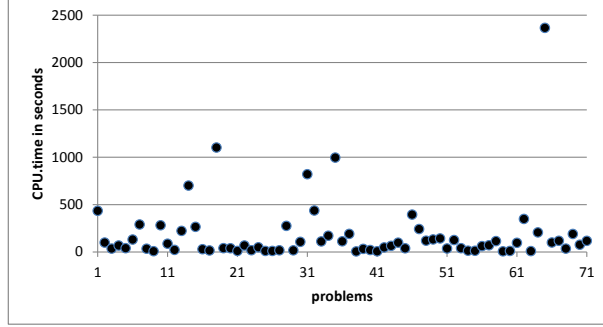


Figure 14: Fixed charge KP multiple-set V, CPU time in seconds by using REFIX

and $fmax = \max\{\sum_{j=1}^n c_{ij} : i = 1, \dots, n\}$. Next we randomly perturb f_i^0 as follows: $u_i = -(1 - \beta)f_i^0$ and $l_i = -(1 + \beta)f_i^0$.

Note that we may delete the integrality of the x -variables in the SPLP and the same is true for the $Q^{+(r)}$ corresponding problem. We use **NEW** and **BB** to solve the multiparametric SPLP by using the *Depth-First Search Strategy* to choose the next problem and with the following rule to choose the branching variable: let (\bar{x}, \bar{y}) be an optimal solution for $\bar{Q}_{(K0, K1, K2)}^{+(r)}$, if $(u_j - l_j) \min\{\bar{y}_j, 1 - \bar{y}_j\} = \max\{(u_k - l_k) \min\{\bar{y}_k, 1 - \bar{y}_k\} : 0 < \bar{y}_k < 1\}$ then we use j to define the descendants of $Q_{(K0, K1, K2)}^{+(r)}$.

The fifth set of problems (set V) was generated with $n \in \{100, 150\}$, $Dl = 1$, $Du = 100$, $Fmin \in \{100, 150, 200\}$, $Fmax \in \{400, 600, 800\}$. The level of perturbation was 5 percent ($\beta = 0.05$). We generate one problem for each combination to obtain 18 problems. We use **NEW** with $\alpha = 0.0001$, $\lambda = 0.005$, $tmax = 1800$ seconds and $rmax = 100$. Also, we use **BB** with $\lambda = 0.005$.

The following analysis corresponds to problems 1 to 16. The CPU-time averages were 438.24 and 894.18 seconds by using **BB** and **NEW** respectively and the CPU-time corresponding to **BB** was less than the CPU-time corresponding to **NEW** except for one problem (see figure 15). **BB** ensures a relative error less or equal to 0.005, the relative error average was 0.0052 by using **NEW** with four values greater than 0.005. The worse value was 0.0066 (see figure 16). The generated solutions averages were 46.12 and 31.37 by using **BB** and **NEW** respectively and the number of generated solutions corresponding to **BB** was

greater than the number of generated solutions corresponding to **NEW** except for two problems (see figure 17). The relative errors reached with **NEW** were 0.0059 and 0.0065 for problems 17 and 18 respectively. The corresponding CPU-times were 1857 and 1819 seconds. We use **BB** for problems 17 and 18 with the same relative errors reached by using **NEW**. **BB** ensures the same relative errors for problems 17 and 18 with CPU-time 935 and 755 seconds respectively. Next we use **BB** with $\lambda = 0.005$ for problems 17 and 18 and the CPU-times were 2716 and 2294 seconds respectively.

