Comput Optim Appl (2009) 43: 39–65
DOI 10.1007/s10589-007-9125-x

# A cutting plane algorithm for the capacitated facility location problem

**Pasquale Avella · Maurizio Boccia**

Received: 7 July 2006 / Revised: 30 May 2007 / Published online: 6 December 2007


**Abstract** The *Capacitated Facility Location Problem* (*CFLP*) is to locate a set of facilities with capacity constraints, to satisfy at the minimum cost the order-demands of a set of clients. A multi-source version of the problem is considered in which each client can be served by more than one facility.

In this paper we present a reformulation of the *CFLP* based on *Mixed Dicut Inequalities*, a family of minimum knapsack inequalities of a mixed type, containing both binary and continuous (flow) variables. By aggregating flow variables, any Mixed Dicut Inequality turns into a binary minimum knapsack inequality with a single continuous variable. We will refer to the convex hull of the feasible solutions of this minimum knapsack problem as the *Mixed Dicut polytope*.

We observe that the Mixed Dicut polytope is a rich source of valid inequalities for the *CFLP*: basic families of valid *CFLP* inequalities, like Variable Upper Bounds, Cover, Flow Cover and Effective Capacity Inequalities, are valid for the Mixed Dicut polytope. Furthermore we observe that new families of valid inequalities for the *CFLP* can be derived by the lifting procedures studied for the minimum knapsack problem with a single continuous variable.

To deal with large-scale instances, we have developed a Branch-and-Cut-and-Price algorithm, where the separation algorithm consists of the complete enumeration of the facets of the Mixed Dicut polytope for a set of candidate Mixed Dicut Inequalities. We observe that our procedure returns inequalities that dominate most of the known classes of inequalities presented in the literature. We report on computational

P. Avella
RCOST—Research Center on Software Technology, Universitá del Sannio, C.so Garibaldi 107, 82100 Benevento, Italy
e-mail: avella@unisannio.it

M. Boccia (✉)
Dipartimento di Ingegneria, Universitá del Sannio, Piazza Roma 21, Palazzo ex INPS, 82100 Benevento, Italy
e-mail: maurizio.boccia@unisannio.it

experience with instances up to 1000 facilities and 1000 clients to validate the approach.

**Keywords** Capacitated facility location problem · Mixed dicut inequalities · Facet-enumeration

## 1 Introduction

The Capacitated Facility Location Problem (*CFLP*) is to locate a set of facilities with capacity constraints, with the aim of satisfying at the minimum cost the order-demands of a set of clients. A multi-source version of the problem is considered in which each client can be served by more than one facility.

*CFLP* has received a considerable amount of attention in the literature. Approximation results have been presented by Korupolu, Plaxton and Rajaraman [15] and by Chudak and Williamson [12]. Cornuejols, Sridharan and Thizy [13] report on a comparison between greedy, interchange and Lagrangean heuristics. A Lagrangean heuristic for larger CFLP instances (up to 500 facilities and 1000 clients) is described in Beasley [10]. Barahona and Chudak [7] present a heuristic based on Volume Subgradient and on randomized rounding [4]. In [5], a fine-tuned Lagrangean heuristic is used to solve large-scale instances.

Leung and Magnanti [16], Aardal, Pochet and Wolsey [3] and Aardal [2] studied the description of the *CFLP* polytope. For a deep and exhaustive discussion of polyhedral results we refer the reader to [1]. Aardal [2] also presented a Branch-and-Cut algorithm for solving instances up to 75 facilities and 75 clients, a remarkable result at that time. More recently, Klose and Görtz [14] proposed a Branch-and-Price algorithm based on Dantzig-Wolfe decomposition and on stabilized column generation for the solution of instances with up to 200 facilities and 500 clients.

Avella, Sassano and Vasil'ev [6] studied the related *p-median* problem and proposed an approach based on a Branch-and-Cut algorithm with delayed column generation.

In this paper, we use the *Mixed Dicut Inequalities*, introduced by Ortega and Wolsey [19], to reformulate the *CFLP* as a *Fixed Charge Network Flow Problem*. Mixed Dicut Inequalities are minimum knapsack inequalities of a mixed type, containing both binary and continuous (flow) variables. By aggregating the flow variables into a single continuous variable $s$, any Mixed Dicut Inequality turns into a minimum knapsack with a single continuous variable. We refer to the convex hull of the feasible solutions of this minimum knapsack problem as the *Mixed Dicut polytope*.

We observe that the Mixed Dicut polytope is a rich source of valid inequalities for the *CFLP*: basic families of valid inequalities, like Variable Upper Bounds, Cover and Flow Cover and Effective Capacity Inequalities, are valid for the Mixed Dicut polytope. Furthermore new families of valid inequalities can be derived by using the lifting procedures introduced by Marchand and Wolsey [17] for the knapsack problem with a single continuous variable.

A Branch-and-Cut-and-Price algorithm is proposed to handle large scale instances. The LP-relaxation is solved by delayed column generation. The cutting plane generation consists of the complete enumeration of the facets of the Mixed Dicut polytope for a set of candidate base Mixed Dicut Inequalities. The procedure returns valid inequalities dominating most of the known classes presented in the literature.

The remainder of the paper is organized as follows. In Sect. 2 we describe the standard formulation of the *CFLP*. In Sect. 3 we review the literature concerning polyhedral properties of the *CFLP*. In Sect. 4 we present the reformulation of the *CFLP* as a Fixed Charge Network Flow problem and introduce the family of the Mixed Dicut Inequalities. In Sect. 5 we review all the known families of valid inequalities for the *CFLP* and prove that they are valid for the Mixed Dicut Polytope. In Sect. 6 we define a new class of valid inequalities by extending to the Mixed Dicut Polytope the results presented in [17] for the 0-1 Knapsack Problem with a Single Continuous Variable. In Sect. 7 we describe the facet-enumeration and the selection of candidate Mixed Dicut Inequalities. In Sect. 8 we describe the Branch-and-Cut-and-Price algorithm developed to deal with large-scale instances. Finally in Sect. 9 we report on computational results validating the usefulness of the proposed approach.

## 2 Problem formulation

Let $M = \{1, \dots, m\}$ be a set of facilities, $N = \{1, \dots, n\}$ a set of clients and let $G(M \cup N, A)$ be a complete bipartite graph. Let $u_k$ be the capacity of the facility $k$ and let $d_j$ be the demand of client $j \in N$. Let $h_k$ be the fixed cost of opening facility $k \in M$. Let $c_{kj}$ be the cost of sending one unit of demand from the facility $k \in M$ to the client $j \in N$ through the arc $(k, j) \in A$. Let variables $f_{kj}$ be the flow of the demand from facility $k \in M$ to the client $j \in N$.

A subset $K \subseteq M$ of facilities is *feasible* if the client demands can be assigned to the facilities in $K$ while respecting the capacity bounds, i.e. if the following *Transportation Problem*, denoted as *Transp*$(K)$, admits a feasible solution:

$$\begin{aligned} \min \quad & \sum_{k \in K} \sum_{j \in N} c_{kj} f_{kj}, \\ & \sum_{k \in K} f_{kj} = d_j, \quad j \in N, \\ & \sum_{j \in N} f_{kj} \le u_k, \quad k \in K, \\ & f_{kj} \ge 0, \quad k \in M, j \in N. \end{aligned}$$

Let $c(K)$ be the value of the optimal solution of *Transp*$(K)$ and let $h(K) = \sum_{k \in K} h_k$ be the sum of the fixed costs of the open facilities. The Capacitated Facility Location Problem (*CFLP*) is to find a feasible subset $K \subseteq M$ of open facilities, minimizing the objective function $z(K) = h(K) + c(K)$.

Let $y_k$ be the binary variable associated with the facility $k \in M$, i.e. $y_k = 1$ if $k$ is open (i.e., $k \in K$), 0 otherwise. Formulation $F1$ of the *CFLP* is:

$$\min \quad \sum_{k \in M} h_k y_k + \sum_{k \in M} \sum_{j \in N} c_{kj} f_{kj},$$

$$\sum_{k \in M} f_{kj} = d_j, \quad j \in N, \tag{1}$$

$$\sum_{j \in N} f_{kj} \leq u_k y_k, \quad k \in M, \tag{2}$$

$$y_k \in \{0, 1\}, \quad k \in M,$$

$$f_{kj} \geq 0, \quad k \in M, j \in N.$$

Constraints 1 require that the whole demand of each client should be satisfied. Capacity constraints 2 require that no client can be supplied from a closed facility and that the total demand supplied from the facility $k \in K$ does not exceed its capacity $u_k$.

## 3 Basic polyhedral results

Let $X(G, d, u)$ be the set of the integer feasible solutions of formulation $F1$ and let $CAP(G, d, u) = \text{conv}(X(G, d, u))$ be the *Capacitated Facility Location Polytope*. The dimension of the $CAP(G, d, u)$ is $|M| \times |N| + |M| - |N|$ if $G$ is a complete bipartite graph [1]. In this section we review the main classes of valid inequalities introduced in the literature. For any $S \subset N$, we will denote $d(S) = \sum_{j \in S} d_j$.

### 3.1 Variable upper bounds

Let $k \in M$, $j \in N$. The Variable Upper Bound

$$f_{kj} \leq u_k y_k \tag{3}$$

is valid for $CAP(G, d, u)$. Equation (3) can be strengthened to $f_{kj} \leq d_j y_k$, if $d_j < u_k$.

**Proposition 3.1** (Aardal [1]) *Let $k \in M$, $j \in N$. The Variable Upper Bound $f_{kj} \leq d_j y_k$ is facet-defining for $CAP(G, d, u)$ if $\sum_{h \in M} u_h - \max_{h \in M} u_h > d(N)$ and $d_j < u_k$.*

### 3.2 Cover inequalities

Let $C \subset M$ be a subset of facilities and let $\overline{C} = M \setminus C$. The set $C$ is said to be a *cover* if $\sum_{k \in \overline{C}} u_k < d(N)$. Since the capacity of the facilities in $\overline{C}$ is not large enough to satisfy the client demand, at least one of the facilities in $C$ must be open and the *Cover Inequality*

$$\sum_{k \in C} y_k \geq 1 \tag{4}$$

is valid for $CAP(G,d,u)$. The cover $C$ is *minimal* if each subset $S \subset C$ is not a cover.

**Proposition 3.2** (Aardal [1]) *Let $C \subset M$ be a minimal cover and let $u_{\min} = \min_{k\in C} u_k$ be the minimum capacity of a facility in $C$. The Cover Inequality*:

$$\sum_{k\in C} y_k \geq 1$$

*is facet-defining for* $\mathrm{conv}(X(G,d,u) \cap \{y_k = 1, k \in \overline{C}\})$ *if* $\sum_{k\in\overline{C}} u_k + u_{\min} > d(N)$.

Sequential lifting can make inequality (4) facet-defining for $CAP(G,d,u)$ [1].

### 3.3 Flow cover and effective capacity inequalities

Let $J \subset N$ and let $C \subset M$ be a *flow-cover*, i.e. a subset of facilities with the property that $\lambda = \sum_{k\in C} u_k - d(J) > 0$.

**Theorem 3.1** (Aardal [1]) *The* Flow Cover Inequality

$$\sum_{k\in C}\sum_{j\in J} f_{kj} + \sum_{k\in C}(u_k - \lambda)^+(1-y_k) \leq d(J), \tag{5}$$

*where* $(u_k - \lambda)^+ = \max(0, (u_k - \lambda))$, *is valid for $CAP(G,d,u)$.*

Aardal [1] proved that Flow Cover inequalities are facet-defining if $\max_{i\in M}\{u_i\} > \lambda$ and $\sum_{k\in M} u_k > d(J) + u_r, \forall r \in M$. Flow Cover Inequalities generalize the *Residual Capacity Inequalities*, introduced by Leung and Magnanti [16] for the special CFLP where all the facilites have the same capacity $u$.

Aardal [1] also introduced the family of *Effective Capacity Inequalities*, that dominate Flow Cover Inequalities. Suppose the bipartite graph $G(M \cup N, A)$ is not complete and let $J_k \subseteq J$ be the subset of clients reachable by the facility $k \in M$. The *effective capacity* of the facility $k$ is $\overline{u}_k = \min\{u_k, d(J_k)\}$. Let $J \subset N$ and let $C \subset M$ be an *effective flow-cover* for $J$, i.e. a subset of facilities with the property that $\sum_{k\in M} \overline{u}_k - d(J) = \lambda > 0$, $\lambda \leq \max_{k\in C} u_k$.

**Theorem 3.2** (Aardal [1]) *The* Effective Capacity Inequality

$$\sum_{k\in C}\sum_{j\in J_k} f_{kj} + \sum_{k\in C}(\overline{u}_k - \lambda)^+(1-y_k) \leq d(J) \tag{6}$$

*is valid for $CAP(G,d,u)$.*

Aardal provides necessary and sufficient conditions guaranteeing that Effective Capacity Inequalities are facet-defining for $CAP(G,d,u)$.

## 4 Fixed charge network flow reformulation

Let $o$ be a supernode and let $O = \{(o,k) : k \in M\}$ be an additional set of arcs connecting $o$ to $M$. Let $x_{kj}$ be the variable representing the capacity installed on the arc $(k, j)$. With these additions, it is easy to see that the *CFLP* is a Single Commodity Uncapacitated Fixed Charge Network Flow problem on the graph $G_o(\{o\} \cup M \cup N, O \cup A)$, where the amount of flow $\sum_{j \in N} d_j$ must be sent from the source node $o$ to the sink nodes $N$ through the transhipment nodes $M$. A minor difference with the standard version is that capacities $x_{ij}$ are not forced to be integral.

$$\min \quad \sum_{k \in M} h_k y_k + \sum_{k \in M} \sum_{j \in N} c_{kj} x_{kj},$$

$$\sum_{k \in M} f_{ok} = \sum_{j \in N} d_j, \tag{7}$$

$$\sum_{j \in N} f_{kj} - f_{ok} = 0, \quad k \in M, \tag{8}$$

$$\sum_{k \in M} f_{kj} = d_j, \quad j \in N, \tag{9}$$

$$f_{ok} \le u_k y_k, \quad k \in M, \tag{10}$$

$$f_{kj} \le x_{kj}, \quad k \in M, j \in N, \tag{11}$$

$$y_k \in \{0, 1\}, \quad k \in M,$$

$$f_{kj} \ge 0, \quad k \in M, j \in N,$$

$$x_{kj} \ge 0, \quad k \in M, j \in N.$$

Let $K \subset M$ and let $J \subset N$. The set of arcs $(K, J) = \{(o,k) \in O : k \in K\} \cup \{(k, j) \in A : k \in M \setminus K, j \in J\}$ defines a *dicut* on $G_o$ separating $o$ and the node set $J$. By the Max Flow-Min Cut theorem it is easy to show that the *CFLP* can be reformulated using the Mixed Dicut Inequalities introduced in Ortega and Wolsey [19].

**Proposition 4.1** (Ortega and Wolsey [19]) *Let $K \subset M$ and let $J \subset N$. The Mixed Dicut Inequality*

$$\sum_{k \in K} u_k y_k + \sum_{k \in M \setminus K} \sum_{j \in J} x_{kj} \ge d(J) \tag{12}$$

*is valid for $CAP(G, d, u)$.*

Since the capacity $x_{kj}$ is fractional, in any optimal solution we will get $x_{kj} = f_{kj}$, for any value of $f_{kj}$ and it follows that we can project out variables $x_{kj}$ and the *Mixed Dicut Formulation $F2$* of the *CFLP* is:

$$\min \quad \sum_{k\in M} h_k y_k + \sum_{k\in M}\sum_{j\in N} c_{kj} f_{kj},$$

$$\sum_{k\in K} u_k y_k + \sum_{k\in M\setminus K}\sum_{j\in J} f_{kj} \geq \sum_{j\in J} d_j, \quad K\subset M, J\subset N, \tag{13}$$

$$y_k \in \{0,1\}, \quad k\in M,$$

$$0 \leq f_{kj} \leq d_j, \quad k\in M, j\in N.$$

## 5 *CFLP* inequalities and the mixed dicut polytope

By letting $s=\sum_{k\in M\setminus K}\sum_{j\in J} f_{kj}$, any Mixed Dicut Inequality (13) turns into a minimum knapsack inequality with a single continuous variable. Let $K\subseteq M$ and $J\subseteq N$ and let

$$\sum_{k\in K} u_k y_k + s \geq d(J) \tag{14}$$

be a Mixed Dicut Inequality. Let $Y(K,J)=\{(y,s)\in\{0,1\}^n\times\mathbb{R}_+ : \sum_{k\in K} u_k y_k + s\geq d(J)\}$ be the set of incidence vectors of the feasible solutions and let $CUT(K,J) = \mathrm{conv}(Y(K,J))$ be the *Mixed Dicut Polytope* induces by $K$ and $J$. It is easy to prove that $CUT(K,J)$ is full-dimensional. A deeper analysis of $CUT(K,J)$ leads to tighter cutting planes families that strengthen formulations $F1$ and $F2$. In the following we observe that the families of valid inequalities described in Sect. 3 are valid for $CUT(K,J)$ for some $K\subseteq M$ and $J\subseteq N$.

### 5.1 Variable upper bounds

**Proposition 5.1** *Let $k\in M$ and let $j\in N$. The Variable Upper Bound (3) $f_{kj}\leq u_k y_k$ is valid for $CUT(\{k\},\{j\})$.*

*Proof* By adding the equality $\sum_{k\in M} f_{kj}=d_j$ to the Variable Upper Bound $f_{kj}\leq u_k y_k$, we get the Mixed Dicut Inequality $u_k y_k + s\geq d_j$. □

### 5.2 Cover inequalities

Consider the dicut $(\{o\}:M)=\{(o,k):k\in M\}$ formed by the arcs leaving the node $o$ and entering the set $M$.

**Proposition 5.2** *The Cover Inequality (4) is valid for $CUT(M,N)$.*

*Proof* Trivial, since $CUT(M,N)=\mathrm{conv}\{y\in\{0,1\}^{|M|} : \sum_{k\in M} u_k y_k \geq d(N)\}$ is a minimum knapsack polytope. □

### 5.3 Flow cover inequalities

In this subsection we show that the Flow Cover Inequality can be derived by sequential lifting as a valid inequality for the *Mixed Dicut Polytope*. The following two lemmas can be used to down lift a valid inequality for a *Mixed Dicut Polytope*. Theorem 5.1 shows that such an inequality is a Flow Cover Inequalitiy (5).

**Lemma 5.1** *Let $C$ be a flow cover, let $S \subset C$ and $h \in C \setminus S$. The optimal solution of the minimum knapsack problem*:

$$\begin{aligned}
\min \quad & z = \sum_{k \in S} (u_k - \lambda)^+ y_k + s, \\
& s + \sum_{k \in S} u_k y_k \geq d(J) - \sum_{k \in C \setminus S} u_k - u_h, \qquad (15)\\
& s \geq 0, \\
& y_k \in \{0, 1\}, \quad k \in S
\end{aligned}$$

*is $y_k = 1$ for $k \in S$, $s = (u_h - \lambda)^+$. It follows that the optimal solution value is $z^* = \sum_{k \in S} (u_k - \lambda)^+ + (u_h - \lambda)^+$.*

*Proof* Since $\lambda = \sum_{k \in C} u_k - \sum_{j \in J} d_j$, we have $d(J) - \sum_{k \in C \setminus S} u_k = -\lambda + \sum_{k \in S} u_k$, so we can write:

$$\begin{aligned}
\min \quad & z = s + \sum_{k \in S} (u_k - \lambda)^+ y_k, \\
& s + \sum_{k \in S} u_k y_k \geq -\lambda + \sum_{k \in S} u_k - u_h, \\
& s \geq 0, \\
& y_k \in \{0, 1\}, \quad k \in S.
\end{aligned}$$

Suppose now that $y_k = 0$ in any optimal solution. If we set $y_k = 1$ for $k \in S$, we must add the $y_k$ coefficient $(u_k - \lambda)$, but at the same time $s$ decreases by $u_k$. The net amount $(u_k - \lambda)^+ - u_k$ is negative, a contradiction. □

**Lemma 5.2** *Let $S \subseteq C$ where $C$ is a flow cover and let $s + \sum_{k \in S} (u_k - \lambda)^+ y_k \geq \sum_{k \in S} (u_k - \lambda)^+$ be a valid inequality for $CUT(C, J) \cap \{y_k = 1, k \in C \setminus S\}$. Let $h \in C \setminus S$. The inequality*

$$s + \sum_{k \in S} (u_k - \lambda)^+ y_k + \alpha_h y_h \geq \sum_{k \in S} (u_k - \lambda)^+ + \alpha_h$$

*is valid for $CUT(C, J) \cap \{y_k = 1, k \in C \setminus (S \cup \{h\})\}$ if $\alpha_h = (u_h - \lambda)^+$.*

*Proof* Let $z^*$ be the optimal solution of the problem (15). Using the standard procedure for down-lifting [18], the inequality $s + \sum_{k \in S} (u_k - \lambda)^+ y_k + \alpha_h y_h \geq$

$\sum_{k\in S}(u_k-\lambda)^+ + \alpha_h$ is valid iff $\alpha_h = z^* - \sum_{k\in S}(u_k-\lambda)^+$. By Lemma 5.1 we have that $z^* = \sum_{k\in S}(u_k-\lambda)^+(u_h-\lambda)^+$ and the proof is complete. □

Now we are able to prove the following theorem.