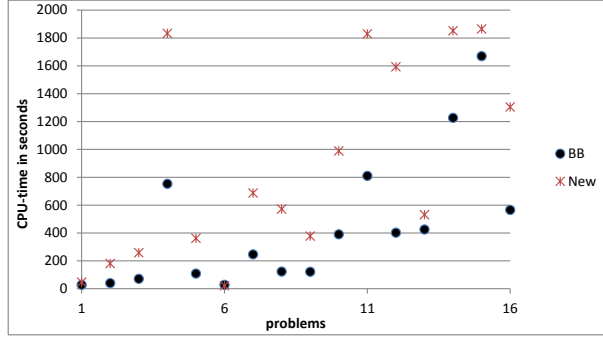


Figure 15: SPLP-set V, CPU time in seconds by using NEW and BB

The sixth set of problems (set VI) was generated with $n \in \{50, 75, 100, 125, 120, 200\}$, $Dl = 100$, $Du = 10000$, $Fmin \in \{25000, 50000, 75000\}$ and $Fmax = 100000$. The level of perturbation was 5 percent ($\beta = 0.05$). We generate one problem for each combination to obtain 18 problems. We use **NEW** with $\alpha = 0.0001$, $\lambda = 0.0075$, $tmax = 3600$ seconds and $rmax = 100$. Also, we use **BB** with $\lambda = 0.0075$. Both algorithms ensure the relative error defined. The CPU-time and the number of generate solutions may be seen in figures 18 and 19 respectively.

4 Summary and further extensions

We have designed algorithms to approximate the solution for the multiparametric 0-1-mixed integer linear programming problem relative to the objective function. We consider the uncertainty for the parameters that define the cost vector corresponding to a subset of 0-1-variables by assuming that each parameter belongs to a known interval. By using the concept of the *the most favo-*