**Theorem 5.1** *The Flow Cover Inequality* (5) *is valid for* $CUT(C,J)$

*Proof* The inequality $s \geq 0$ is valid for $CUT(C,J)\cap\{y_k=1, k\in C\}$. Applying iteratively Lemma 5.2 to compute lifting coefficients, we have that the inequality

$$s+\sum_{k\in C}(u_k-\lambda)^+ y_k \geq \sum_{k\in C}(u_k-\lambda)^+ \tag{16}$$

is valid for $CUT(C,J)$. By subtracting the equality $\sum_{k\in M}\sum_{j\in J} f_{kj} = d(J)$ to the (16) multiplied by $-1$, we obtain the Flow Cover Inequality $\sum_{k\in C}\sum_{j\in J} f_{kj} + \sum_{k\in C}(u_k-\lambda)^+(1-y_k) \leq d(J)$. □

### 5.4 Effective capacity inequalities

Let $K\subset M$ be a subset of facilities and suppose that $G_o(M\cup N, O\cup A)$ is sparse. Let $J_k\subseteq N$ be the subset of clients reachable by the facility $k\in K$ and let $J^K=\bigcup_{k\in K} J_k$ be the subset of clients reached by all facilities in $K$. The set of arcs $\{(o,k)\in O: k\in K\}\cup\{(k,j)\in A: k\in M\setminus K, j\in J_k\}$ define a mixed dicut on $G_o$. Let $\overline{u}_k = \min\{u_k, d(J_k)\}$. As in [1], the Mixed Dicut Inequality defined by $K$ and $J^K$ can be strengthened as:

$$\sum_{k\in K}\overline{u}_k y_k + s \geq \sum_{j\in J^K} d_j$$

that is valid for $CAP(G,d,u)$. The *Effective Mixed Dicut Polytope* is:

$$ECUT(K) = \text{conv}\left\{y_k\in\{0,1\}, k\in K, s\in\mathbb{R}_+ : \sum_{k\in K}\overline{u}_k y_k + s \geq d(J^K)\right\}.$$

Studying this polytope we can define new cutting planes families valid for $CAP(G,d,u)$. Particularly, if $C\subset M$ is an *effective flow-cover* (Sect. 3.3), using the lifting procedure described in Sect. 5.3 for the Flow Cover Inequalities, we obtain the *Effective Capacity Inequality*

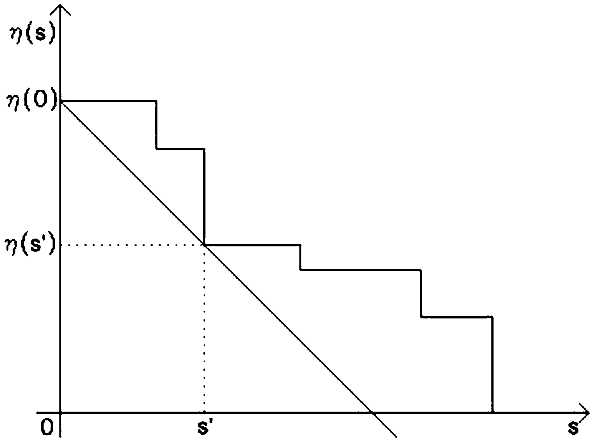
$$\sum_{k\in C}\sum_{j\in J_k} f_{kj} + \sum_{k\in C}(\overline{u}_k-\lambda)^+(1-y_k) \leq d(J^K)$$

defined in Sect. 3.3.

## 6 More families of valid inequalities

Let $Y_{\overline{s}}(K,J) = \{(y,s)\in Y(K,J): s=\overline{s}\}$ be the subset of the incidence vectors of the feasible solutions of a Mixed Dicut Inequality having the property that $s=\overline{s}$. Let $CUT_{\overline{s}}(K,J) = \text{conv}(Y_{\overline{s}}(K,J))$.

**Fig. 1** The function $\eta(s)$




Another way to generate facets of $CUT(K, J)$ is to set $s = \overline{s}$, consider facets of the underlying polytope $CUT_{\overline{s}}(K, J)$ and reintroduce the continuous variable $s$ by lifting. This procedure leads to new families of valid inequalities for $CUT(K, J)$ and, consequently, for $CAP(G, u, d)$. Marchand and Wolsey [17] introduced a lifting procedure for the case where $\sum_{k\in K} u_k \geq d(J)$ and $\overline{s} = 0$.

**Theorem 6.1** (Marchand and Wolsey [17]) *Let* $CUT_0(K, J) = \text{conv}\{(y, s) \in Y(K, J) : s = 0\}$ *and let* $\sum_{k\in K} \pi_k y_k \geq \pi_0$ *be a valid inequality for* $CUT_0(K, J)$. *The inequality*

$$\sum_{k\in K} \pi_k y_k \geq \pi_0 - \frac{s}{\beta} \tag{17}$$

*is valid for* $CUT(K, J)$ *iff* $\beta \leq \min_{s>0} \frac{s}{\pi_0 - \eta(s)}$, *where*

$$\eta(s) = \min\left\{\sum_{k\in K} \pi_k y_k : \sum_{k\in K} u_k y_k \geq d(J) - s,\ y \in \{0, 1\}^{|K|}\right\}.$$

*Proof* We will give a geometric proof of the validity of (17). $\eta(s)$ is a step function (Fig. 1). The inequality $\sum_{k\in K} \pi_k y_k \geq \gamma(s)$ is valid for $CUT(K, J)$ if $\gamma(s) \leq \eta(s)$, for each $s \in \mathbb{R}$.

Let $s'$ be such that $\max_{s>0} \frac{\pi_0 - \eta(s)}{s} = \frac{\pi_0 - \eta(s')}{s'}$. Let $\gamma(s) = \pi_0 - \frac{s}{\beta}$ be the line of minimum slope intersecting $\eta(s)$ and crossing the points $(0, \pi_0)$ and $(s', \eta(s'))$. It is easy to observe that $\eta(s) \geq \gamma(s)$ and the proof is complete. □

The lifting procedure can be generalized by removing the restriction that $\sum_{k\in K} u_k \geq d(J)$.

Let $s_{\min} = \max(0, d(J) - \sum_{k\in K} u_k)$ be the minimum value of $s$ that guarantees that the set of the feasible solutions $Y(K, J)$ is nonempty.

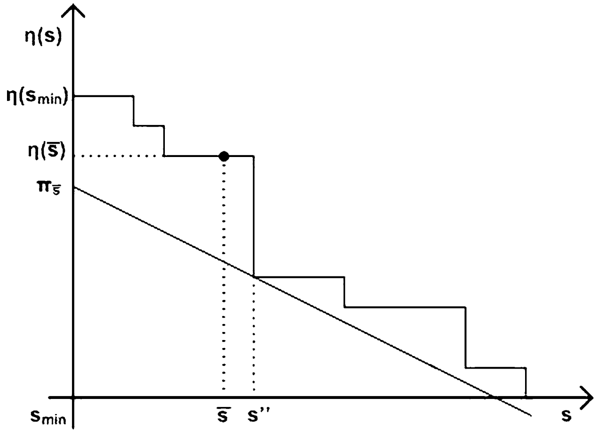



**Fig. 2** The function $\eta(s)$ in the case $\sum_{k\in K} u_k < d(J)$

**Theorem 6.2** *Let* $\bar{s} \geq s_{\min}$ *and let* $\sum_{k\in K} \pi_k y_k \geq \pi_{\bar{s}}$ *be a valid inequality for the polytope* $CUT_{\bar{s}}(K, J)$.

*The inequality*

$$\sum_{k\in K} \pi_k y_k \geq \pi_{\bar{s}} - \frac{s - s_{\min}}{\theta} \tag{18}$$

*is valid for* $CUT(K, J)$ *if* $\theta \leq \min_{s>s_{\min}} \frac{s-s_{\min}}{\pi_{\bar{s}}-\eta(s)}$, *where*

$$\eta(s) = \min\left\{\sum_{k\in K} \pi_k y_k : \sum_{k\in K} u_k y_k \geq d(J) - s,\ y \in \{0,1\}^{|K|}\right\}. \tag{19}$$

*Proof* The function $\eta(s)$ is defined for $s \geq s_{\min}$. Let $s''$ be such that $\max_{s>s_{\min}} \frac{\pi_{\bar{s}}-\eta(s)}{s-s_{\min}} = \frac{\pi_{\bar{s}}-\eta(s'')}{s''-s_{\min}}$. Let $\pi_{\bar{s}} - \frac{s-s_{\min}}{\theta}$ be the line of minimum slope intersecting $\eta(s)$ and crossing the points $(s_{\min}, \pi(\bar{s}))$ and $(s'', \eta(s''))$ (Fig. 2). It is immediate that $\eta(s) \geq \pi_{\bar{s}} - \frac{s-s_{\min}}{\theta}$ and the proof is complete. □

## 7 The separation procedure

The above results suggest that the facial structure of $CUT(K, J)$ can be exploited to generate strong cutting planes. We propose a two-step separation algorithm based on the enumeration of the facets of $CUT(K, J)$. The first step consists of choosing a Mixed Dicut Inequality $\sum_{k\in K} u_k y_k + \sum_{k\in M\setminus K} \sum_{j\in J} f_{kj} \geq d(J)$, with $K$ suitably small, as a *base inequality*. Then for each base inequality we enumerate the feasible solutions of $Y(K, J)$ and run a facet-enumeration routine to list the facets of $CUT(K, J)$, returning violated cutting planes.

An accurate selection of the base inequalities and a fast enumeration procedure of the feasible solutions are crucial to the success of the separation algorithm.

7.1 Selection of base inequalities

Let $(y^*, f^*)$ be the current fractional solution. The procedure to select candidate base inequalities is inspired by the separation heuristic described in [2] for the Flow Cover Inequalities and is based on the following observations.

*Remark 7.1* A valid inequality for $CUT(K, J)$ can be violated by $(y^*, f^*)$ only if at least one of the variables $y_k$, $k \in K$, has a fractional value.

*Remark 7.2* The chances of finding a violated valid inequality for $CUT(K, J)$ increase if $\overline{s}$ is small (the smaller $\overline{s}$, the larger the r.h.s. of the inequality $\sum_{k\in K} u_k y_k \geq d(J) - \overline{s}$).

*Remark 7.3* The chances of finding a violated valid inequality for $CUT(K, J)$ increase if we choose $K$ and $J$ such that the facilities in $K$ serve only the clients in $J$ in the current fractional solution.

The following heuristic attempts to generate base inequalities according to Remarks 7.1, 7.2 and 7.3. The selection procedure is iterated for each triple of facilities $K = \{k1, k2, k3\} \subset M$ such that there exists at least a $j$ with $0 < y^*_{kj} < 1$. The main steps of the procedure are summarized below. The facet-enumeration routine is efficient only if $|K|$ is suitably small. Let $\max_K$ denote the maximum number of $y$-variables in a *base inequality* used for the facet enumeration. Computational experience showed us that a good choice is to set $\max_K = 9$. Moreover the parameter $\theta$ used in the steps 1 and 4 is set to 0.01.

**Procedure 7.1** (Selection of the base inequalities)

Initialize $K$ and $J$

(0) Let $K = \{k1, k2, k3\}$ be a set of facilities with the property that $0 < y^*_{k1} < 1$.
(1) Let $J$ be the set of all clients $j \in J$ such that $\sum_{k\in K} f^*_{kj} \geq \theta d_j$.
(2) Generate the base inequality $\sum_{k\in K} u_k y_k + s \geq d(J)$.

Enlarge $K$

(3) Let $k = \arg\max_{k\in M\setminus K} \sum_{j\in J} f_{kj}$. Add $k$ to $K$.
(4) Add to $J$ all the clients $j \in N \setminus J$ such that $\sum_{k\in K} f^*_{kj} \geq \theta d_j$.
(5) If $|K| \leq \max_k$, $\sum_{k\in K} u_k y_k + s \geq d(J)$ is a base inequality else *STOP*.

Adjust $J$

(6) For each $j \in J$ compute $\phi(j) = \sum_{k\in M\setminus K} f^*_{kj}$.
(7) Let $j_{\min} = \arg\max_{j\in J} \phi(j)$.
(8) If $\sum_{k\in K} u_k < d(J)$ then remove $j_{\min}$ from $J$, generate the base inequality $\sum_{k\in K} u_k y_k + s \geq d(J)$ and go to step 3.

### 7.2 Enumeration of the feasible solutions satisfying a base inequality

Let $K \subseteq M$, $J \subseteq N$ and let $Y(K, J)$ denote the set of the feasible solutions satisfying the Mixed Dicut Inequality (14). For any partial solution $\overline{y}$, let $s_{\min}(\overline{y}) = \max(0, d(J) - \sum_{k \in K} u_k \overline{y}_k)$ be the minimum value of $\overline{s}$ that makes the solution $(\overline{y}, \overline{s})$ feasible for the (14). We refer to any solution $(y, s_{\min}(y))$ as a "minimal" feasible solution of (14) and we denote by $Y_{s_{\min}}(K, J) = \{(y, s) \in Y(K, J) : s = s_{\min}\}$ the set defined by the minimal feasible solutions of (14). In this section we will prove that the facets of $CUT(K, J)$ are in one-to-one correspondence with a subset of the facets of $\mathrm{conv}(Y_{s_{\min}}(K, J))$. We observe that the polytope $\mathrm{conv}(Y_{s_{\min}}(K, J))$ is full-dimensional.

**Lemma 7.1** *An inequality $\sum_{k \in K} \alpha_k y_k + \sigma s \geq \beta$ is valid for $CUT(K, J)$ only if $\sigma \geq 0$.*

*Proof* Let $\sum_{k \in K} \alpha_k y_k + \sigma s \geq \beta$ be a valid inequality for $CUT(K, J)$ and let $(y, s) \in Y(K, J)$ be a feasible solution satisfying (14). If $\sigma < 0$, it is always possible to set $s$ large enough (and the solution remains feasible) such that $\sum_{k \in K} \alpha_k y_k + \sigma s < \beta$, a contradiction. □

The following theorem proves that any facet of $CUT(K, J)$ is also a facet of $\mathrm{conv}(Y_{s_{\min}}(K, J))$.

**Theorem 7.1** *An inequality*

$$\sum_{k \in K} \alpha_k y_k + \sigma s \geq \beta \tag{20}$$

*is facet-defining for* $\mathrm{conv}(Y_{s_{\min}}(K, J))$ *if it is facet-defining for $CUT(K, J)$.*

*Proof* If $\sum_{k \in K} \alpha_k y_k + \sigma s \geq \beta$ is valid for $CUT(K, J)$ then $\sigma \geq 0$. If $\sum_{k \in K} \alpha_k y_k + \sigma s \geq \beta$ is facet-defining for $CUT(K, J)$, there exist $|K| + 1$ affinely independent feasible solutions satisfying (20) as an equality (roots).

Let $(\overline{y}, \overline{s})$ be a root of (20) and suppose that $\overline{s} > s_{\min}$. Since $(\overline{y}, \overline{s})$ is a root, we have $\sum_{k \in K} \alpha_k \overline{y}_k + \sigma \overline{s} = \beta$. But $(\overline{y}, s_{\min})$ is still a feasible solution and, since $\sigma \geq 0$, it violates (20), i.e. $\sum_{k \in K} \alpha_k \overline{y}_k + \sigma s_{\min} < \beta$, a contradiction.

It follows that all the $|K| + 1$ roots are of the form $(\overline{y}, s_{\min})$ and belong to $Y_{s_{\min}}(K, J)$, so we can conclude that (20) is facet-defining for $\mathrm{conv}(Y_{s_{\min}}(K, J))$. □

**Theorem 7.2** *Let*

$$\sum_{k \in K} \alpha_k y_k + \sigma s \geq \beta \tag{21}$$

*be a facet-defining inequality of* $\mathrm{conv}(Y_{s_{\min}}(K, J))$. *The* (21) *inequality is facet-defining for $CUT(K, J)$ if and only if $\sigma \geq 0$.*

*Proof* Any facet of $\mathrm{conv}(Y_{s_{\min}}(K, J))$ with $\sigma \geq 0$ is valid for $CUT(K, J)$ and there exist $|K| + 1$ affinely independent feasible solutions belonging to $Y_{s_{\min}}(K, J) \subseteq Y(K, J)$, satisfying it as an equality. It follows that if (21) is a facet of $\mathrm{conv}(Y_{s_{\min}}(K, J))$, then it is also a facet of $CUT(K, J))$. If $\sigma < 0$, by Lemma 7.1 it is not even valid for $CUT(K, J)$. □

Theorems 7.1 and 7.2 prove that the facets of $CUT(K, J)$ are in one-to-one correspondence with the facets of $\mathrm{conv}(Y_{s_{\min}}(K, J))$ with $\sigma \geq 0$.

A main algorithmic consequence of this result is that to generate facets of $CUT(K, J)$ we need only to enumerate the "minimal" feasible solutions of the set $Y_{s_{\min}}(K, J)$.

## 8 The branch-and-cut-and-price algorithm

MIP solvers cannot handle large *CFLP* instances (i.e. 500 facilities and 500 clients) on standard workstations. This is mainly due to the difficulty of solving large scale LP-relaxations, requiring a large amount of memory.

To deal with large-scale instances, we present a Branch-and-Cut-and-Price algorithm, based on *delayed column generation* [11] to solve the LP-relaxation at each node of the search tree, and on the separation algorithm described in Sect. 7 to tighten the formulation.

The separation procedure described in Sect. 7 is applied only to the base inequalities with no more than $\max_K\ y$ variables. In addition we look for violated *Flow Cover* inequalities over the base inequalities with more then $\max_K\ y$variables.

Delayed column generation allows to solve large scale linear programs by using a relatively small subset of the original columns. An initial subset of "promising" columns, forming the *core problem*, is selected according to some criteria. The core problem is solved to optimality and the dual variables are used to price out the external columns, i.e. to compute reduced costs for the columns not belonging to the core. If the reduced costs are non-negative the procedure stops, else the core problem is enlarged by adding a subset of columns with negative reduced cost.

Delayed column generation and cut generation procedures are used iteratively to reduce the gap between lower and upper bound.

Computational experience has shown that cutting planes are more effective to reduce the gap in the upper level nodes of the search tree, so we run separation only at the nodes whose depth is less or equal than 2. A violated cutting plane is added to the problem only if the amount of the violation is larger than a given threshold $\epsilon$ (in our experiments we set $\epsilon = 0.01$).

The cutting planes generated by the separation procedure have the form:

$$\sum_{k \in K} \alpha_k y_k + \sigma \sum_{k \in M \setminus K} \sum_{j \in J} f_{kj} \geq \beta$$

and are globally valid. Once a violated cut has been generated, it is added to the current formulation. In the subsequent iterations it can be deleted from the formulation if its slack keeps larger than a given threshold for a given number of iterations.

In addition we use a cut pool to store, for each cut, the sets $K$ and $J$. These information are sufficient to compute the reduced cost of the external variables (e.g. not belonging to the core) and to keep the inequality globally valid when a new variable is added to the core.

Let $f_{\overline{kj}}$ be an external variable. To compute its reduced cost we have to consider the dual variables associated to all the inequalities in the current formulation, such that $\overline{k} \notin K$ and $\overline{j} \in J$. If $f_{\overline{kj}}$ is moved to the core, it has to be added with coefficient $\sigma$ to each added inequality with $\overline{k} \notin K$ and $\overline{j} \in J$ to keep it globally valid.

In the remainder of this section we will detail the main modules of the algorithm.

## 8.1 Selection of the "core problem"

For the effectiveness of delayed column generation it is crucial to choose a "promising" initial set of variables forming the *core problem*. To this purpose we adopt a heuristic based on *Lagrangean reduced costs* as in [6]. Let $S \subseteq M$ be a subset of facilities, the *core problem* is defined by all the facility variables $y_k$ and a subset $X(S)$ of the $f_{kj}$ variables leaving the facilities of $S$. We denote the core problem by the pair $(S, X(S))$.

To select the variables of the core problem, we solve the Lagrangean relaxation of the equality constraints (1) by the subgradient method.

The Lagrangian function is:

$$\theta(\lambda) = \sum_{k \in M} h_k y_k + \sum_{k \in M} \sum_{j \in N} (c_{kj} - \lambda_j) f_{kj}.$$

For each $k \in M$, let $F_k$ be the Lagrangean reduced cost of $y_k$, i.e.

$$F_k = f_k + \sum_{j \in N} (c_{kj} - \lambda_j^*) f_{kj}^*, \quad k \in M,$$

where $\lambda^*$ is the vector of multipliers computed by the subgradient method and $f^*$ is the corresponding solution of the Lagrangean relaxation problem.

In order to select a "promising" pair $(S, X(S))$, we sort in non-decreasing order the facilities in $M$ according to the value $F_k$, then we choose its elements by extracting them in order from this list by the following rule.

Let $r$ be the ratio between the total capacity of the facilities in $M$ and the total client demand of the clients in $J$. To decide the value of $|S|$, we could observe that $\lceil |M|/r \rceil$ is an upper bound on the minimal number of open facilities needed to satisfy the total client demand. In other words, we can say that at least a feasible solution with less than $\lceil |M|/r \rceil$ open facilities exists. To increase the probability of including in $S$ all the facilities opened in the optimal solution of the problem, we refine the estimation of $|S|$ by multiplying $\lceil |M|/r \rceil$ by a factor $w = 1.1$.

Let $\gamma = \lceil w|M|/r \rceil$. We define $S$ as the set of the first $\gamma$ facilities sorted in increasing order with respect to the $F_k$.

The set $X(S)$ is defined by setting a threshold value $\delta$ and by selecting all the arcs leaving the nodes of $S$, whose reduced cost $\bar{c}_{kj} = c_{kj} - \lambda_j^*$ does not exceed $\delta$.

With the aim of keeping the core size small, we initially set

$$\delta = \text{average}\{\bar{c}_{kj} | k \in S,\ j \in J\}$$

and then we attempt to reduce it until the condition is satisfied that at least three arcs belonging to $X(S)$ must leave each client. It is trivial to observe that this rule cannot guarantee the feasibility of the core problem $(S, X(S))$. Moreover, our experiments showed that the core problem has very high probability to contain a feasible solution if this rule is satisfied. If such a feasible solution does not exist, it is possible to reduce $\delta$ until the core problem becomes feasible.