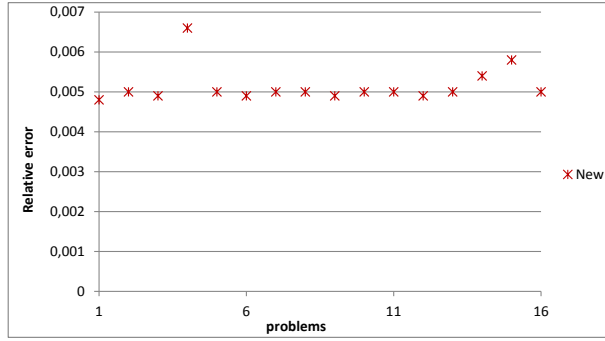


Figure 16: SPLP-set V, relative errors by using NEW

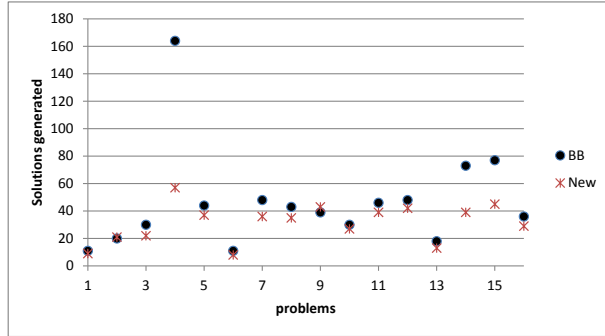


Figure 17: SPLP-set V, solutions generated by using NEW and BB

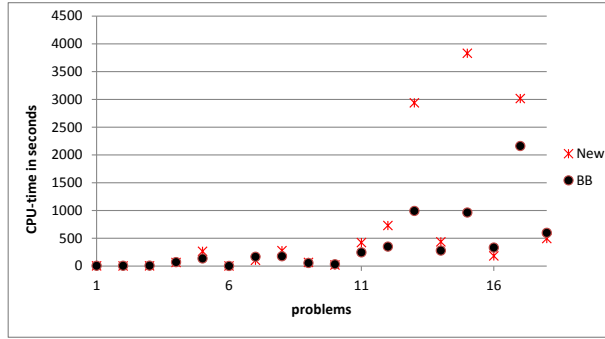


Figure 18: SPLP-set VI, CPU time in seconds by using NEW and BB

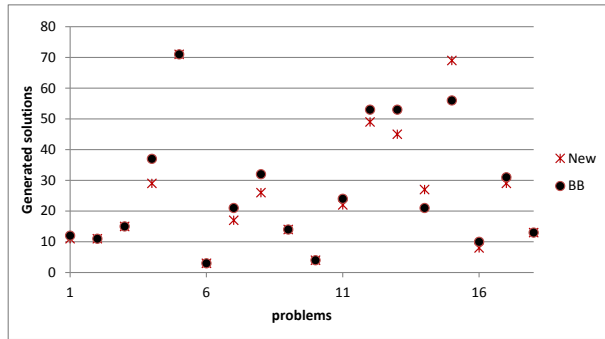


Figure 19: SPLP-set VI, solutions generated by using NEW and BB

table scenario for y we redesigned our *old* algorithm. With the *new* algorithm we can solve problems with higher dimensions. However the problem to be solved in each step of the *new* algorithm may be still a hard problem and then algorithms based on relaxations may be useful. We have presented two alternatives that use relaxations based on the *relax and fix* and *Branch and Bound* approaches widely used for nonparametric problems.

We have presented computational experience with three algorithms: **NEW**, **REFIX** and **BB** to approximate the solution of the multiparametric version of three well known problems: FCHKP, FCHMKP and SPLP. We hope that our computational experience has shown that our approach is promising and we want to highlight that approaches designed for nonparametric problems were used without major changes to approximate the solutions for multiparametric problems.

The use of well known approaches is not restricted to *relax and fix* and *Branch and Bound*. Approaches based on *Benders Decomposition* (with the generated cuts still valid when we go from $Q^{+(r)}$ to $Q^{+(r+1)}$), *local search* (by solving $Q^{+(r)}$ in a neighbourhood of a reference solution, returning to the global search by solving $Q^{+(r)}$ when the local search fails in order to generate the next reference solution and so on) and lagrangean relaxations (by dualizing the constraints relative to the generated solutions to define a nonparametric lagrangean problem with the same structure than the original problem P) may be easily designed.

Also, we can design specialized multiparametric algorithms in order to solve hard problems with higher dimensions by using algorithms to solve $Q^{+(r)}$ (or a relaxation of $Q^{+(r)}$) that use the structure of P. Our approach turns out to be, from this point of view, a general methodology.

5 References

- [1] Fatma Kılınç-Karzan, Alejandro Toriello, Shabbir Ahmed, George Nemhauser, Martin Savelsbergh: ‘Approximating the stability region for binary mixed.integer program’, Operations Research Letters 37 (2009) 250-254.
- [2] Alexander Mitsos: ‘Parametric mixed integer 0-1 linear programming: the general case of a single parameter’, European Journal of Operational Research 94 (3) (2009) 663-686.
- [3] Li,Z and M.G.Ierapetritou: ‘A new methodology for the general multiparametric mixed integer linear programming’, Ind. Eng. Chem. Res. 46

(2007) 5141-5151.

[4] : Viveck Dua and Efstratios N. Pistikopoulos: ‘An algorithm for the solution of Multiparametric Mixed Integer Programming Problem’, *Annals of Operations Research* 99 (1-4) (2000) 123-139.

[5] Larry Jenkins: ‘Parametric methods in integer linear programming’, *Annals of Operations Research* 27 (1990) 77-96.

[6] Alejandro Crema: ‘An algorithm for the multiparametric 0-1 integer linear programming’, *European Journal of Operational Research* 125 (2000) 18-24.

[7] IBM ILOG CPLEX Optimization Studio V12.2.

[8] Umit Akinc: ‘Approximate and exact algorithms for the fixed-charge knapsack problem’, *European Journal of Operational Research* 170 (2006) 363-375.

[9] Takeo Yamada and Takahiro Takeoka: ‘An exact algorithm for the fixed-charge multiple knapsack problem’, *European Journal of Operational Research* 192 (2009) 700-705.

[10] Francisco Barahona and Fabian A. Chudak: ‘Near-optimal solutions to large-scale facility location problems’, *Discrete Optimization* 2 (2005) 35-50

[11] Laurence A. Wolsey: ‘Integer Programming’, *Wiley-Interscience Series in Discrete Mathematics and Optimization*, Wiley-Interscience publication (1998).