## 8.2 Pricing the external columns

Starting from the core problem, the LP-relaxation is solved by *delayed column generation*. Since the LP is highly degenerate, the addition of variables with large negative reduced costs very often does not alter the current LP solution, while it increases the size of the core, which soon becomes unmanageable. To contrast this phenomenon we adopt the following column generation strategy.

Let $(x, y)$ be the current solution of the core problem, let $\gamma_k$ be the optimal reduced cost of variable $y_k$ in the core problem and let $\lambda_j$, $j \in N$ be the dual variables associated with the equality constraints (1). For each $k \in M$, we define the *facility reduced cost* $\rho_k$ as:

$$\rho(k) = \gamma_k + \sum_{j \in N} (c_{kj} - \lambda_j)^-, \quad k \in M,$$

where $(c_{kj} - \lambda_j)^- = \min(0, (c_{kj} - \lambda_j))$.

The facilities of $M$ are sorted in increasing order according to the *facility reduced cost* $\rho_k$, $k \in M$. Let $K = k_1, \ldots, k_h$ be the set of the first $h$ facilities after sorting. We add to the core problem only the variables $x$ leaving the facilities in $K$ whose reduced cost does not exceed the threshold $\delta$.

The main column generation iteration can be summarized as follows.

Step 0 Solve the core problem $(S, X(S))$. Let $\lambda_j$, $j \in N$ be the optimal dual variables for the equality constraints of the core problem and let $\gamma_k$, $k \in M$ be the optimal reduced costs for the variables $y_k$, $k \in M$.

Step 1 For each $k \in M$, compute the node reduced cost
$\rho(k) = \gamma_k + \sum_{j \in N} (c_{kj} - \lambda_j)^-$, $k \in M$.

Step 2 Sort in ascending order the nodes according to the $\rho(k)$
$\rho(k_1) \leq \rho(k_2) \leq \cdots \leq \rho(k_m)$,
let $K = k_1, \ldots, k_h$ ($h \leq |N|$ fixed parameter).

Step 3 If $\rho(k_1) \geq 0$ then Stop.

Step 4 Set
$S := S \cup K$,
$X(S) := X(S) \cup \{f_{kj} : \pi_{kj} < \delta, k \in K,\ j \in N\}$,
being $\pi_{kj}$ the reduced cost of the variable $f_{kj}$.

Step 5 Return to Step 0.

We remark that the setting of parameter $h$ determines a trade-off between the number of iterations and the size of the optimal master problem. By setting $h$ to a large value, the size of the optimal master problem rapidly increases, while setting $h$ to small values leads to a smaller size of the master problem, but to a large number of iterations. In our experiments, setting $h = 4$ (and $\delta = 0.01$) turned out to be a good compromise.

### 8.3 Branching strategy

The branching strategy is to set node variables $y_k$ to 1 if facility $k \in M$ is open, 0 otherwise. The node selection criterion is "best-bound". The variable selection criterion is to choose the "most" fractional node variable, i.e. the variable whose fractional value is closest to 0.5. Finally we use standard reduced cost fixing to reduce the size of the problem.

### 8.4 Primal heuristic

An initial upper bound is obtained by running a MIP solver over the core problem, as suggested in [5]. In the search tree, we use a *plunging heuristic*, which consists of rounding to the nearest integer those variables whose fractional value is close either to 0 or to 1.

## 9 Computational results

In this section we report on the computational experience with the Branch-and-Cut-and-Price algorithm. The experiments ran on a Pentium IV 1.7 GHz Personal Computer, with 512 MB RAM, using Cplex 8.1 as *LP* solver and the routine *Qhull* [8], developed by the Geometric Center of University of Minnesota, as a facet-generation routine.

Two different test-beds, denoted respectively as *TBED*1 and *TBED*2, were used to evaluate the algorithm.

The instances of *TBED*1 were randomly generated according to Cornuejols, Sridharan and Thizy [13], Aardal [1, 2] and Barahona and Chudak [7]. The generation procedure is summarized below:

(a) Points representing the clients and the facilities are uniformly randomly generated in a unit square. Transportation costs per unit flow are then computed by multiplying the Euclidean distances by 10.
(b) Demands are drawn from $U[5, 35]$ i.e. they are uniformly distributed in the interval $[5, 35]$.
(c) Capacities $u_k$ are drawn from $U[10, 160]$ and then are scaled by using the *capacity ratio* $r = \sum_{k\in M} u_k/d(N)$.
(d) Fixed costs are generated according to the formula $h_k = U[0, 90] + U[100, 110]\sqrt{u_k}$, to take into account the economies of scale.

In [1, 2, 13] the ratio $r$ was set to 2, 3, 5, 10. We could observe that setting $r = 2, 3$ leads to easy instances, efficiently solved to exact optimality by the Lagrangian heuristic, so in our experiments we set $r = 5, 10, 15, 20$.

The instances of *TBED*1 are organized into 5 groups, each containing instances of the same size. The sizes ($|M| \times |N|$) here considered are: $300 \times 300$, $300 \times 1500$, $500 \times 500$, $700 \times 700$ and $1000 \times 1000$. Each group includes four different subsets of problems according to the ratio $r$ (we set $r = 5, 10, 15, 20$). Five instances have been randomly generated for each value of $r$. The test bed described is available at the web page http://www.ing.unisannio.it/boccia/.

The test-bed *TBED*2 consists of the twelve largest instances from the OR library [9], namely those denoted as *CAPA*, *CAPB* and *CAPC*. Each set consists of 4 instances with 100 facilities and 1000 clients.

Tables 1–5 report on the experimental results for *TBED*1. Column *Ratio* is the value of $r$, $UB_{\text{Heur}}$ is the initial upper bound and column $UB_{\text{Heur}}$ *Time* is the time spent to compute the initial upper bound. Columns *LP* and $LP_{\text{Cuts}}$ report on the value of the LP relaxation before and after the addition of cutting planes, respectively. Columns *BLB* and *BUB* report respectively on the best lower bound and on the best upper bound yielded by Branch-and-Cut-and-Price within the time limit. Column *%Gap* shows the % of the integrality gap closed by the cut generation routine at the root node, i.e. *%Gap* = $(LP_{\text{Cuts}} - LP)/(BUB - LP)$. Columns *B&C&P nodes* and *B&C&P Time* report respectively on the number of nodes of the search tree and on the total computation time spent by the Branch-and-Cut-and-Price algorithm.

We remark that in our computational experience we consider the strong formulation (formulation F1 + variable upper bounds + the aggregate capacity constraint). The cutting planes added to the formulation are the *strong cutting planes* generated by the separation procedure described in Sect. 7 and the *Flow Cover* inequalities generated by the Aardal's separation procedure [2].

The time limit for the Branch-and-Cut-and-Price algorithm was set to 100000 seconds. All the instances of *TBED*1 were solved to optimality within the time limit, with the exceptions of $I$1000-11 and $I$1000-12 where, however, the algorithm yielded a very small gap.

We observe that the cut generation strategy led very often to a significant reduction of the integrality gap, with the exception of $r = 5$, where the LP lower bound was already very close to the optimal solution.

Tables 1, 2 also report on the lower bound at the root node computed by Cplex 8.1 solver ($Cplex_{\text{root}}$), and the total time necessary to Cplex to solve the instances (*Cplex Time*). We set the Cplex cut generation option to the *aggressive mode* for Cover and Flow cover Inequalities, while all the other parameters were set to their default values.

We observe that Cplex cut generation routines cannot appreciably raise the lower bound, so even for the smaller instances ($300 \times 300$ and $500 \times 500$), our algorithm outperforms Cplex. Furthermore we remark that Cplex could not deal with the larger instances, due to the memory requirements.

Experimental results for *TBED*2 (OR-Library) are reported in Table 6. The OR-Library instances are widely used in literature, but they are generally easier than those of *TBED*1. The easiest (i.e. those solvable in less than 400 secs.) instances of *TBED*2 (*CAPA*-2, *CAPA*-3, *CAPA*-4) are efficiently solved by Cplex, which slightly outperforms our algorithm. A completely different result is obtained for the hardest instances *CAPB*-3, *CAPB*-3, *CAPB*-4 and *CAPC*-2, where cut generation turned out to be very useful.

**Table 1** Computational results for test instances with 300 facilities and 300 clients

| *Name* | *Ratio* | $UB_{\text{Heur}}$ | $UB_{\text{Heur}}$ *Time* (s) | *LP* | $CPLEX_{\text{root}}$ | $LP_{\text{Cuts}}$ | % $Gap_{\text{Cuts}}$ | *BLB* | *BUB* | *B&C&P nodes* | *B&C&P Time* (s) | *Tot. Time* (s) | *CPLEXTime* (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I300-1 | 5 | 16350,66 | 13,14 | 16292 | 16293,09 | 16305,56 | 23,11 | 16350,66 | 16350,66 | 294 | 385,41 | 398,55 | 1019,20 |
| I300-2 | 5 | 15948,44 | 30,53 | 15862,21 | 15886,06 | 15908,62 | 53,82 | 15948,44 | 15948,44 | 707 | 784,59 | 815,12 | 1531,70 |
| I300-3 | 5 | 15474,84 | 13,77 | 15414,95 | 15421,71 | 15448,49 | 56,00 | 15474,84 | 15474,84 | 138 | 250,69 | 264,46 | 1168,00 |
| I300-4 | 5 | 17989,97 | 16,35 | 17926,00 | 17927,69 | 17942,59 | 25,93 | 17989,97 | 17989,97 | 542 | 698,48 | 714,83 | 1367,06 |
| I300-5 | 5 | 18037,61 | 17,61 | 17962,8 | 17966,73 | 18003,96 | 55,02 | 18037,61 | 18037,61 | 498 | 731,85 | 749,46 | 1057,84 |
| I300-6 | 10 | 11255,48 | 5,73 | 11177,54 | 11177,84 | 11213,94 | 49,42 | 11251,19 | 11251,19 | 176 | 445,84 | 452,57 | 1150,41 |
| I300-7 | 10 | 11393,29 | 8,62 | 11307,61 | 11308,64 | 11348,68 | 48,37 | 11392,52 | 11392,52 | 290 | 587,63 | 596,25 | 1689,50 |
| I300-8 | 10 | 11377,34 | 7,58 | 11331,09 | 11333,18 | 11351,62 | 44,39 | 11377,34 | 11377,34 | 26 | 108,61 | 116,19 | 559,39 |
| I300-9 | 10 | 10980,05 | 8,29 | 10787,81 | 10790,35 | 10836,91 | 54,41 | 10878,05 | 10878,05 | 146 | 720,11 | 728,4 | 1437,89 |
| I300-10 | 10 | 11265,13 | 7,42 | 11174,33 | 11183,18 | 11220,10 | 78,32 | 11232,77 | 11232,77 | 16 | 112,23 | 119,65 | 558,94 |
| I300-11 | 15 | 10024,47 | 6,64 | 9977,14 | 9980,05 | 10013,70 | 78,12 | 10023,94 | 10023,94 | 12 | 80,17 | 86,81 | 268,72 |
| I300-12 | 15 | 9398,19 | 6,61 | 9300,66 | 9304,21 | 9322,28 | 60,09 | 9336,64 | 9336,64 | 24 | 138,83 | 145,44 | 311,08 |
| I300-13 | 15 | 10107,95 | 6,53 | 9980,73 | 9981,73 | 10019,09 | 49,33 | 10058,49 | 10058,49 | 182 | 152,14 | 158,67 | 699,20 |
| I300-14 | 15 | 9768,72 | 7,55 | 9669,64 | 9670,32 | 9688,51 | 63,51 | 9699,35 | 9699,35 | 15 | 352,21 | 359,76 | 481,97 |
| I300-15 | 15 | 9909,3 | 5,66 | 9788,4 | 9798,21 | 9826,81 | 71,43 | 9842,17 | 9842,17 | 28 | 136,03 | 141,69 | 253,88 |
| I300-16 | 20 | 9331,77 | 4,79 | 9096,07 | 9102,65 | 9147,07 | 81,31 | 9158,79 | 9158,79 | 18 | 114,02 | 118,81 | 239,09 |
| I300-17 | 20 | 9227,56 | 5,48 | 9133,87 | 9135,21 | 9148,75 | 39,37 | 9171,67 | 9171,67 | 42 | 283,22 | 288,70 | 382,89 |
| I300-18 | 20 | 9584,07 | 5,25 | 9533,71 | 9535,80 | 9538,49 | 24,04 | 9553,59 | 9553,59 | 14 | 96,41 | 101,66 | 323,89 |
| I300-19 | 20 | 9111,4 | 5,14 | 9013,6 | 9016,52 | 9042,78 | 72,75 | 9053,71 | 9053,71 | 12 | 182,95 | 188,09 | 302,21 |
| I300-20 | 20 | 9068,42 | 4,92 | 9044,75 | 9044,83 | 9046,21 | 92,40 | 9046,33 | 9046,33 | 6 | 9,94 | 14,86 | 78,06 |

**Table 2** Computational results for test instances with 500 facilities and 500 clients

| *Name* | *Ratio* | $UB_{\text{Heur}}$ | $UB_{\text{Heur}}$ *Time* (s) | *LP* | $CPLEX_{\text{root}}$ | $LP_{\text{Cuts}}$ | % $Gap_{\text{Cuts}}$ | *BLB* | *BUB* | *B&C&P* *nodes* | *B&C&P* *Time* (s) | *Tot. Time* (s) | *CPLEXTime* (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| I500-1 | 5 | 26412,41 | 36,6 | 26374,61 | 26374,85 | 26389,11 | 22,37 | 26412,41 | 26412,41 | 430 | 1244,58 | 1281,18 | 7924,83 |
| I500-2 | 5 | 28144,83 | 109,36 | 28082,01 | 28082,19 | 28112,05 | 61,64 | 28130,74 | 28130,74 | 1624 | 3957,41 | 4066,77 | 18621,55 |
| I500-3 | 5 | 27915,12 | 80,16 | 27857,05 | 27857,05 | 27881,38 | 51,26 | 27904,51 | 27904,51 | 1056 | 2481,22 | 2561,38 | 5277,44 |
| I500-4 | 5 | 28159,03 | 128,23 | 28093,08 | 28095,70 | 28118,12 | 37,38 | 28159,03 | 28159,03 | 1326 | 2556,72 | 2684,95 | 9524,72 |
| I500-5 | 5 | 24702,77 | 136,12 | 24645,63 | 24645,86 | 24673,95 | 49,56 | 24702,77 | 24702,77 | 1856 | 4762,75 | 4898,87 | 15085,59 |
| I500-6 | 10 | 15756,82 | 13,76 | 15672,36 | 15674,79 | 15737,70 | 77,36 | 15756,82 | 15756,82 | 70 | 728,08 | 741,84 | 6426,59 |
| I500-7 | 10 | 16110,8 | 22,47 | 16029,01 | 16030,09 | 16083,96 | 68,45 | 16109,28 | 16109,28 | 272 | 2131,50 | 2153,97 | 8379,30 |
| I500-8 | 10 | 16041,73 | 24,85 | 15969,94 | 15970,57 | 16013,99 | 61,36 | 16041,73 | 16041,73 | 372 | 3140,30 | 3165,15 | 9434,33 |
| I500-9 | 10 | 16327,71 | 48,95 | 16237,47 | 16238,33 | 16271,21 | 37,43 | 16327,71 | 16327,71 | 1556 | 1919,58 | 1968,53 | 28226,28 |
| I500-10 | 10 | 15815,13 | 19,37 | 15751,91 | 15752,02 | 15793,18 | 65,27 | 15815,13 | 15815,13 | 112 | 1074,33 | 1093,70 | 4709,19 |
| I500-11 | 15 | 13442,91 | 17,76 | 13406,01 | 13406,25 | 13423,97 | 56,64 | 13437,72 | 13437,72 | 40 | 226,89 | 244,65 | 1292,37 |
| I500-12 | 15 | 14698,03 | 33,71 | 14610,1 | 14610,52 | 14657,67 | 73,27 | 14675,02 | 14675,02 | 94 | 1136,27 | 1169,98 | 3401,39 |
| I500-13 | 15 | 13667,23 | 12,71 | 13631,35 | 13631,38 | 13648,24 | 48,39 | 13666,25 | 13666,25 | 54 | 341,31 | 354,02 | 2732,64 |
| I500-14 | 15 | 13595,1 | 21,74 | 13514,93 | 13516,25 | 13555,93 | 63,00 | 13580,01 | 13580,01 | 88 | 1211,64 | 1233,38 | 3909,23 |
| I500-15 | 15 | 13930,68 | 35,66 | 13833,42 | 13834,19 | 13864,24 | 51,34 | 13896,76 | 13896,76 | 48 | 309,44 | 345,10 | 4954,72 |
| I500-16 | 20 | 12786,29 | 8,59 | 12492,68 | 12493,64 | 12563,69 | 77,34 | 12584,49 | 12584,49 | 92 | 863,81 | 872,40 | 2503,06 |
| I500-17 | 20 | 13347,4 | 17,38 | 13290,26 | 13291,31 | 13326,48 | 63,39 | 13347,4 | 13347,4 | 44 | 582,33 | 599,71 | 1972,23 |
| I500-18 | 20 | 12831,12 | 14,38 | 12780,16 | 12780,75 | 12814,16 | 67,27 | 12831,12 | 12831,12 | 30 | 372,36 | 386,74 | 1241,22 |
| I500-19 | 20 | 13489,61 | 14,41 | 13450,63 | 13451,72 | 13472,63 | 56,44 | 13489,61 | 13489,61 | 50 | 309,47 | 323,88 | 1635,11 |
| I500-20 | 20 | 12342,26 | 14,75 | 12312,46 | 12311,51 | 12331,63 | 64,33 | 12342,26 | 12342,26 | 18 | 210,24 | 224,99 | 1267,27 |

**Table 3** Computational results for test instances with 700 facilities and 700 clients

| *Name* | *Ratio* | $UB_{\text{Heur}}$ | $UB_{\text{Heur}}$ *Time* (s) | *LP* | $LP_{\text{Cuts}}$ | % $Gap_{\text{Cuts}}$ | *BLB* | *BUB* | *B&C&P nodes* | *B&C&P Time* (s) | *Tot. Time* (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| I700-1 | 5 | 36908,58 | 162,22 | 36839,96 | 36880,10 | 60,84 | 36905,93 | 36905,93 | 1390 | 9685,74 | 9847,96 |
| I700-2 | 5 | 34311,71 | 188,44 | 34250,16 | 34284,06 | 55,08 | 34311,71 | 34311,71 | 2846 | 13407,77 | 13596,21 |
| I700-3 | 5 | 34294,63 | 195,14 | 34253,73 | 34275,57 | 53,39 | 34294,63 | 34294,63 | 582 | 1793,42 | 1988,56 |
| I700-4 | 5 | 38090,9 | 263,23 | 38029,87 | 38060,86 | 50,77 | 38090,9 | 38090,9 | 1356 | 5932,48 | 6195,71 |
| I700-5 | 5 | 37802,1 | 309,99 | 37741,12 | 37770,65 | 48,42 | 37802,1 | 37802,1 | 2308 | 13842,76 | 14152,75 |
| I700-6 | 10 | 19927,34 | 50,79 | 19821,97 | 19876,60 | 61,59 | 19910,67 | 19910,67 | 698 | 8855,63 | 8906,42 |
| I700-7 | 10 | 21297,3 | 49,78 | 21195,93 | 21248,75 | 52,10 | 21297,3 | 21297,3 | 714 | 8202,86 | 8252,64 |
| I700-8 | 10 | 20678,87 | 44,09 | 20592,18 | 20628,74 | 53,94 | 20659,96 | 20659,96 | 226 | 3284,31 | 3328,4 |
| I700-9 | 10 | 20990,27 | 86,3 | 20894,19 | 20934,76 | 47,34 | 20979,88 | 20979,88 | 1278 | 16482,74 | 16569,04 |
| I700-10 | 10 | 22060,00 | 49,51 | 21960,88 | 22028,48 | 71,51 | 22055,41 | 22055,41 | 1126 | 18217,81 | 18267,32 |
| I700-11 | 15 | 17164,31 | 22,99 | 17005,24 | 17067,18 | 53,90 | 17120,15 | 17120,15 | 882 | 10366,94 | 10389,93 |
| I700-12 | 15 | 18130,45 | 33,01 | 18048,2 | 18110,54 | 75,82 | 18130,42 | 18130,42 | 104 | 2044,25 | 2077,26 |
| I700-13 | 15 | 17239,96 | 53,44 | 17163,15 | 17211,76 | 63,28 | 17239,96 | 17239,96 | 156 | 2089,08 | 2142,52 |
| I700-14 | 15 | 17337,63 | 32,47 | 17247,16 | 17289,06 | 46,31 | 17337,63 | 17337,63 | 508 | 6951,25 | 6983,72 |
| I700-15 | 15 | 18145,49 | 44,69 | 18069,79 | 18101,94 | 42,47 | 18145,49 | 18145,49 | 132 | 3316,49 | 3361,18 |
| I700-16 | 20 | 16000,03 | 16,08 | 15945,28 | 15992,58 | 86,39 | 16000,03 | 16000,03 | 14 | 611,53 | 627,61 |
| I700-17 | 20 | 16172,74 | 15,91 | 16132,83 | 16148,25 | 39,70 | 16171,67 | 16171,67 | 48 | 1190,84 | 1206,75 |
| I700-18 | 20 | 16414,8 | 14,4 | 16350,5 | 16389,79 | 61,10 | 16414,8 | 16414,8 | 58 | 1321,14 | 1335,54 |
| I700-19 | 20 | 16366,78 | 20,34 | 16321,91 | 16347,92 | 57.97 | 16366,78 | 16366,78 | 48 | 1134,83 | 1155,17 |
| I700-20 | 20 | 15434,21 | 19,14 | 15369,66 | 15413,97 | 68.64 | 15434,21 | 15434,21 | 34 | 987.08 | 1006.22 |

**Table 4** Computational results for test instances with 1000 facilities and 1000 clients

| *Name* | *Ratio* | $UB_{\text{Heur}}$ | $UB_{\text{Heur}}$ *Time* (s) | *LP* | $LP_{\text{Cuts}}$ | % $Gap_{\text{Cuts}}$ | *BLB* | *BUB* | *B&C&P* *nodes* | *B&C&P* *Time* (s) | *Tot. Time* (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| I1000-1 | 5 | 49537,2 | 780,29 | 49459,67 | 49481,11 | 42,76 | 49509,81 | 49509,81 | 6028 | 51271,42 | 52051,71 |
| I1000-2 | 5 | 50692,48 | 1042,11 | 50626,88 | 50654,17 | 44,23 | 50688,57 | 50688,57 | 6314 | 63156,37 | 64198,48 |
| I1000-3 | 5 | 47204,24 | 429,19 | 47153,15 | 47170,19 | 34,43 | 47202,64 | 47202,64 | 3118 | 27431,82 | 27861,01 |
| I1000-4 | 5 | 48871,06 | 1359,42 | 48807,68 | 48835,51 | 45,73 | 48868,54 | 48868,54 | 4572 | 36904,57 | 38263,99 |
| I1000-5 | 5 | 50753,6 | 594,03 | 50704,38 | 50728,03 | 60,39 | 50743,54 | 50743,54 | 584 | 4258,63 | 4852,66 |
| I1000-6 | 10 | 27843,64 | 403,73 | 27736,3 | 27791,79 | 63,39 | 27823,84 | 27823,84 | 2388 | 53816,33 | 54220,06 |
| I1000-7 | 10 | 27264,77 | 436,91 | 27162,18 | 27211,99 | 55,26 | 27252,32 | 27252,32 | 3168 | 59527,63 | 59964,54 |
| I1000-8 | 10 | 27375,37 | 285,44 | 27253,53 | 27325,56 | 59,11 | 27375,37 | 27375,37 | 5862 | 89632,67 | 89918,11 |
| I1000-9 | 10 | 26888,12 | 381,57 | 26755,22 | 26821,95 | 66,73 | 26857,09 | 26857,09 | 4154 | 62231,86 | 62613,43 |
| I1000-10 | 10 | 27241,26 | 155,00 | 27106,95 | 27153,69 | 58,39 | 27186,99 | 27186,99 | 1280 | 37458,02 | 37613,02 |
| I1000-11 | 15 | 22180,33 | 445,82 | 22070,4 | 22126,83 | 51,33 | 22169,82 | 22180,33 | 4862 | 100000 | 100445,82 |
| I1000-12 | 15 | 22160,39 | 405,26 | 22054,76 | 22096,64 | 39,65 | 22152,53 | 22160,39 | 4982 | 100000 | 100405,26 |
| I1000-13 | 15 | 22657,09 | 228,86 | 22552,13 | 22598,60 | 44,29 | 22657,09 | 22657,09 | 2076 | 81290,88 | 81519,74 |
| I1000-14 | 15 | 22327,62 | 242,58 | 22241,76 | 22270,97 | 41,58 | 22312,01 | 22312,01 | 1800 | 41580,07 | 41822,65 |
| I1000-15 | 15 | 22648,52 | 171,64 | 22523,36 | 22584,18 | 57,33 | 22629,44 | 22629,44 | 2334 | 74408,75 | 74580,39 |
| I1000-16 | 20 | 21331,81 | 119,04 | 21258,77 | 21283,95 | 34,47 | 21331,81 | 21331,81 | 986 | 29174,49 | 29293,53 |
| I1000-17 | 20 | 21188,89 | 79,09 | 21153,12 | 21177,51 | 68,18 | 21188,89 | 21188,89 | 72 | 3416,14 | 3495,23 |
| I1000-18 | 20 | 20713,44 | 48,12 | 20658,4 | 20690,11 | 57,62 | 20713,43 | 20713,43 | 74 | 2481,39 | 2529,51 |
| I1000-19 | 20 | 20537,45 | 44,11 | 20470,11 | 20509,10 | 57,90 | 20537,45 | 20537,45 | 190 | 4810,89 | 4855,00 |
| I1000-20 | 20 | 21560,86 | 102,84 | 21496,89 | 21515,44 | 29,00 | 21560,86 | 21560,86 | 446 | 7295,78 | 7398,62 |

**Table 5** Computational results for test instances with 300 facilities and 1500 clients

| *Name* | *Ratio* | $UB_{\text{Heur}}$ | $UB_{\text{Heur}}$ *Time* (s) | *LP* | $LP_{\text{Cuts}}$ | % $Gap_{\text{Cuts}}$ | *BLB* | *BUB* | *B&C&P nodes* | *B&C&P Time* (s) | *Tot. Time* (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| I3001500-1 | 5 | 158935,48 | 54,78 | 154624,95 | 154714,33 | 71,22 | 154750,44 | 154750,44 | 30 | 2279,86 | 2334,64 |
| I3001500-2 | 5 | 159301,74 | 47,62 | 159103,79 | 159169,25 | 42,85 | 159256,54 | 159256,54 | 34 | 1442,53 | 1490,15 |
| I3001500-3 | 5 | 157102,54 | 33,09 | 156876,02 | 156986,90 | 42,85 | 157016,09 | 157016,09 | 44 | 2053,34 | 2086,43 |
| I3001500-4 | 5 | 157417,03 | 45,41 | 157232,20 | 157360,58 | 73,73 | 157406,33 | 157406,33 | 24 | 1582,67 | 1628,08 |
| I3001500-5 | 5 | 160946,20 | 33,38 | 160807,26 | 160878,28 | 51,10 | 160946,22 | 160946,22 | 28 | 790,63 | 824,01 |
| I3001500-6 | 10 | 163847,89 | 37,47 | 156535,30 | 156608,52 | 85,25 | 156621,18 | 156621,18 | 18 | 907,25 | 944,72 |
| I3001500-7 | 10 | 158574,43 | 43,11 | 156928,59 | 156935,40 | 31,48 | 156950,22 | 156950,22 | 8 | 648,08 | 691,19 |
| I3001500-8 | 10 | 160445,08 | 29,98 | 157588,13 | 157685,79 | 98,17 | 157687,61 | 157687,61 | 6 | 677,58 | 707,56 |
| I3001500-9 | 10 | 158193,09 | 33,11 | 156835,94 | 156891,43 | 95,73 | 156893,90 | 156893,90 | 8 | 995,45 | 1028,56 |
| I3001500-10 | 10 | 162740,55 | 29,69 | 157631,34 | 157664,81 | 70,98 | 157678,49 | 157678,49 | 4 | 863,11 | 892,80 |
| I3001500-11 | 15 | 152145,52 | 38,31 | 149994,10 | 149995,16 | 16,54 | 150000,51 | 150000,51 | 2 | 384,49 | 422,80 |
| I3001500-12 | 15 | 157950,52 | 46,55 | 154805,01 | 154870.32 | 83,22 | 154883,49 | 154883,49 | 4 | 532,54 | 579,09 |
| I3001500-13 | 15 | 151593,03 | 40,30 | 151591,41 | 151593,03 | 100,00 | 151593,03 | 151593,03 | 0 | 107,81 | 148,11 |
| I3001500-14 | 15 | 155324,57 | 48,83 | 151788,11 | 151788,85 | 100,00 | 151788,85 | 151788,85 | 0 | 221,70 | 270,53 |
| I3001500-15 | 15 | 158297,33 | 44,48 | 156398,26 | 156402.85 | 23,93 | 156417,44 | 156417,44 | 6 | 578.30 | 622,78 |
| I3001500-16 | 20 | 157653,38 | 44,76 | 155489,08 | 155489.08 | 0,00 | 155489,36 | 155489,36 | 2 | 248,28 | 293,04 |
| I3001500-17 | 20 | 159623,66 | 51,82 | 155979,56 | 155995,60 | 29,83 | 156033,33 | 156033,33 | 4 | 235,36 | 287,18 |
| I3001500-18 | 20 | 158481,55 | 47,28 | 156765,72 | 156777,58 | 100,00 | 156777,58 | 156777,58 | 0 | 208,61 | 255,89 |
| I3001500-19 | 20 | 157696,77 | 49,83 | 155943,43 | 155946,26 | 100,00 | 155946,26 | 155946,26 | 0 | 268,09 | 317,92 |
| I3001500-20 | 20 | 159969,34 | 37,34 | 156405,82 | 156405,82 | 0,00 | 156407,23 | 156407,23 | 4 | 278,50 | 315,84 |

**Table 6** Computational results for OR-library test instances

| *Name* | $UB_{\text{Heur}}$ | $UB_{\text{Heur}}$ *Time* (s) | *LP* | $LP_{\text{Cuts}}$ | % $Gap_{\text{Cuts}}$ | *BLB* | *BUB* | *B&C&P nodes* | *B&C&P Time* (s) | *Tot. Time* (s) | *CPLEXTime time* (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CAPA-1 | 19240822,45 | 25,94 | 18832965,51 | 18859491,32 | 4,05 | 19240822,45 | 19240822,45 | 608 | 293,32 | 319,26 | 302,22 |
| CAPA-2 | 18558218,48 | 37,34 | 17899195,33 | 17913463,32 | 2,64 | 18439009,95 | 18439009,95 | 452 | 126,70 | 164,04 | 184,52 |
| CAPA-3 | 17765201,95 | 31,10 | 17443692,23 | 17538229,48 | 29,40 | 17765201,95 | 17765201,95 | 129 | 157,47 | 190,57 | 133,47 |
| CAPA-4 | 17120003,28 | 37,89 | 17160439,01 | 17160439,01 | 0,00 | 17160439,01 | 17160439,01 | 2 | 57,92 | 95,81 | 36,80 |
| CAPB-1 | 13689257,48 | 29,41 | 13606434,33 | 13617854,73 | 22,38 | 13657464,23 | 13657464,23 | 622 | 227,43 | 256,84 | 295,72 |
| CAPB-2 | 13360787,78 | 36,05 | 13310946,83 | 13318438,82 | 15,03 | 13360787,78 | 13360787,78 | 853 | 621,51 | 587,56 | 1195,17 |
| CAPB-3 | 13198556,45 | 40,31 | 13115539,25 | 13115587,18 | 0,01 | 13198556,45 | 13198556,45 | 5121 | 1247,34 | 1487,65 | 5691,25 |
| CAPB-4 | 13082516,51 | 46,09 | 13071059,91 | 13071059,91 | 0,00 | 13082516,51 | 13082516,51 | 5684 | 531,28 | 577,30 | 1714,62 |
| CAPC-1 | 11646596,68 | 37,32 | 11585301,47 | 11603283,63 | 29,33 | 11646596,68 | 11646596,68 | 368 | 137,43 | 169,75 | 228,16 |
| CAPC-2 | 11570340,30 | 33,22 | 11521991,70 | 11529562,18 | 15,66 | 11570340,30 | 11570340,30 | 1048 | 865,19 | 898,41 | 3974,36 |
| CAPC-3 | 11518743,75 | 42,88 | 11508583,02 | 11509371,39 | 7,75 | 11518743,75 | 11518743,75 | 345 | 126,58 | 169,46 | 233,03 |
| CAPC-4 | 11505767,40 | 35,76 | 11500107,25 | 11500372,41 | 4,68 | 11505767,40 | 11505767,40 | 48 | 37,65 | 72,81 | 54,97 |

**Table 7** Lower bounds for the test instances with 700 facilities and 700 clients

| *Name* | *LP* | $LP_{FC}$ | % $Gap_{FC}$ | $LP_{\text{Cuts}}$ | % $Gap_{\text{Cuts}}$ | *BUB* |
|---|---|---|---|---|---|---|
| I700-1 | 36839,96 | 36870,23 | 45.88 | 36880,10 | 60,84 | 36905,93 |
| I700-2 | 34250,16 | 34279,99 | 48,46 | 34284,06 | 55,08 | 34311,71 |
| I700-3 | 34253,73 | 34269,35 | 38,19 | 34275,57 | 53,39 | 34294,63 |
| I700-4 | 38029,87 | 38056,51 | 43,65 | 38060,86 | 50,77 | 38090,9 |
| I700-5 | 37741,12 | 37763,54 | 36,77 | 37770,65 | 48,42 | 37802,1 |
| I700-6 | 19821,97 | 19866,74 | 50.47 | 19876,60 | 61,59 | 19910,67 |
| I700-7 | 21195,93 | 21237,86 | 41,36 | 21248,75 | 52,10 | 21297,30 |
| I700-8 | 20592,18 | 20618,63 | 39,02 | 20628,74 | 53,94 | 20659,96 |
| I700-9 | 20894,19 | 20925,64 | 36,70 | 20934,76 | 47,34 | 20979,88 |
| I700-10 | 21960,88 | 22019,29 | 61,79 | 22028,48 | 71,51 | 22055,41 |
| I700-11 | 17005,24 | 17050,04 | 38,99 | 17067,18 | 53,90 | 17120,15 |
| I700-12 | 18048,2 | 18095,96 | 58,08 | 18110,54 | 75,82 | 18130,42 |
| I700-13 | 17163,15 | 17202.93 | 51,79 | 17211,76 | 63,28 | 17239,96 |
| I700-14 | 17247,16 | 17267.56 | 22,55 | 17289,06 | 46,31 | 17337,63 |
| I700-15 | 18069,79 | 18092,06 | 29,42 | 18101,94 | 42,47 | 18145,49 |
| I700-16 | 15945,28 | 15976,82 | 57,61 | 15992,58 | 86,39 | 16000,03 |
| I700-17 | 16132,83 | 16137,36 | 11,66 | 16148,25 | 39,70 | 16171,67 |
| I700-18 | 16350,5 | 16381,14 | 47,65 | 16389,79 | 61,10 | 16414,8 |
| I700-19 | 16321,91 | 16335,61 | 30,53 | 16347,92 | 57,97 | 16366,78 |
| I700-20 | 15369,66 | 15383,66 | 21,69 | 15413,97 | 68,64 | 15434,21 |

To validate the usefulness of the proposed separation procedure, Tables 7, 8 compare the lower bounds obtained at the root node by generating only the *Flow Cover* inequalities (columns $LP_{FC}$ and $\%Gap_{FC}$) with those obtained by generating both the *Flow Cover* inequalities and the cutting planes described in Sect. 7 (columns $LP_{\text{Cuts}}$ and $\%Gap_{\text{Cuts}}$). Column *BUB* reports on the best upper bound yielded by Branch-and-Cut-and-Price within the time limit. We observe that the cutting plane algorithm allows to significantly reduce the gap between lower and upper bound for all the tested instances. The amount of the reduction is in the range [7%,47%].

## 10 Conclusions

In this paper we presented a reformulation of *CFLP* based on *Mixed Dicut Inequalities*. Each Mixed Dicut inequality is in fact a minimum knapsack inequality and we refer to the convex hull of the feasible solutions of this knapsack problem as the *Mixed Dicut polytope*. To deal with large-scale instances, we developed a Branch-and-Cut-and-Price algorithm where the separation procedure consists of the complete enumeration of the facets of the Mixed Dicut polytope for a set of candidate Mixed Dicut Inequalities.

**Table 8** Lower bounds for the test instances with 1000 facilities and 1000 clients

| *Name* | *LP* | $LP_{FC}$ | % $Gap_{FC}$ | $LP_{\mathrm{Cuts}}$ | % $Gap_{\mathrm{Cuts}}$ | *BUB* |
|---|---|---|---|---|---|---|
| I1000-1 | 49459,67 | 49476,60 | 33,76 | 49481,11 | 42,76 | 49509,81 |
| I1000-2 | 50626,88 | 50650,19 | 37,78 | 50654,17 | 44,23 | 50688,57 |
| I1000-3 | 47153,15 | 47166,85 | 27,68 | 47170,19 | 34,43 | 47202,64 |
| I1000-4 | 48807,68 | 48831,79 | 39,61 | 48835,51 | 45,73 | 48868,54 |
| I1000-5 | 50704,38 | 50725,68 | 54,39 | 50728,03 | 60,39 | 50743,54 |
| I1000-6 | 27736,3 | 27783,87 | 54,34 | 27791,79 | 63,39 | 27823,84 |
| I1000-7 | 27162,18 | 27205,96 | 48,57 | 27211,99 | 55,26 | 27252,32 |
| I1000-8 | 27253,53 | 27309,75 | 46,14 | 27325,56 | 59,11 | 27375,37 |
| I1000-9 | 26755,22 | 26809,37 | 53,15 | 26821,95 | 66,73 | 26857,09 |
| I1000-10 | 27106,95 | 27142,36 | 44,24 | 27153,69 | 58,39 | 27186,99 |
| I1000-11 | 22070,4 | 22109,56 | 35,62 | 22126,83 | 51,33 | 22180,33 |
| I1000-12 | 22054,76 | 22076,71 | 20,78 | 22096,64 | 39,65 | 22160,39 |
| I1000-13 | 22552,13 | 22585,98 | 32,25 | 22598,60 | 44,29 | 22657,09 |
| I1000-14 | 22241,76 | 22262,74 | 29,86 | 22270,97 | 41,58 | 22312,01 |
| I1000-15 | 22523,36 | 22569,80 | 43,78 | 22584,18 | 57,33 | 22629,44 |
| I1000-16 | 21258,77 | 21269,97 | 15,33 | 21283,95 | 34,47 | 21331,81 |
| I1000-17 | 21153,12 | 21167,83 | 41,12 | 21177,51 | 68,18 | 21188,89 |
| I1000-18 | 20658,40 | 20682,04 | 42,96 | 20690,11 | 57,62 | 20713,43 |
| I1000-19 | 20470,11 | 20501,97 | 47,31 | 20509,10 | 57,90 | 20537,45 |
| I1000-20 | 21496,89 | 21506,18 | 14,52 | 21515,44 | 29,00 | 21560,86 |

A natural evolution of this work is to study separation algorithms based on facet-enumeration which are able to deal with base inequalities having a larger support. Another research direction will consists of applying facet-enumeration techniques to knapsack and to mixed knapsack polytopes in the more general context of Mixed-Integer Programming problems.

**Acknowledgements** The authors wish to thank Antonio Sassano and Sara Mattia for their helpful suggestions.

## References

1. Aardal, K.: On the solution of one and two-level capacitated facility location problems by the cutting plane approach. Ph.D. thesis, Université Catholique de Louvain, Louvain-la-Neuve, Belgium (1992)
2. Aardal, K.: Capacitated facility location: separation algorithms and computational experience. Math. Program. **81**, 149–175 (1998)
3. Aardal, K., Pochet, Y., Wolsey, L.A.: Capacitated facility location: valid inequalities and facets. Math. Operat. Res. **20**, 562–582 (1995)
4. Anbil, R., Barahona, F.: The volume algorithm: producing primal solutions with a subgradient method. Math. Program. Ser. A **87**, 385–399 (2000)
5. Avella, P., Boccia, M., Sforza, A., Vasiliev, I.: An effective heuristic for large-scale capacitated plant location problems. Technical Report, available online at http://www.ing.unisannio.it/boccia/ (2004)

6. Avella, P., Sassano, A., Vasil'ev, I.: Computational study of large-scale p-median problems. Math. Program. **109**(1), 89–114 (2007)
7. Barahona, F., Chudak, F.A.: Near-optimal solution to large scale facility location problems. Internal report, IBM research division, T.J. Watson Research Center, RC 21606 (1999)
8. Barber, C.B., Dobkin, D.P., Huhdanpaa, H.T.: The quickhull algorithm for convex hulls. ACM Trans. Math. Softw. **22**(4), 469–483 (1996); http://www.qhull.org
9. Beasley, J.E.: OR-Library: distributing test problems by electronic mail. J. Operat. Res. Soc. **41**(11), 1069–1072 (1990)
10. Beasley, J.E.: Lagrangean heuristics for location problems. Eur. J. Operat. Res. **65**, 383–399 (1993)
11. Bertsimas, D., Tsitsiklis, J.N.: Introduction to Linear Optimization. Athena Scientific, Belmont (1997)
12. Chudak, F.A., Williamson, D.P.: Improved approximation algorithms for capacitated facility location problems. In: 7th International IPCO Conference Proceedings, pp. 99–113 (1999)
13. Cornuejols, G., Sridharan, R., Thizy, J.M.: A comparison of heuristics and relaxations for the capacitated plant location problem. Eur. J. Oper. Res. **50**, 280–297 (1991)
14. Klose, A., Görtz, S.: A branch-and-price algorithm for the capacitated facility location problem. Eur. J. Operat. Res. (2006, forthcoming). Available online at http://www.sciencedirect.com
15. Korupolu, M.R., Plaxton, C.G., Rajaraman, R.: Analysis of a local search heuristic for facility location problems. DIMACS Technical Report, pp. 98–30 (1998)
16. Leung, J.M.Y., Magnanti, T.L.: Valid inequalities and facets of the capacitated plant location problem. Math. Program. **44**, 271–291 (1989)
17. Marchand, H., Wolsey, L.A.: The 0-1 knapsack problem with a single continuous variable. Math. Program. **85**, 15–33 (1999)
18. Nemhauser, G.L., Wolsey, L.A.: Integer and Combinatorial Optimization. Wiley, New York (1988)
19. Ortega, F., Wolsey, L.: A branch and cut algorithm for the single commodity uncapacitated fixed charge network flow problem. Networks **41**, 143–158 (2003)